Documentation 8.2

ZABBIX

05.11.2026

Contents

Manual de usuario **5**

Bienvenido a la documentación de Zabbix **5**

Primeros pasos con Zabbix . . . 5
Zabbix Cloud . . . 5
Aplicación móvil . . . 5
Centro de desarrolladores . . . 5
Comunidad y otros recursos . . . 5
Instalación y primeros pasos . . . 6
Instalación de la interfaz web . . . 6
Guías de inicio rápido . . . 11
Instalación: guías adicionales . . . 85
Actualización . . . 167
Problemas conocidos . . . 184
Requisitos . . . 192
Novedades y notas de actualización: 8.0 . . . 203
Notas de actualización . . . 203
Cambios incompatibles . . . 203
Otro . . . 203
Novedades y notas de actualización: 8.0.x . . . 203
Appliance de Zabbix . . . 203
Procesos . . . 207
server . . . 209
Proxy . . . 236
agent . . . 282
Agent 2 . . . 313
Java gateway . . . 378
Remitente . . . 383
Obtener . . . 385
JS . . . 386
Servicio web . . . 387
Adaptador bridge . . . 389
MCP server . . . 393
Variables de entorno . . . 400
Configuración . . . 402
Hosts y grupos de hosts . . . 413
Items . . . 431
Detección de problemas con triggers . . . 658
Visualización . . . 711
Templates y grupos de templates . . . 737
Templates listos para usar . . . 737
Notificaciones al producirse eventos . . . 751
Macros . . . 801
Usuarios y grupos de usuarios . . . 817
Informes programados . . . 828
Exportación de datos . . . 832
Exportación/importación de configuración . . . 841
Expresiones regulares . . . 908
Eventos y resolución . . . 914
Reconocimiento y supresión de problemas . . . 916
Cierre manual de problemas . . . 920

Períodos de mantenimiento . . . 921
Etiquetado . . . 927
Complex event processing . . . 930
Correlación de eventos . . . 934
Descubrimiento . . . 941
Descubrimiento de red . . . 941
Autoregistro del agent activo . . . 949
Descubrimiento de bajo nivel . . . 953
Supervisión de máquinas virtuales . . . 977
Claves de item de monitoreo de VMware . . . 978
Campos de clave de descubrimiento de máquinas virtuales . . . 996
Ejemplos JSON para items de VMware . . . 1001
Ejemplo de configuración de monitorización de VMware . . . 1005
Supervisión web . . . 1010
Elementos de monitorización web . . . 1019
Escenario de la vida real . . . 1020
Supervisión de servicios . . . 1027
Árbol de servicios . . . 1028
SLA . . . 1032
Ejemplo de configuración . . . 1034
Actualización de la monitorización de servicios . . . 1038
Cifrado y seguridad . . . 1039
Cifrado mediante certificados . . . 1048
Cifrado mediante claves precompartidas . . . 1055
Conexión segura a la base de datos . . . 1058
Conexión segura entre el frontend y el server . . . 1071
Control de acceso . . . 1075
Criptografía . . . 1079
Asegurar el web server . . . 1080
Almacenamiento de secretos . . . 1083
Confianza en el certificado TLS de Google Chrome . . . 1092
Buenas prácticas de seguridad . . . 1094
Interfaz web . . . 1094
Menú . . . 1095
Secciones del frontend . . . 1105
Configuración de usuario . . . 1381
Búsqueda global . . . 1385
Modo de mantenimiento del frontend . . . 1387
Parámetros de la página . . . 1388
Definiciones . . . 1391
Idiomas adicionales del frontend . . . 1391
Creación de tu propio tema . . . 1392
Modo de depuración . . . 1394
Cookies utilizados por Zabbix . . . 1394
Zonas horarias . . . 1395
Restablecimiento de contraseña . . . 1396
Período de tiempo y selectores de host . . . 1397
Parámetros de configuración: frontend de Zabbix . . . 1398
API . . . 1405
Referencia de métodos . . . 1410
Apéndice 1. Comentario de referencia . . . 2302
Apéndice 2. Cambios de 7.4 a 8.0 . . . 2308
Apéndice 3. Cambios en 8.0 . . . 2308
Extensiones . . . 2308
Módulos cargables . . . 2310
Módulos de frontend . . . 2319
Apéndices . . . 2320
Instalación y configuración . . . 2321
Items . . . 2367
Funciones compatibles . . . 2408
Macros . . . 2408
Compatibilidad de versiones . . . 2446
Sintaxis del período de tiempo . . . 2447

Símbolos de unidades . . . 2448
Ejecución de comandos . . . 2449
Ejemplos de escape . . . 2450
Qué es Zabbix . . . 2452

Zabbix Cloud **2456**

Zabbix Cloud **2456**
Configurar y administrar Zabbix Cloud . . . 2456
Conozca Zabbix Cloud . . . 2457
Desplegar Zabbix en la nube . . . 2457
Configuración del nodo . . . 2466
Descripción general . . . 2467
Pestaña Descripción general . . . 2468
Pestaña de filtros de acceso . . . 2471
Pestaña Encryption . . . 2472
Pestaña Backups . . . 2472
Pestaña History . . . 2473
Pestaña Mantenimiento . . . 2474
Pestaña Upgrade . . . 2474
Añadir usuarios . . . 2476
Introducción . . . 2476
Añadir usuarios a las organizaciones . . . 2476
Añadir usuarios a los nodos . . . 2477
Diferencias clave de Zabbix Cloud . . . 2478
Introducción . . . 2478
Registro de auditoría . . . 2480
Configurar un proxy de Zabbix Cloud . . . 2482

Aplicación móvil **2490**

Aplicación móvil **2490**
Configurar Zabbix para la aplicación . . . 2490
Más información sobre la aplicación . . . 2491
Configurar Zabbix para la aplicación . . . 2491
Descripción general de la aplicación . . . 2492
Acceso móvil seguro . . . 2500

Centro de desarrolladores **2501**

Centro para desarrolladores **2501**
Construir sobre Zabbix . . . 2501
Creación de un plugin . . . 2501
Interfaces de plugin . . . 2507
Biblioteca Python para Zabbix . . . 2509
Instalación . . . 2509
Guía de inicio rápido . . . 2509
Usar la API de Zabbix . . . 2513
Recopilar datos del agent de Zabbix . . . 2516
Enviar datos al servidor o proxy de Zabbix . . . 2517
Registro de depuración . . . 2521
Módulos . . . 2522
Estructura de archivos del módulo . . . 2522
Widgets . . . 2530
Tutoriales . . . 2538
Ejemplos . . . 2565
Cambios en el desarrollo de extensiones . . . 2566

Páginas de manual de Zabbix **2566**
zabbix_agent2 . . . 2566
zabbix_agentd . . . 2567
zabbix_get . . . 2569
zabbix_js . . . 2571
zabbix_proxy . . . 2572
zabbix_sender . . . 2574

zabbix_server 2579
zabbix_web_service 2582

Aviso de derechos de autor **2583**

Manual de usuario

Bienvenido a la documentación de Zabbix

Su recurso de referencia para trabajar con la monitorización de Zabbix, desde configuraciones básicas hasta configuraciones avanzadas. Este manual cubre todo lo necesario para instalar, configurar y operar Zabbix.

Novedades en Zabbix 8.0

Primeros pasos con Zabbix

Instalación Instrucciones paso a paso para instalar Zabbix en su plataforma preferida, cubriendo varios sistemas operativos y configuraciones de bases de datos.

Requisitos Una lista de plataformas soportadas y requisitos previos de software para ayudarle a preparar su entorno para una implementación exitosa de Zabbix.

Guías de inicio rápido Guías prácticas y concisas, orientadas a tareas, que le muestran los conceptos básicos: desde sus primeros pasos de configuración hasta la recepción de su primera alerta de problema.

Zabbix Cloud

Comience a utilizar Zabbix Cloud Instrucciones guiadas para poner en marcha rápidamente su instancia de Zabbix Cloud, incluida la configuración inicial y la conexión de sus dispositivos monitorizados.

Zabbix Cloud vs. on-premises Compare las diferencias en gestión, escalabilidad y mantenimiento entre las dos implementaciones, y descubra cuál puede soportar su flujo de trabajo de la manera más fluida.

Explora Zabbix Cloud Obtén una visión general de Zabbix Cloud, incluidas sus características clave, ventajas y lo que incluye una suscripción. Para preguntas comunes sobre la configuración y el uso, consulta las preguntas frecuentes.

Aplicación móvil

Obtén la aplicación Instala la aplicación móvil de Zabbix (próximamente) en tu dispositivo y comienza a supervisar tu entorno de Zabbix desde cualquier lugar.

Configura Zabbix para la aplicación Instala el adaptador de puente de Zabbix para permitir que los dispositivos móviles se conecten a Zabbix. A continuación, añade tu dispositivo y activa las notificaciones push.

Descripción general de la aplicación Aprenda los conceptos básicos del uso de la aplicación móvil de Zabbix.

Centro de desarrolladores

Módulos de frontend Guías y referencias para crear módulos personalizados que amplíen o modifiquen el frontend de Zabbix para adaptarse a casos de uso específicos.

Widgets Una descripción de la estructura y lógica de los widgets, con instrucciones para crear elementos personalizados del dashboard adaptados a sus necesidades.

Plugins Una visión general sobre cómo desarrollar y gestionar plugins de Zabbix para ampliar la funcionalidad o integrarse con sistemas externos.

Comunidad y otros recursos

Foros de Zabbix Un espacio para intercambiar ideas, encontrar soluciones y compartir experiencias en todos los niveles de experiencia en Zabbix.

Blog de Zabbix Noticias, tutoriales y casos de estudio seleccionados por el equipo de Zabbix y colaboradores de la comunidad.

Vídeos prácticos Una biblioteca de vídeos con demostraciones, debates y consejos para ayudarle a sacar el máximo partido a Zabbix.

Instalación y primeros pasos

La siguiente secuencia describe el proceso de configuración de Zabbix:

1. Verifique que el software requerido esté instalado y en ejecución.
2. Instale Zabbix.
3. Instale la interfaz web de Zabbix.
4. Inicie la supervisión.

A continuación se proporcionan los detalles de cada paso.

Software necesario Zabbix funciona correctamente con el siguiente software (a menudo denominado pila LAMP):

- **Sistema operativo Linux:** Ubuntu, Debian, CentOS u otros
- **Web server:** Apache o NGINX
- **Database server:** MySQL/MariaDB o PostgreSQL
- **PHP**

En la mayoría de los sistemas Linux modernos, es posible que PHP ya esté instalado. En ese caso, solo tendrá que configurar el database server y el web server.

Para obtener información más detallada, como las versiones de software compatibles, consulte Software necesario.

Instalar Zabbix La forma **más sencilla** de instalar Zabbix es mediante **paquetes de la distribución** (como alternativa, desde contenedores). Los paquetes instalan dependencias de software menores.

Las instrucciones de instalación precisas, adaptadas a su sistema operativo, están disponibles en el **sitio web de Zabbix**.

Seleccione su plataforma y siga las instrucciones proporcionadas. Se recomienda instalar un Zabbix server, un Zabbix agent 2 y el frontend (interfaz web) para comenzar. Más adelante puede añadir proxies de Zabbix y más agentes.

Parte del proceso de instalación consiste en crear una base de datos (para almacenar los datos de configuración y del historial) e importar los datos iniciales en ella.

Instalar la interfaz web de Zabbix Complete los pasos del **asistente de instalación** de la interfaz web.

Se le solicitará la contraseña de la base de datos (creada en el paso anterior), para que el frontend pueda comunicarse con la base de datos.

Iniciar la monitorización Inicie sesión en la interfaz web de Zabbix.

Utilice las **guías de inicio rápido** para comenzar a monitorizar recursos populares (Linux, Apache, MySQL u otros). Estas guías rápidas explican con qué templates debe trabajar para comenzar a monitorizar recursos populares.

Puede comenzar con la guía de inicio rápido para monitorizar un servidor Linux.

Instalación de la interfaz web

Esta sección proporciona instrucciones para instalar la interfaz web de Zabbix.
Siga estos pasos después de instalar el backend de Zabbix.

El frontend de Zabbix está escrito en PHP y requiere un web server compatible, como Apache o NGINX, con PHP configurado.
Consulte los requisitos del frontend para obtener información sobre el software y las bibliotecas necesarios.

Para utilizar un idioma distinto del inglés en la interfaz web de Zabbix, debe instalarse su configuración regional en el web server, junto con la extensión gettext de PHP.
Tenga en cuenta que cambiar el idioma puede modificar el formato de la hora y la fecha.

Para configurar SSL para el frontend de Zabbix, consulte Configuración de SSL para el frontend de Zabbix.

Asistente de instalación

Para instalar la interfaz web, abra la URL del frontend de Zabbix en un navegador:

- Para Apache: `http://<server_ip_or_name>/zabbix`
- Para NGINX: `http://<server_ip_or_name>`

Pantalla de bienvenida

El asistente de instalación del frontend se abre en la pantalla de bienvenida.

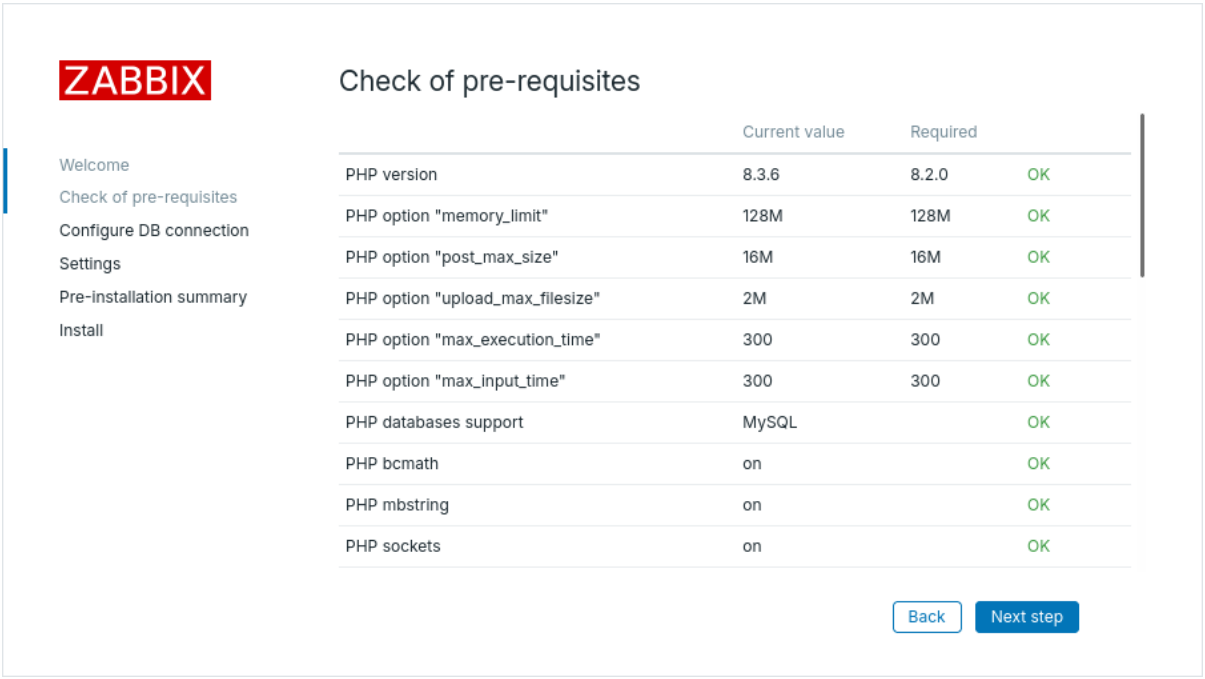


El idioma predeterminado de todo el sistema es el inglés (en_US), con el formato de hora y fecha de EE. UU. Para cambiar el idioma predeterminado de todo el sistema, seleccione el idioma deseado en el menú desplegable *Idioma predeterminado*.

Haga clic en el botón *Siguiente paso* para continuar.

Comprobación de los requisitos previos

Asegúrese de que se cumplan todos los requisitos previos obligatorios del frontend de Zabbix.



Consulte Requisitos para obtener una lista de las extensiones PHP necesarias y opcionales, la versión de PHP requerida y las bases de datos compatibles. Tenga en cuenta que los requisitos previos opcionales que no se cumplan se muestran con el estado *Advertencia*, y el proceso de configuración puede continuar aunque no se cumplan.

Valores necesarios para las directivas de configuración de PHP en el archivo `php.ini` utilizado por el servidor web:

```
memory_limit = 128M ; Valor mínimo
post_max_size = 16M ; Valor mínimo
upload_max_filesize = 2M ; Valor mínimo
```

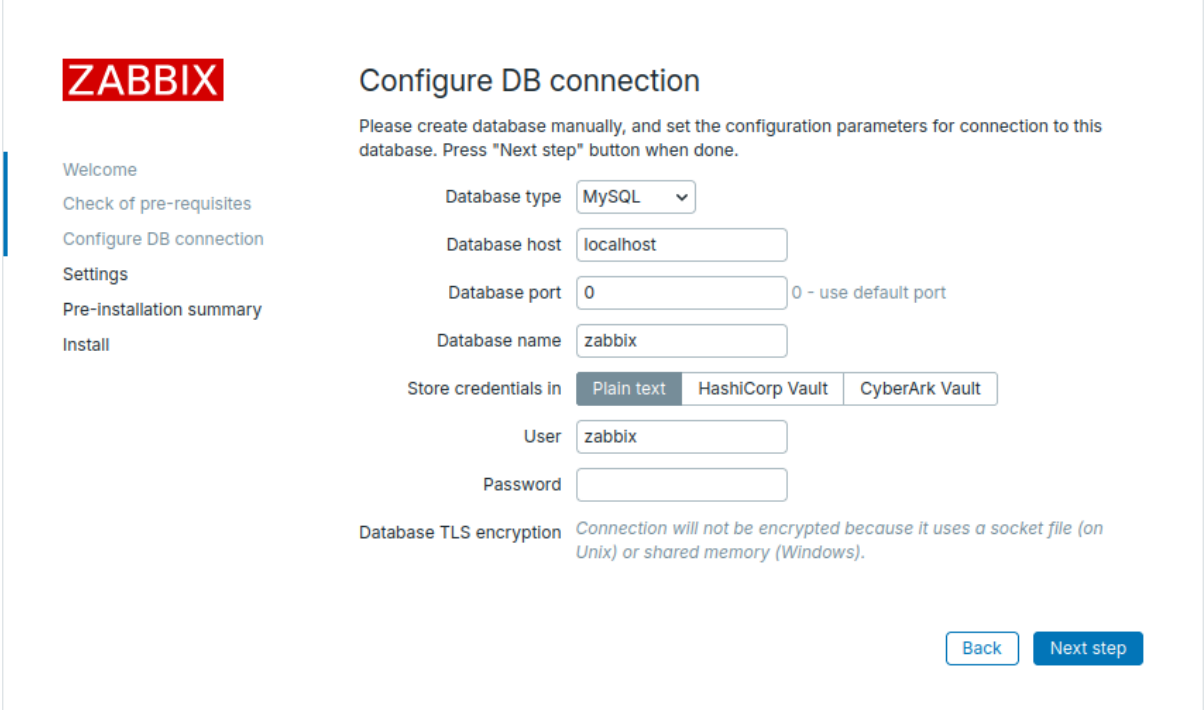
```
max_execution_time = 300 ; Valor mínimo; también se permiten los valores 0 y -1
max_input_time = 300 ; Valor mínimo; también se permiten los valores 0 y -1
session.auto_start = 0 ; Debe estar deshabilitado
arg_separator.output = "&" ; También se permite "&amp;"
```

Attention:
Si es necesario cambiar el usuario o el grupo del servidor web, verifique que el nuevo usuario o grupo tenga los permisos necesarios para el directorio de sesiones de PHP; de lo contrario, es posible que la configuración de Zabbix no pueda continuar.

Si se cumplen todos los requisitos previos obligatorios, continúe con el siguiente paso.

Configurar la conexión a la base de datos

Introduzca los datos para conectarse a la base de datos.



Si el campo *Database host* se deja vacío, se utiliza el socket de dominio Unix predeterminado.

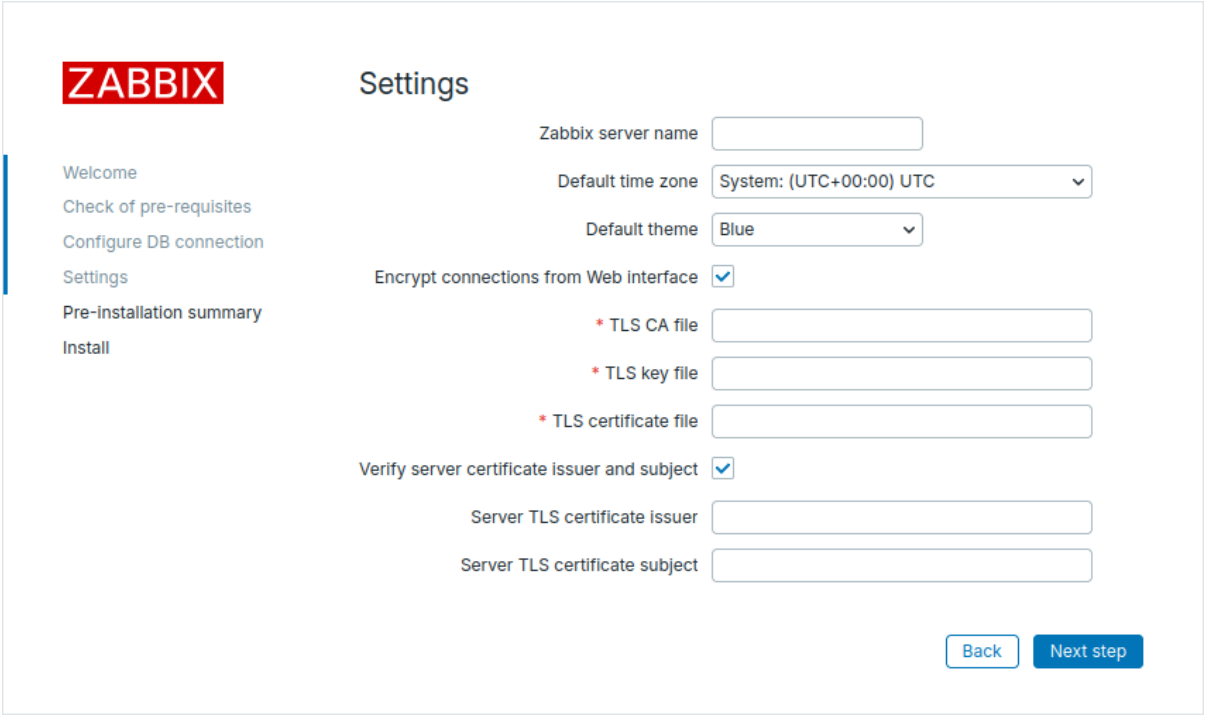
En MySQL, introducir `localhost` en el campo *Database host* también da lugar a una conexión mediante el socket de dominio Unix predeterminado. El asistente no proporciona ninguna opción para configurar el socket de la base de datos. Para utilizar un socket personalizado, configúrelo en los ajustes del Zabbix server (por ejemplo, mediante el parámetro DBSocket= en el archivo de configuración del Zabbix server (`zabbix_server.conf`)). Esto mantiene el frontend coherente con los ajustes de conexión entre el server y la base de datos.

En PostgreSQL, el campo *Database host* puede contener varias entradas `host:port` separadas por comas. La cadena se divide en partes y cada puerto se valida de la misma manera que el campo *Database port*. Después de una validación correcta, la cadena se analiza para obtener los hosts y los puertos con los que se construye la cadena de conexión. Si no se define un puerto para un host en la lista separada por comas, se utiliza el valor de *Database port* como puerto predeterminado. Si se especifica una ruta de socket (por ejemplo, `/var/run/pgbouncer`), se utiliza ese socket de dominio Unix.

Para configurar una conexión segura con la base de datos, seleccione la casilla *Database TLS encryption* y vaya a Frontend to the database para configurar los campos. La casilla *Database TLS encryption* está deshabilitada en los siguientes casos: - En MySQL, *Database host* está establecido en *localhost*. - En PostgreSQL, el valor del campo *Database host* comienza con una barra diagonal o el campo está vacío.

Para configurar Zabbix para recuperar secretos de HashiCorp Vault, seleccione *HashiCorp Vault* en *Store credentials in* y vaya a HashiCorp configuration para completar los campos ampliados. Para configurar Zabbix para recuperar secretos de CyberArk Vault, seleccione *CyberArk Vault* en *Store credentials in* y vaya a CyberArk configuration para completar los campos ampliados.

Configuración



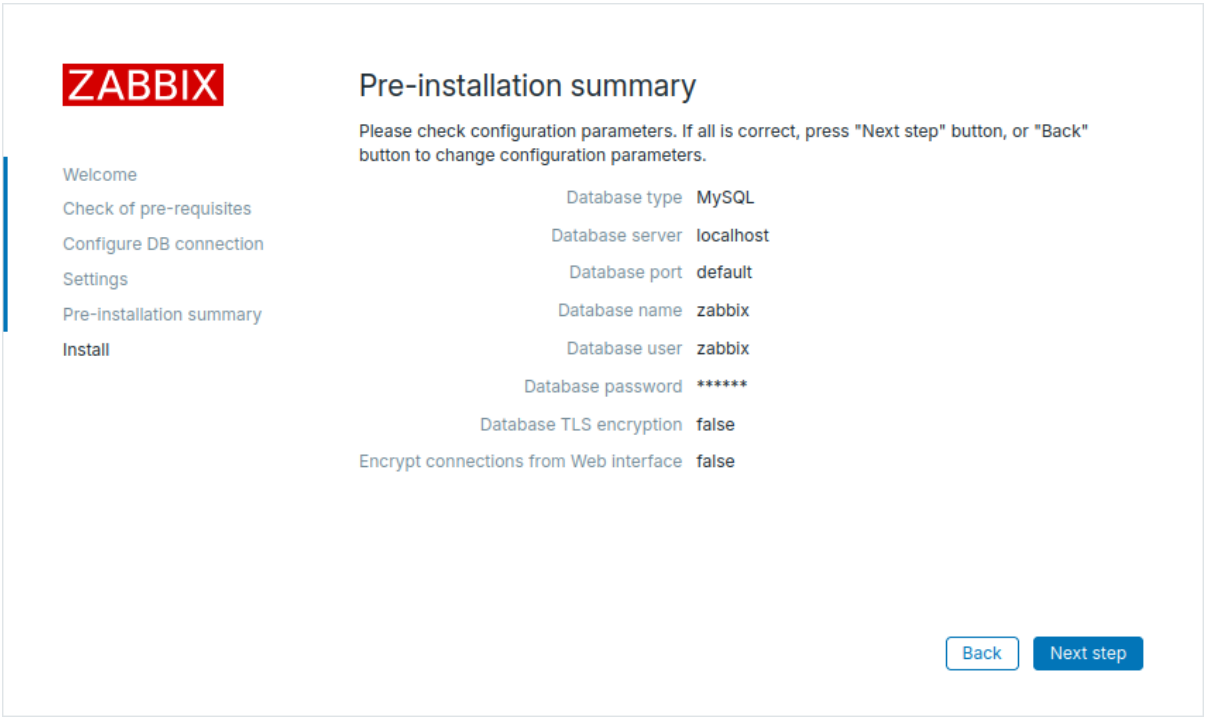
Introducir un nombre para el Zabbix server es opcional; sin embargo, si se proporciona, se mostrará en la barra de menús y en los títulos de las páginas.

Establezca la zona horaria predeterminada y el tema para el frontend.

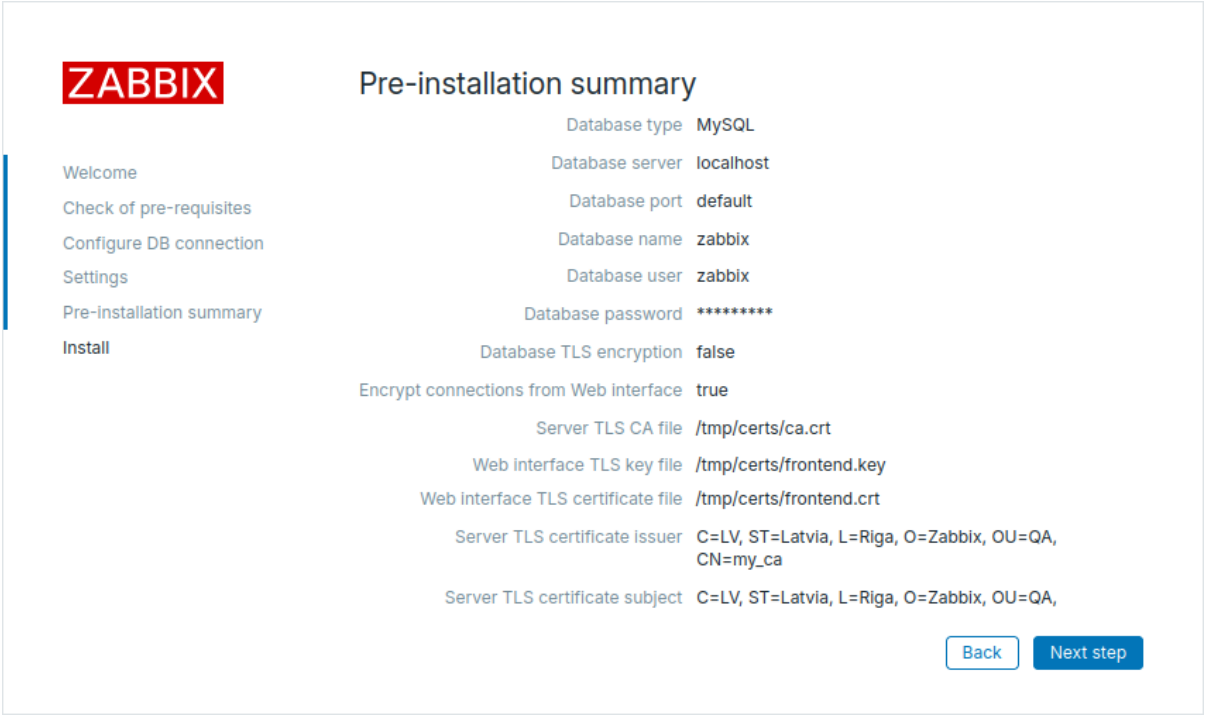
Para configurar conexiones TLS entre el frontend de Zabbix y el Zabbix server, seleccione la casilla *Cifrar las conexiones desde la interfaz web* y continúe con la configuración de la conexión TLS para completar los campos ampliados.

Resumen previo a la instalación

Revise un resumen de la configuración.



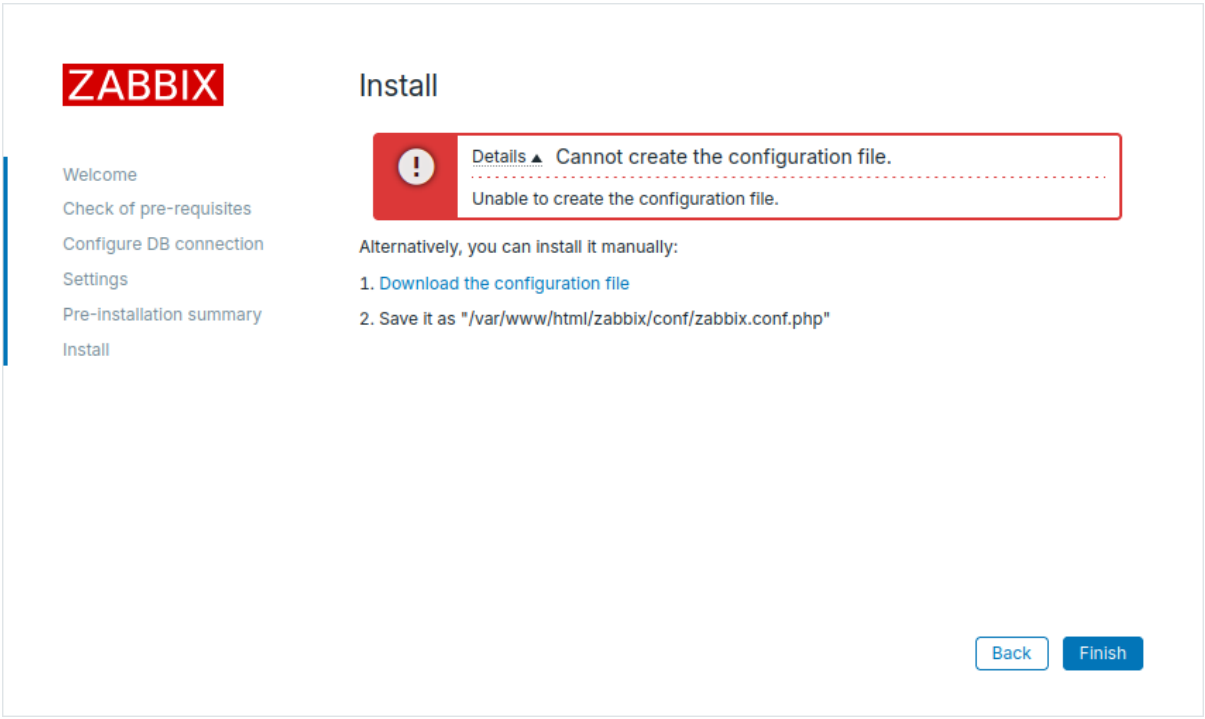
Si TLS está configurado, el asistente mostrará sus detalles.



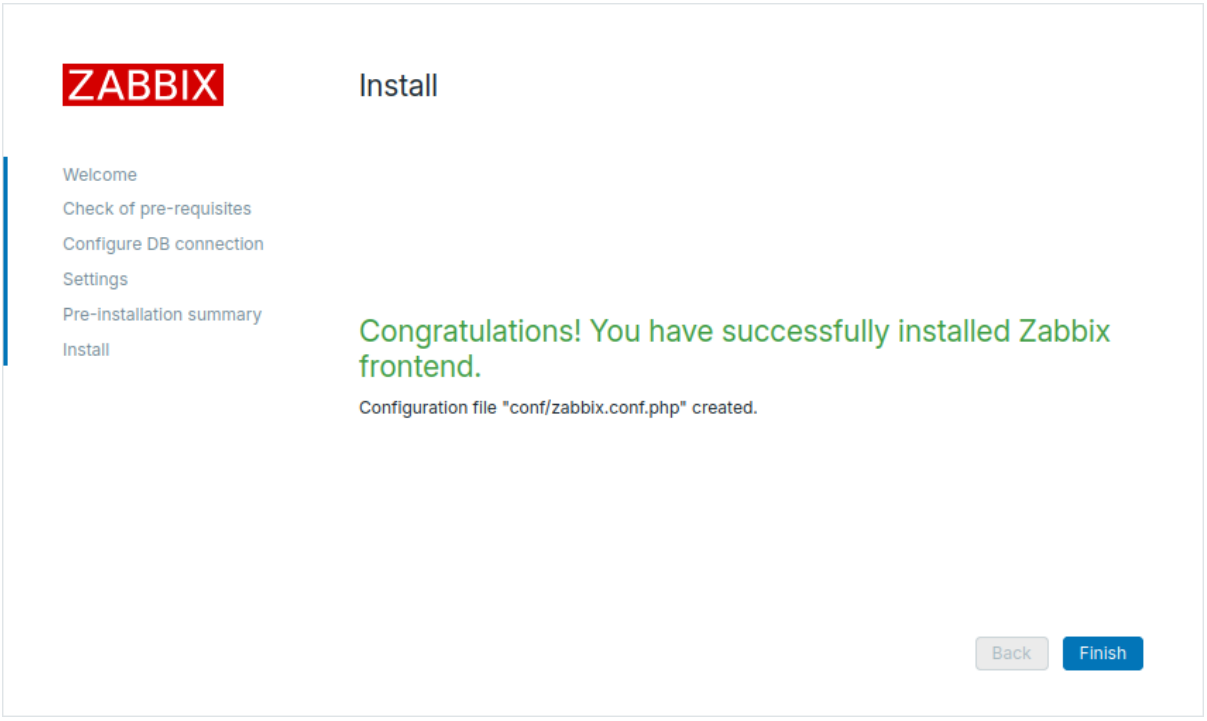
Instalar

Si el usuario del web server tiene acceso de escritura al directorio `conf/`, el archivo de configuración se guardará automáticamente y podrá continuar inmediatamente con el siguiente paso.

Si el usuario del web server no tiene acceso de escritura al directorio `conf/`, debe descargar el archivo de configuración y colocarlo en el directorio `conf/`, dentro del directorio de documentos HTML del web server donde copió los archivos PHP de Zabbix.

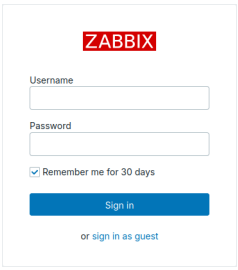


Finalice la instalación.



Iniciar sesión

Inicie sesión con las credenciales predeterminadas: - Nombre de usuario: *Admin* - Contraseña: *zabbix*



Archivo de configuración del frontend

Después de la instalación, puede volver a configurar el frontend de Zabbix actualizando los parámetros del archivo de configuración del frontend `zabbix.conf.php`.

La ruta predeterminada al archivo de configuración depende del sistema operativo y del método de instalación de Zabbix:

- Para la instalación desde paquetes: `/usr/share/zabbix/ui/conf/zabbix.conf.php`
- Para la instalación desde el código fuente: `/var/www/html/zabbix/conf/zabbix.conf.php`
- Para el appliance: `/etc/zabbix/web/zabbix.conf.php`

Consulte las opciones del archivo de configuración para obtener más información sobre la configuración del frontend de Zabbix.

Guías de inicio rápido

Después de haber instalado Zabbix y su interfaz web, es hora de comenzar la monitorización.

Utilice estas guías rápidas para comenzar a monitorizar recursos populares:

- Monitorización de un server Linux
- Monitorización de un server web Apache
- Monitorización de un server MySQL

Las guías de inicio rápido se centran en el uso de templates oficiales de Zabbix para la monitorización. Un template es un conjunto preconfigurado de elementos de monitorización. Un template permite escalar la monitorización: puede probar un template en un único objetivo de monitorización (dispositivo, aplicación, etc.), ajustarlo y, después, aplicarlo a varios objetivos de monitorización similares.

Aunque los templates son adecuados para la monitorización lista para usar, también puede interesarle personalizar sus elementos.

Para comprender mejor los elementos básicos que componen un template, consulte Configuración básica.

Configuración básica

Estas guías presentan los elementos básicos de la configuración de Zabbix:

- Inicio de sesión y configuración de usuario
- Nuevo host
- Nuevo item
- Nuevo trigger
- Recepción de notificaciones de problemas
- Nuevo template

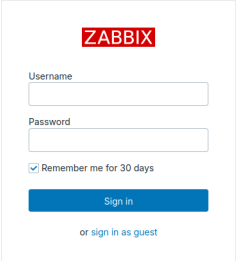
Inicio de sesión y configuración de usuario

Resumen

En esta sección, aprenderá cómo iniciar sesión y configurar un usuario del sistema en Zabbix.

Inicio de sesión

Abra la URL del frontend de Zabbix en un navegador: - Para Apache: http://<server_ip_or_name>/zabbix - Para NGINX: http://<server_ip_or_name>



Esta es la pantalla de bienvenida de Zabbix. Introduzca el nombre de usuario **Admin** y la contraseña **zabbix** para iniciar sesión como superusuario de Zabbix. Se concederá acceso a todas las secciones del menú.

Por motivos de seguridad, se recomienda encarecidamente cambiar la contraseña predeterminada de la cuenta Admin inmediatamente después del primer inicio de sesión.

Inicio de sesión persistente

Para permanecer conectado hasta 30 días, seleccione *Recordarme durante 30 días* antes de hacer clic en *Iniciar sesión*.

Recordarme durante 30 días activado:

- Su sesión permanece activa durante 30 días.
- *Cierre de sesión automático* se anula, manteniéndolo conectado hasta que finalice el período.
- Se iniciará sesión automáticamente en visitas futuras dentro de los 30 días sin volver a introducir credenciales.

Recordarme durante 30 días desactivado:

- Se borra cualquier inicio de sesión automático habilitado previamente.
- La sesión expirará según el intervalo de *Cierre de sesión automático* configurado.

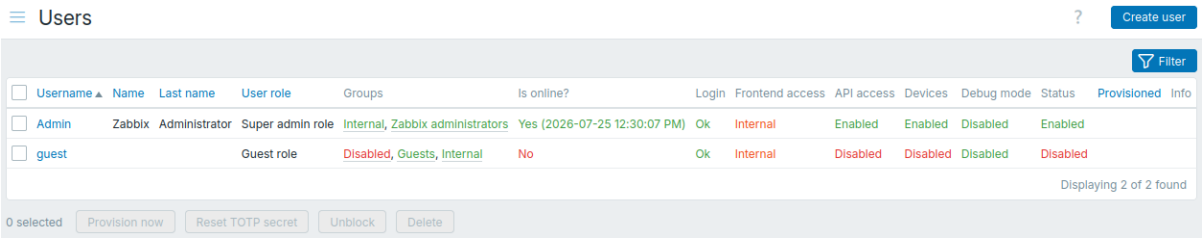
Protección contra ataques de fuerza bruta

En caso de cinco intentos de inicio de sesión fallidos consecutivos, la interfaz de Zabbix se pausará durante 30 segundos para evitar ataques de fuerza bruta y de diccionario.

La dirección IP de un intento de inicio de sesión fallido se mostrará después de un inicio de sesión correcto.

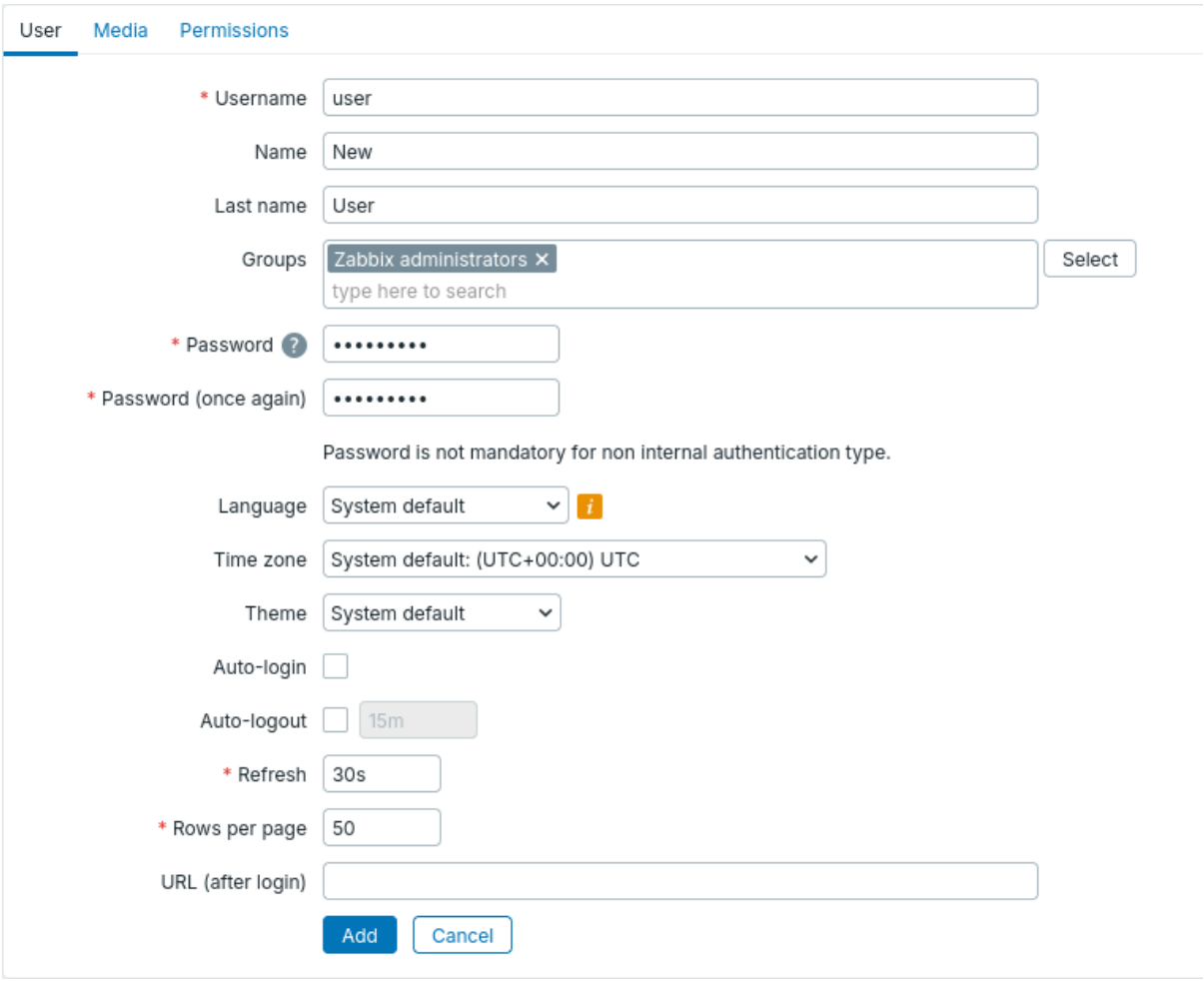
Añadir usuario

Para ver información sobre los usuarios, vaya a *Users* > *Users* en el menú vertical de la barra lateral.



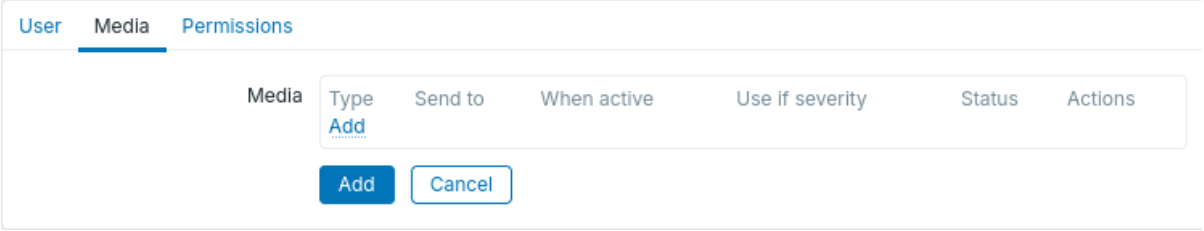
Para añadir un usuario nuevo, seleccione *Create user* en la esquina superior derecha.

En el formulario de usuario nuevo, asegúrese de añadir su usuario a uno de los grupos de usuarios existentes, por ejemplo 'Zabbix administrators'.



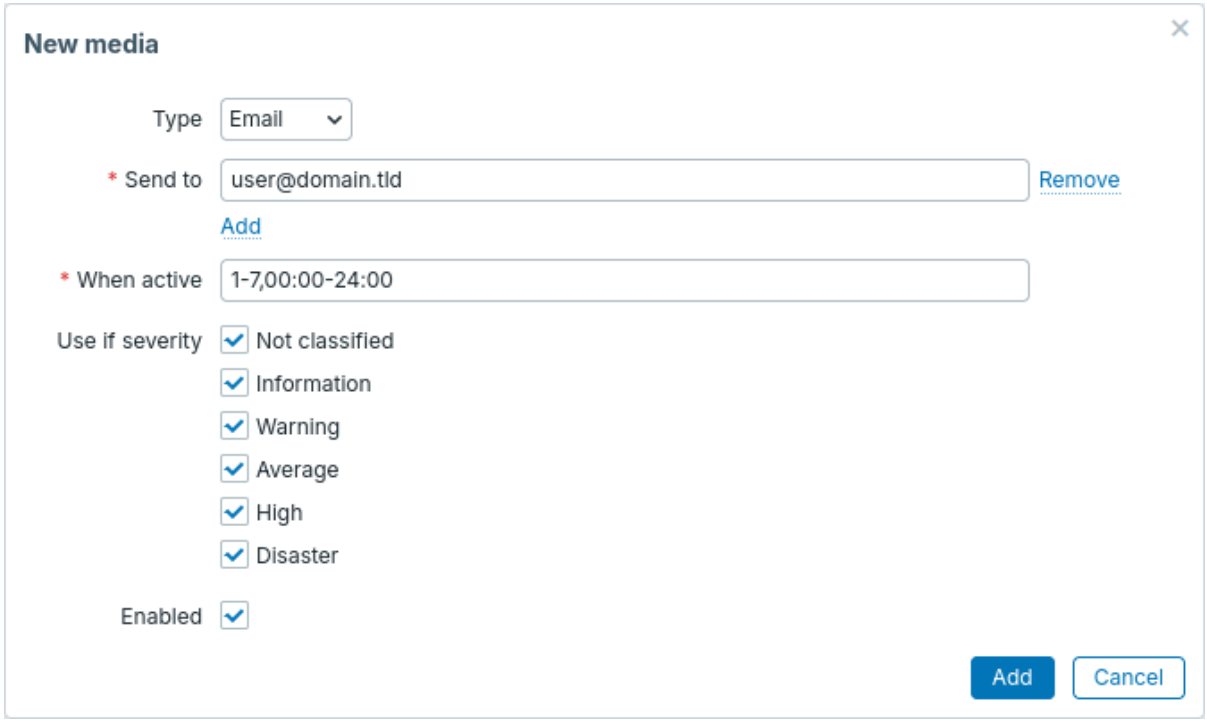
Todos los campos de entrada obligatorios están marcados con un asterisco rojo. Para obtener detalles sobre los campos de entrada de este formulario de configuración, consulte la página User settings.

De forma predeterminada, los usuarios nuevos no tienen ningún medio (métodos de entrega de notificaciones) definido. Para crear uno, vaya a la pestaña *Media* y haga clic en *Add*.



En la ventana emergente, introduzca la dirección de correo electrónico del usuario.
 Todos los tipos de medios están deshabilitados de forma predeterminada y se pueden habilitar en *Alerts* > *Media types*.

Puede especificar un período de tiempo durante el cual el medio estará activo (consulte la página Time period specification para obtener una descripción del formato). De forma predeterminada, un medio está siempre activo. También puede personalizar los niveles de trigger severity para los que el medio estará activo, pero por ahora deje todos habilitados.



Haga clic en *Add* para guardar el medio y, a continuación, vaya a la pestaña *Permissions*.

La pestaña Permissions tiene un campo obligatorio *Role*. El rol determina qué elementos de la frontend puede ver el usuario y qué acciones tiene □□□□□□ para realizar. Haga clic en *Select* y seleccione uno de los roles de la lista. Por ejemplo, seleccione *Admin role* para permitir el acceso a todas las secciones de la frontend de Zabbix, excepto *Administration*. Más adelante, puede modificar los permisos o crear más roles de usuario. Al seleccionar un rol, los permisos aparecerán en la misma pestaña:

Users

User | Media | Permissions

* Role: Admin role ✕ | Select

User type: Admin

Permissions

User group permissions

Group	Type	Permissions
All groups	Hosts	None
All groups	Templates	None

Proxies: None

Proxy groups: None

Permissions can be assigned for user groups only.

Access to UI elements

Dashboards: Dashboards

Monitoring: Problems | Hosts | Latest data | Maps | Discovery

APM: Traces | Metrics | Logs

Services: Services | SLA | SLA report

Inventory: Overview | Hosts

Reports: System information | Scheduled reports | Availability report | Top 100 triggers | Notifications

Data collection: Template groups | Host groups | Templates | Hosts | Maintenance | Discovery

Alerts: Trigger actions | Service actions | Discovery actions | Autoregistration actions | Internal actions

User profile settings

Redirect URL after login

Access to services

Read-write access to services: None

Read-only access to services: All

Access to modules

Action log | Clock | Discovery status | Favorite graphs | Favorite maps | Gauge | Geomap | Graph | Graph (classic) | Graph prototype | Honeycomb | Host availability | Host card | Host navigator | Item card | Item history | Item navigator | Item value | Map | Map navigation tree | Pie chart | Problem hosts | Problems | Problems by severity | Scatter plot | SLA report | System information | Top hosts | Top items | Top triggers | Trigger overview | URL | Web monitoring

Access to API

Disabled

Devices

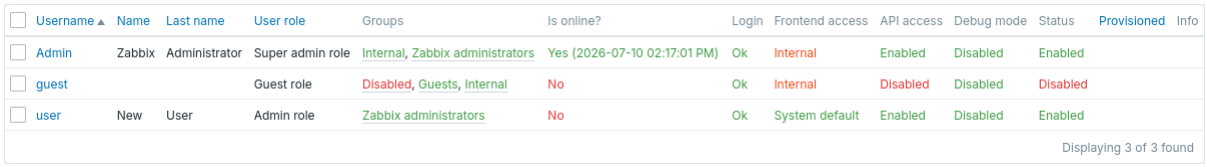
Enabled | Manage own devices

Access to actions

Create and edit dashboards | Create and edit maps | Create and edit maintenance | Add problem comments | Change severity | Acknowledge problems | Suppress problems | Close problems | Execute scripts | Manage API tokens | Manage scheduled reports | Manage SLA | Invoke "Execute now" on read-only hosts | Change problem ranking | Create and edit own media | Select "Server" for monitoring and discovery

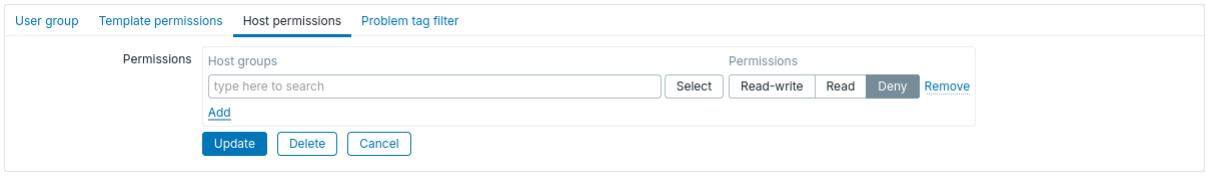
Add | Cancel

Haga clic en *Add* en el formulario de propiedades del usuario para guardar el usuario. El usuario nuevo aparecerá en la lista de usuarios.

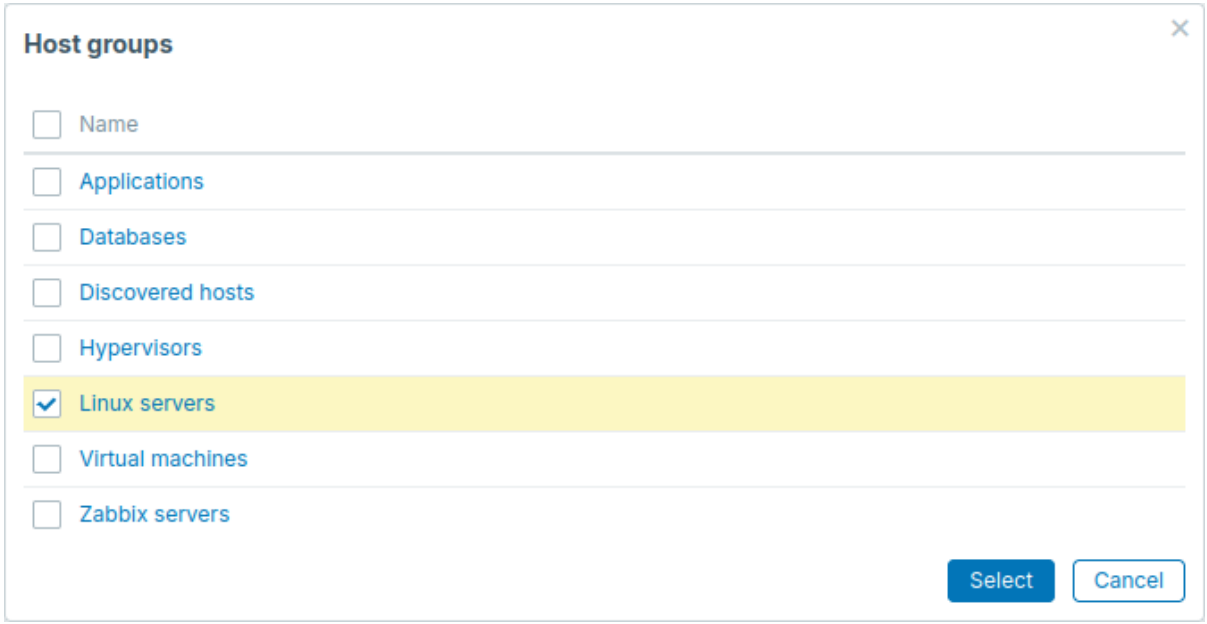


Adición de permisos

De forma predeterminada, un usuario nuevo no tiene permisos para acceder a hosts y templates. Para conceder derechos al usuario, haga clic en el grupo del usuario en la columna *Groups* (en este caso, *Zabbix administrators*). En el formulario de propiedades de *User groups*, vaya a la pestaña *Host permissions* para asignar permisos a grupos de hosts. Haga clic en Add para que se muestre el campo de selección del grupo de hosts:



Luego haga clic en *Select* junto al campo para ver la lista de grupos de hosts. Este usuario debe tener acceso de solo lectura al grupo *Linux servers*, así que marque la casilla correspondiente en la lista y haga clic en *Select* para confirmar su elección.



Haga clic en el botón *Read* para establecer el nivel de permiso y luego en *Update* para guardar los cambios realizados en la configuración del grupo de usuarios.

Para conceder permisos a templates, deberá cambiar a la pestaña *Template permissions* y especificar grupos de templates. Los pasos son idénticos a la asignación de permisos a grupos de hosts. Puede encontrar una descripción general de templates en la sección New template de este Quickstart.

Attention:
En Zabbix, los derechos de acceso a hosts y templates se asignan a user groups, no a usuarios individuales.

¡Listo! Puede intentar iniciar sesión usando las credenciales del nuevo usuario.

Nuevo host

Resumen

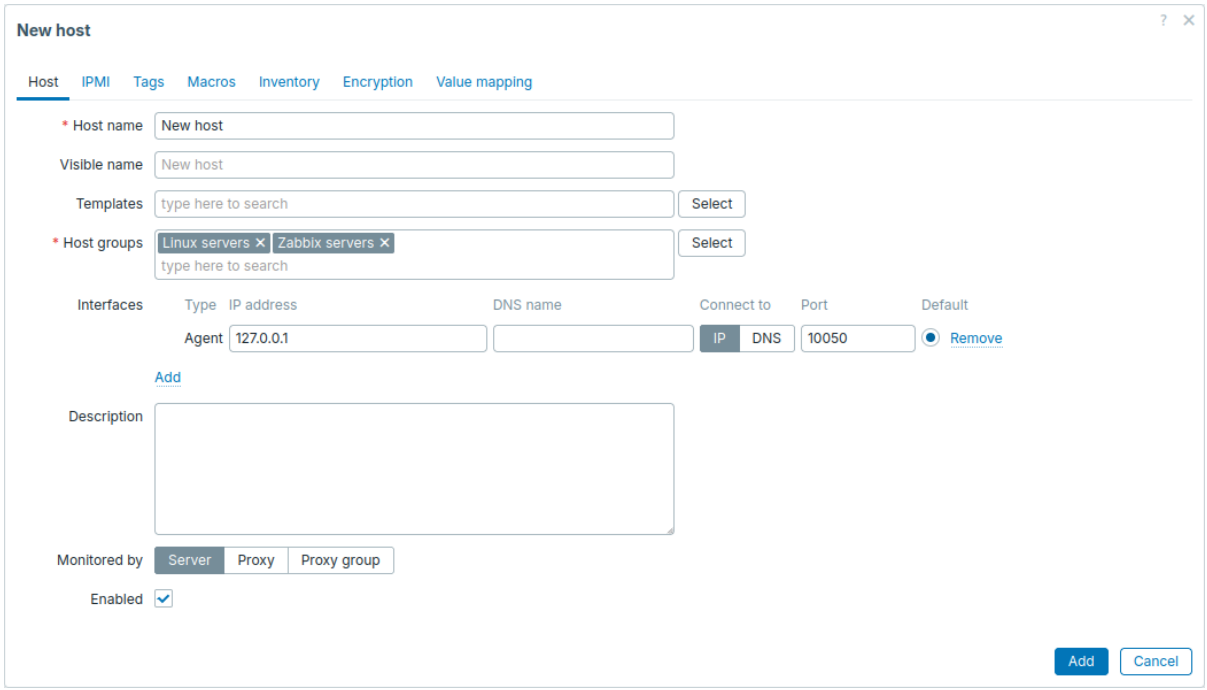
En esta sección aprenderá cómo configurar un nuevo host.

Un host en Zabbix es una entidad conectada a la red (física o virtual) que desea supervisar. La definición de lo que puede ser un ”host” en Zabbix es bastante flexible. Puede ser un servidor físico, un switch de red, una máquina virtual o alguna aplicación.

Añadir un host

La información sobre los hosts configurados en Zabbix está disponible en la sección de menú *Recopilación de datos* > *Hosts*, así como en la sección de menú *Monitorización* > *Hosts*. Ya existe un host predefinido llamado «Zabbix server», pero queremos aprender a añadir otro.

Para añadir un nuevo host, haga clic en *Crear host*. Se abrirá un formulario de configuración del host.



Todos los campos de entrada obligatorios están marcados con un asterisco rojo.

Lo mínimo que debe introducir aquí es:

Nombre del host

- Introduzca un nombre para el host. Se permiten caracteres alfanuméricos, espacios, puntos, guiones y guiones bajos.

Grupos de hosts

- Seleccione uno o varios grupos existentes haciendo clic en el botón *Seleccionar*, o introduzca el nombre de un grupo inexistente para crear un grupo nuevo.

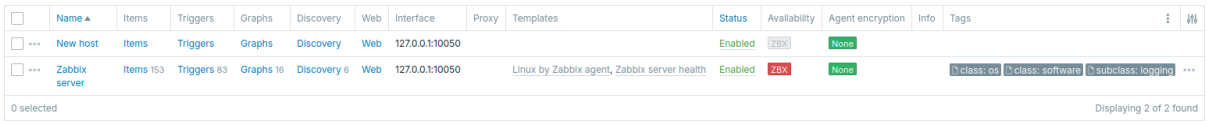
Note:
Todos los permisos de acceso se asignan a grupos de hosts, no a hosts individuales. Por eso, un host debe pertenecer al menos a un grupo.

Interfaces: dirección IP

- Aunque técnicamente no es un campo obligatorio, se necesita una interfaz de host para recopilar determinadas métricas. Para utilizar comprobaciones pasivas del agent de Zabbix, especifique en este campo la IP o el DNS del agent. Tenga en cuenta que también debe especificar la IP o el DNS del server de Zabbix en la directiva «Server» del archivo de configuración del agent de Zabbix. Si el agent y el server de Zabbix están instalados en la misma máquina, debe especificar la misma IP/DNS en ambos lugares.

Las demás opciones pueden dejarse sin cambios, utilizando sus valores predeterminados.

Cuando haya terminado, haga clic en *Añadir*. El nuevo host debería aparecer en la lista de hosts.



La columna *Disponibilidad* contiene indicadores de disponibilidad del host para cada interfaz. Cuando se define una interfaz del agent de Zabbix, podemos utilizar el icono de disponibilidad del agent (con la etiqueta *ZBX*) para determinar la disponibilidad del host:

- ZBX - no se ha establecido el estado del host; todavía no se ha realizado ninguna comprobación de métricas.
- ZBX - el host está disponible; la comprobación de métricas se ha realizado correctamente.

- ZBX - el host no está disponible; la comprobación de métricas ha fallado (mueva el cursor del ratón sobre el icono para ver el mensaje de error).

Puede haber algún error de comunicación, posiblemente causado por credenciales incorrectas de la interfaz. Compruebe que el server de Zabbix está en ejecución e intente actualizar la página más tarde.

Nuevo item

Resumen

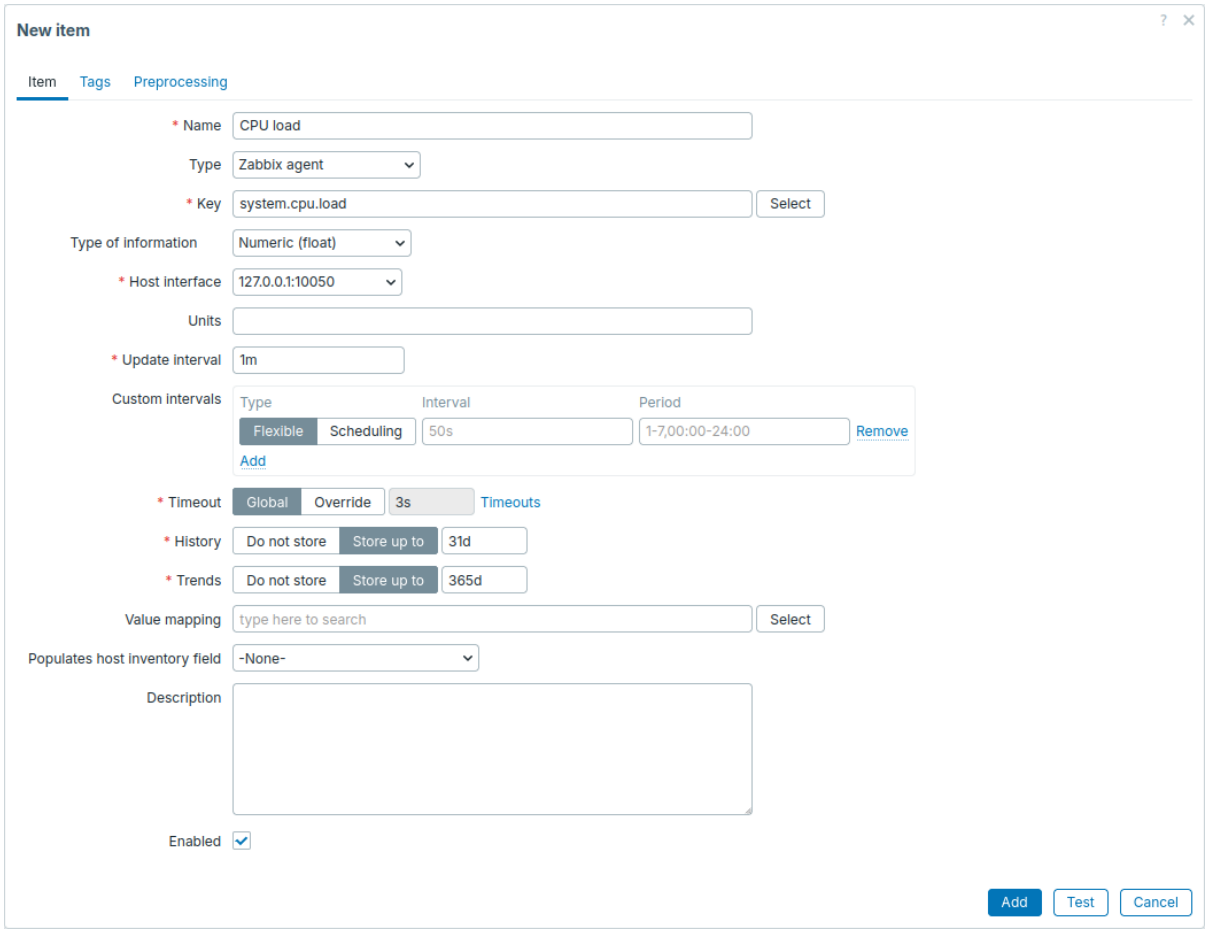
En esta sección, aprenderá cómo configurar un item.

Los items son la base para recopilar datos en Zabbix. Sin items, no hay datos, porque solo un item define una métrica única o qué tipo de datos recopilar de un host.

Agregar item

Todos los items se agrupan en torno a los hosts. Por eso, para configurar un item de ejemplo, vamos a *Recopilación de datos* > *Hosts* y buscamos el "New host" que hemos creado.

Haga clic en el enlace *Items* de la fila de "New host" y, a continuación, haga clic en *Crear item*. Se abrirá un formulario de configuración del item.



Todos los campos de entrada obligatorios están marcados con un asterisco rojo.

Para nuestro item de ejemplo, la información esencial que debe introducir es:

Name

- Introduzca *CPU load* como valor. Este será el nombre del item que se mostrará en las listas y en otros lugares.

Key

- Introduzca manualmente *system.cpu.load* como valor. Este es el nombre técnico de un item que identifica el tipo de información que se recopilará. La clave concreta es solo una de las claves predefinidas que incluye Zabbix agent.

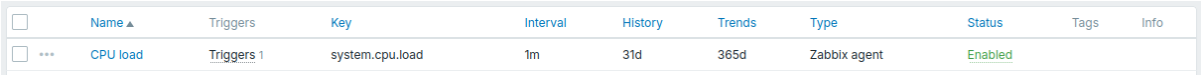
Type of information

- Este atributo define el formato de los datos esperados. Para la clave *system.cpu.load*, este campo se establecerá automáticamente en *Numeric (float)*.

Note:
También puede reducir el número de días durante los que se conservará el historial del item, a 7 o 14. Es una buena práctica para aliviar a la base de datos de tener que almacenar muchos valores históricos.

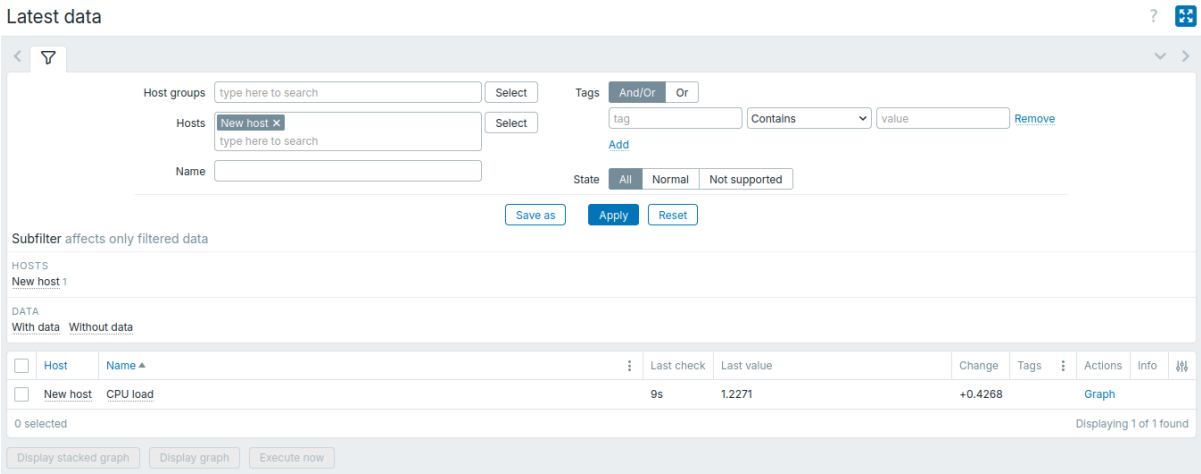
Otras opciones pueden dejarse sin cambios, usando sus valores predeterminados.

Cuando termine, haga clic en *Add*. El nuevo item debería aparecer en la lista de items.



Visualización de datos

Una vez definido un item, quizá te preguntes si realmente está recopilando datos. Para ello, ve a *Monitoring* > *Latest data*, selecciona 'New host' en el filtro y haz clic en *Apply*.



Dicho esto, pueden pasar hasta 60 segundos hasta que lleguen los primeros datos. Ese es, de forma predeterminada, el intervalo con el que el server lee los cambios de configuración y detecta nuevos items para ejecutar.

Si no ves ningún valor en la columna 'Change', quizá solo se haya recibido un valor hasta ahora. Espera 30 segundos para que llegue otro valor.

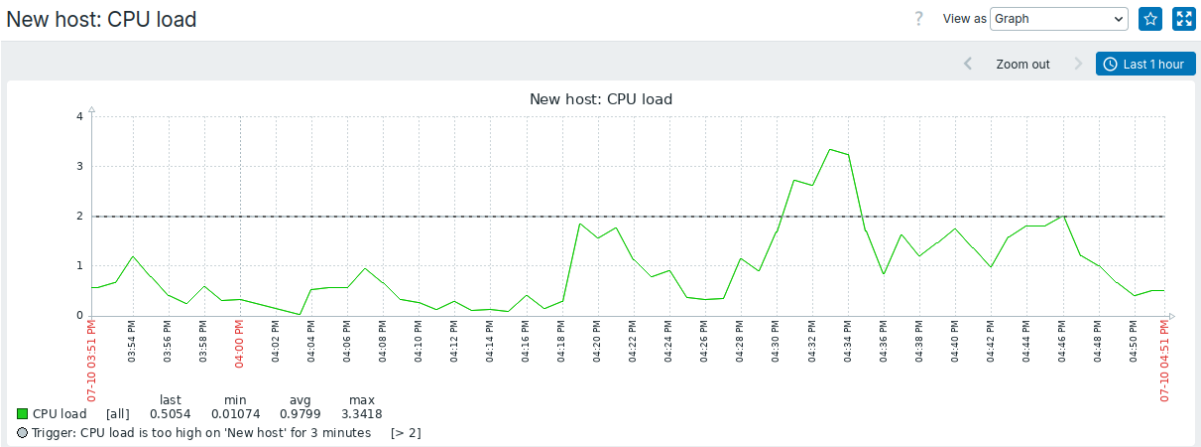
Si no ves información sobre el item como en la captura de pantalla, asegúrate de que:

- hayas rellenado los campos 'Key' y 'Type of information' del item exactamente como en la captura de pantalla;
- tanto el agent como el server estén en ejecución;
- el estado del host sea 'Monitored' y su icono de disponibilidad sea verde;
- el host seleccionado en el filtro de hosts sea el correcto;
- el item esté habilitado.

Gráficos

Una vez que el item haya estado funcionando durante un tiempo, puede ser un buen momento para ver algo visual. Los gráficos simples están disponibles para cualquier item numérico monitorizado sin ninguna configuración adicional. Estos gráficos se generan en tiempo de ejecución.

Para ver el gráfico, vaya a *Monitoring* > *Latest data* y haga clic en el enlace *Graph* situado junto al item.



Nuevo trigger

Resumen

En esta sección aprenderá cómo configurar un trigger.

Los items solo recopilan datos. Para evaluar automáticamente los datos entrantes, necesitamos definir triggers. Un trigger contiene una expresión que define un umbral de lo que se considera un nivel aceptable para los datos.

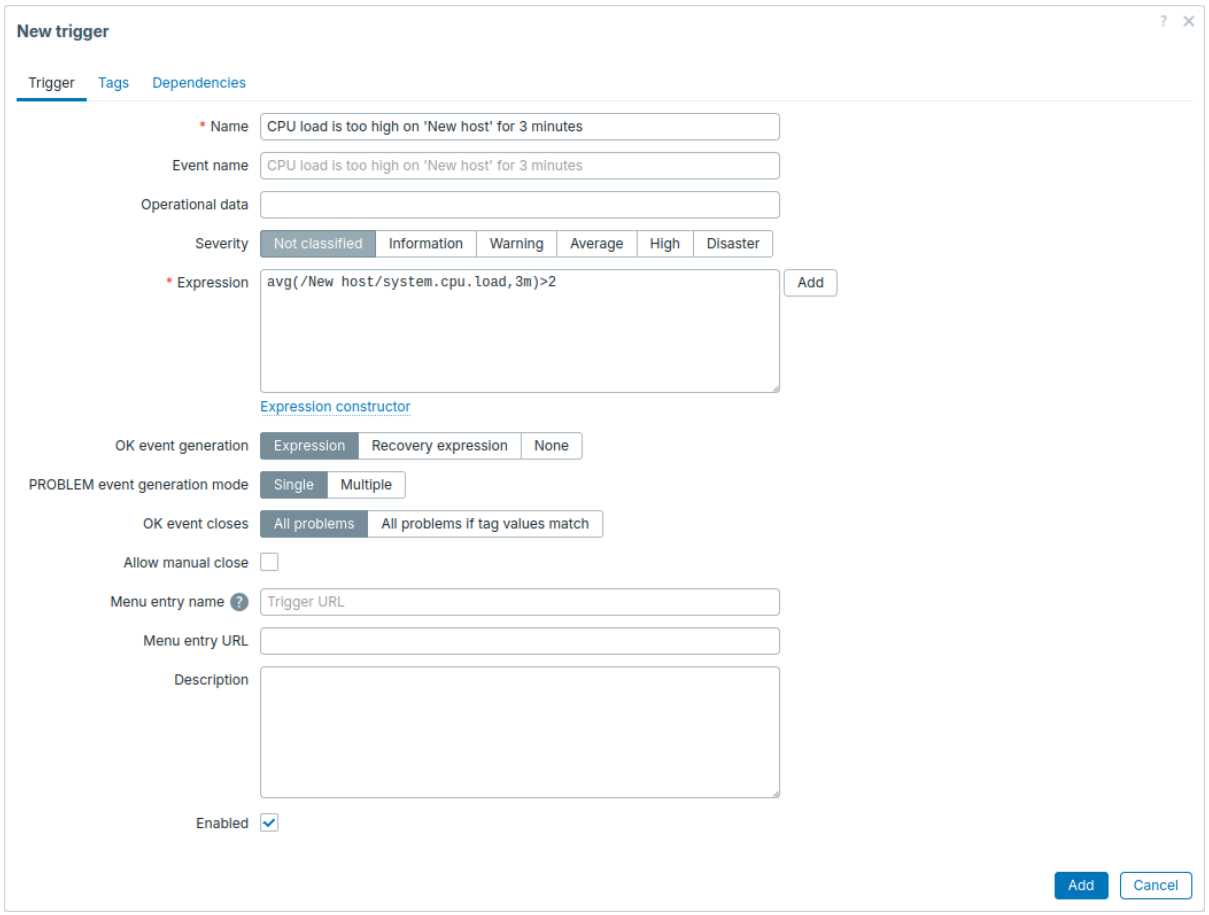
Si los datos entrantes superan ese nivel, un trigger se ”disparará” o pasará a un estado de ’Problem’, lo que indica que ha ocurrido algo que puede requerir atención. Si el nivel vuelve a ser aceptable, el trigger regresa a un estado de ’Ok’.

Añadir trigger

Para configurar un trigger para nuestro item:

1. Vaya a *Data collection* > *Hosts*.
2. Haga clic en el enlace *Triggers* de la fila de ’New host’.
3. Haga clic en *Create trigger*.

Se abre un formulario de configuración de trigger.



Para nuestro trigger, la información esencial que debe introducir aquí es:

Name

- Introduzca *CPU load is too high on ’New host’ for 3 minutes* como valor. Este será el nombre del trigger que se mostrará en listas y en otros lugares.

Expression

- Introduzca: `avg(/New host/system.cpu.load,3m)>2`

Esta es la expresión del trigger. Asegúrese de introducir la expresión correctamente, hasta el último símbolo. La clave del item aquí (`system.cpu.load`) se usa para hacer referencia al item. Esta expresión en particular básicamente indica que se supera el umbral de problema cuando el valor promedio de la carga de CPU durante 3 minutos es superior a 2. Puede obtener más información sobre la sintaxis de las expresiones de trigger.

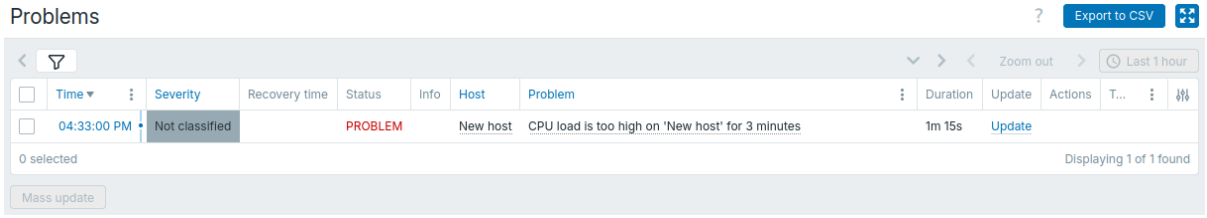
Other options pueden dejarse sin cambios, usando sus valores predeterminados.

Cuando termine, haga clic en *Add*. El nuevo trigger debería aparecer en la lista de triggers.

Mostrando el estado del trigger

Con un trigger definido, puede que le interese ver su estado.

Si la carga de CPU ha superado el nivel de umbral que definió en el trigger, el problema se mostrará en *Monitoring* > *Problems*.



El parpadeo en la columna de estado indica un cambio reciente del estado del trigger, uno que ha tenido lugar en los últimos 30 minutos.

Recepción de notificación de problema

Resumen

En esta sección aprenderá cómo configurar alertas en forma de notificaciones en Zabbix.

Con los items recopilando datos y los triggers diseñados para ”activarse” ante situaciones problemáticas, también sería útil contar con algún mecanismo de alertas para notificar eventos importantes incluso cuando no se esté revisando activamente el frontend de Zabbix.

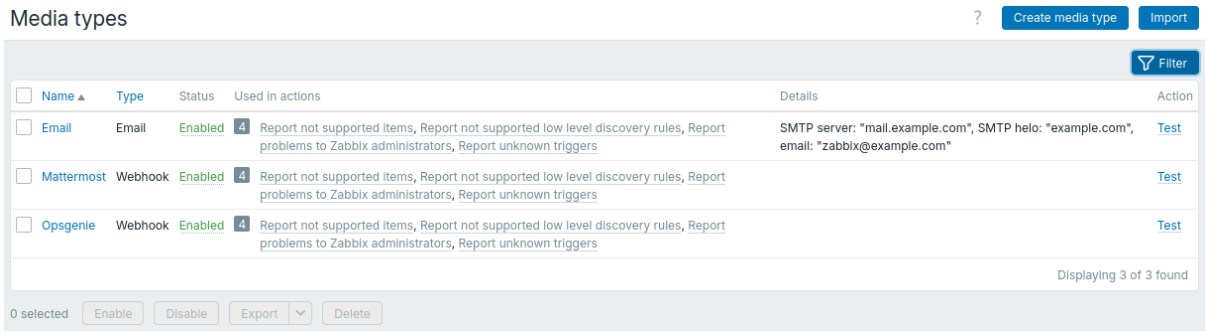
Eso es lo que hacen las notificaciones.

Dado que el correo electrónico es el método de entrega más popular para las notificaciones de problemas, aprenderemos cómo configurar una notificación por correo electrónico.

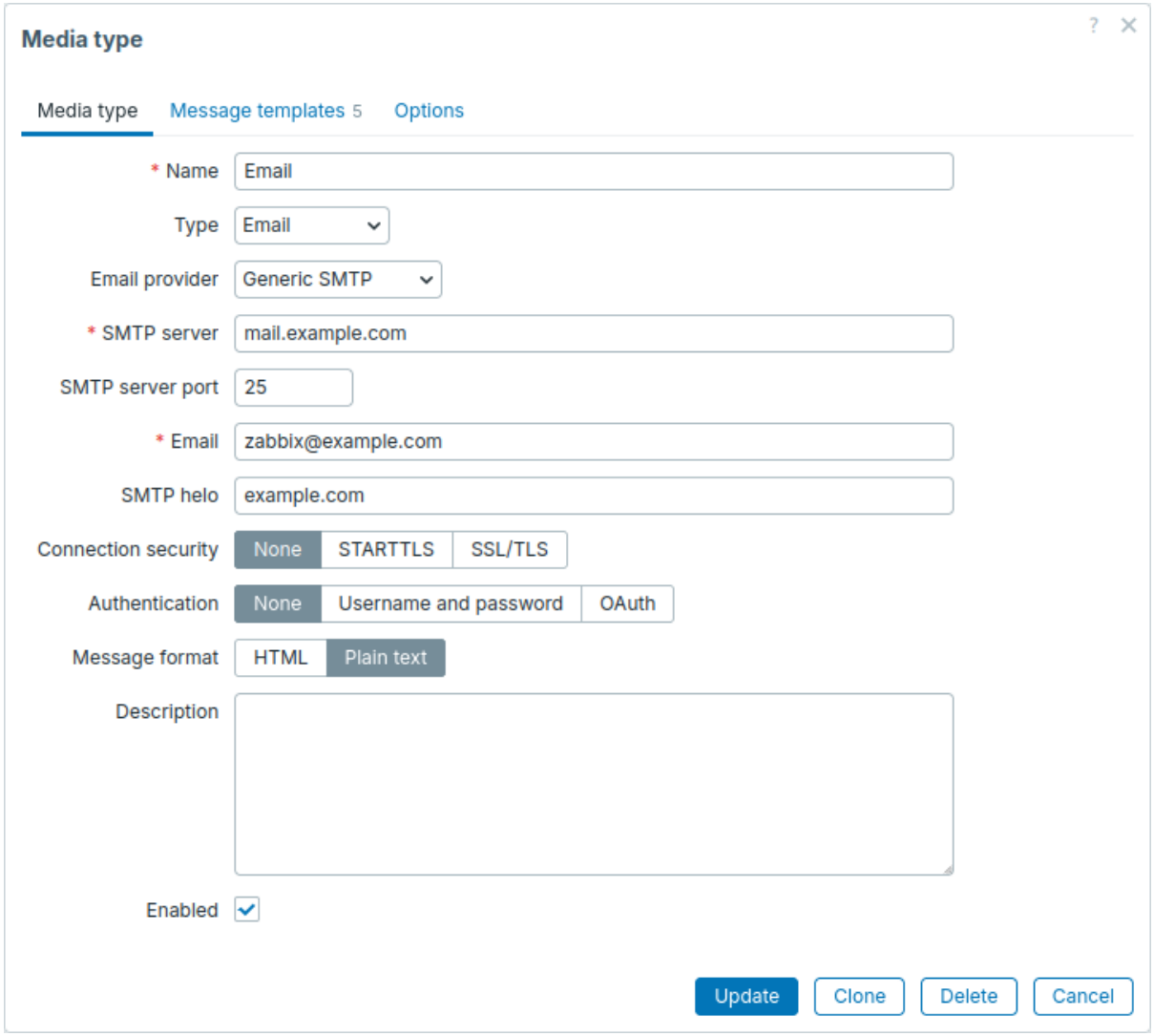
Configuración de correo electrónico

Inicialmente, hay varios métodos de entrega de notificaciones predefinidos en Zabbix. Email es uno de ellos.

Para configurar los ajustes de correo electrónico, vaya a *Alerts* > *Media types* y haga clic en *Email* en la lista de tipos de medios predefinidos.



Se abre un formulario de configuración de ajustes de correo electrónico.



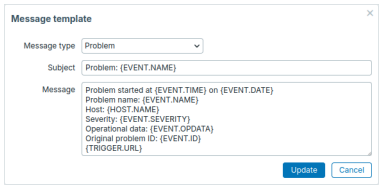
Todos los campos obligatorios están marcados con un asterisco rojo.

En la pestaña *Media type*, establezca los valores de *SMTP server*, *SMTP helo* y *Email* según corresponda a su entorno.

Note:
El valor del campo *Email* se utilizará como dirección 'From' para las notificaciones enviadas desde Zabbix.

A continuación, es necesario definir el contenido del mensaje de problema. El contenido se define mediante una plantilla de mensaje, configurada en la pestaña *Message templates*.

Haga clic en *Add* para crear una plantilla de mensaje y seleccione *Problem* como tipo de mensaje.

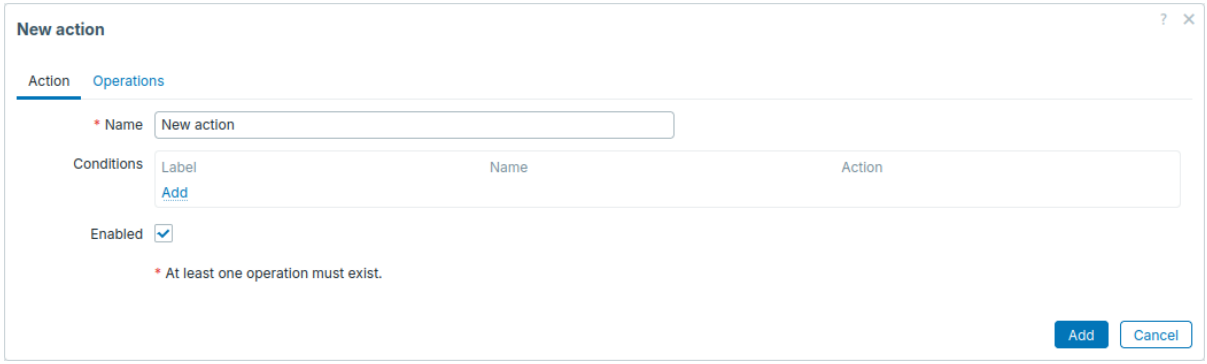


Haga clic en *Add* cuando esté listo y guarde el formulario.

Ahora ha configurado *Email* como un tipo de medio funcional. El tipo de medio también debe vincularse a los usuarios definiendo direcciones de entrega específicas (como hicimos al configurar un nuevo usuario); de lo contrario, no se utilizará.

Nueva acción

La entrega de notificaciones es una de las cosas que hacen las actions en Zabbix. Por lo tanto, para configurar una notificación, vaya a *Alerts* > *Actions* > *Trigger actions* y haga clic en *Create action*.

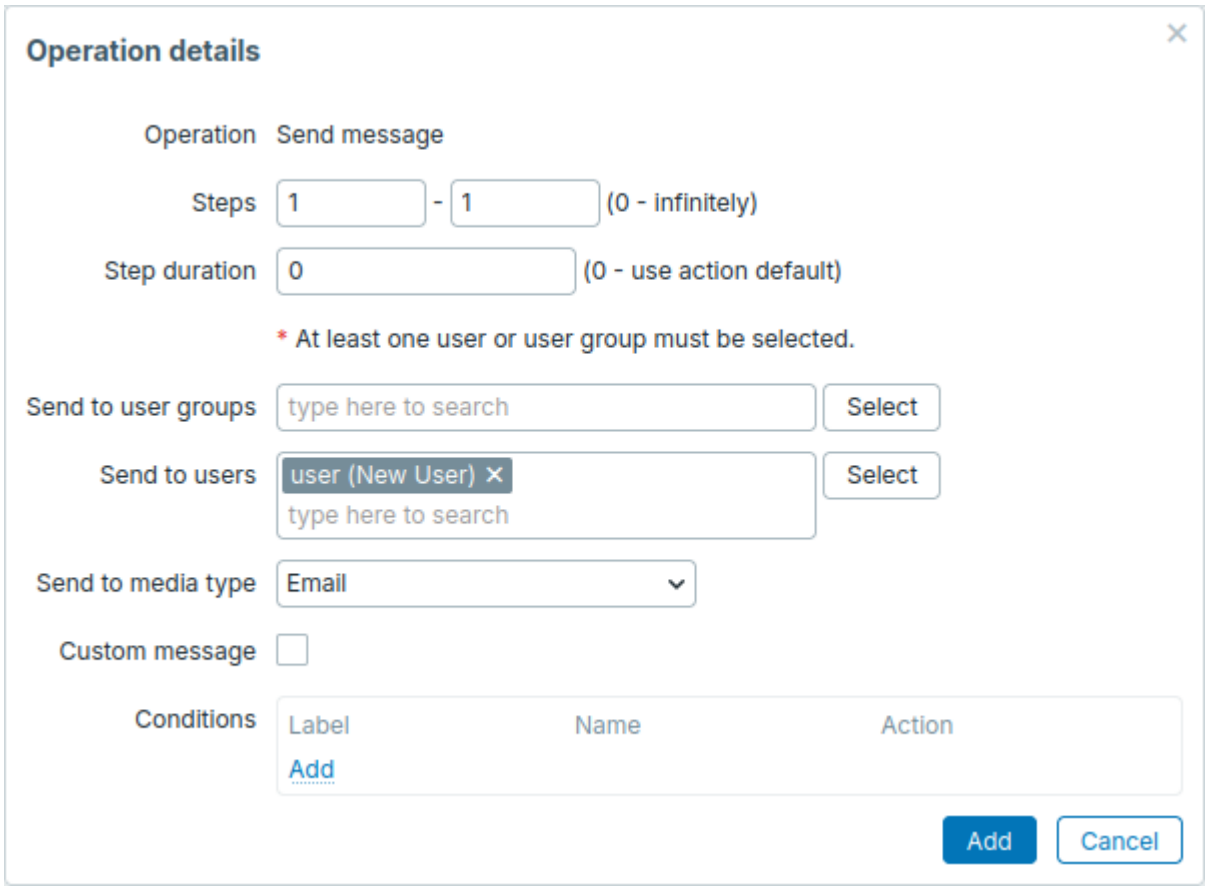


Todos los campos obligatorios están marcados con un asterisco rojo.

En este formulario, introduzca un nombre para la action.

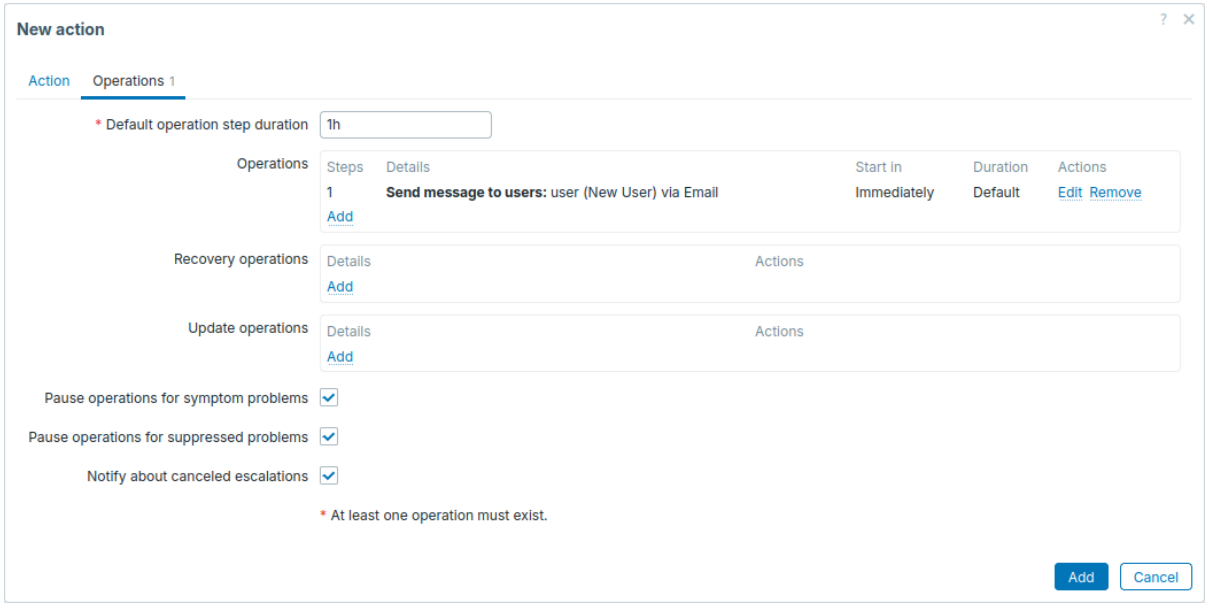
En el caso más simple, si no añadimos ninguna condición más específica, la action se ejecutará ante cualquier cambio de trigger de 'Ok' a 'Problem'.

Aun así, debemos definir qué debe hacer la action, y eso se configura en la pestaña *Operations*. Haga clic en *Add* en el bloque *Operations*, lo que abrirá un nuevo formulario de operación.



Todos los campos obligatorios están marcados con un asterisco rojo.

Aquí, haga clic en *Select* en el bloque *Send to Users* y seleccione el usuario ('user') que hemos definido. Seleccione "Email" como valor de *Send to media type*. Cuando termine, haga clic en *Add*, y la operación debería añadirse:



Eso es todo para una configuración simple de action, así que haga clic en *Add* en el formulario de action.

Recepción de notificaciones

Ahora que la entrega de notificaciones está configurada, sería interesante recibir una de verdad. Para ayudar con eso, podemos aumentar a propósito la carga en nuestro host, de modo que nuestro trigger se "dispare" y recibamos una notificación de problema.

Abra la consola en su host y ejecute:

```
cat /dev/urandom | md5sum
```

Puede ejecutar uno o varios de estos procesos.

Ahora vaya a *Monitoring* > *Latest data* y observe cómo han aumentado los valores de 'CPU Load'. Recuerde que, para que nuestro trigger se "dispare", el valor de 'CPU Load' debe superar '2' durante 3 minutos seguidos. Una vez que ocurra:

- en *Monitoring* > *Problems* debería ver el trigger con un estado `Problem` parpadeante;
- debería recibir una notificación de problema en su correo electrónico.

Attention:
Si las notificaciones no funcionan:

- verifique una vez más que tanto la configuración del correo electrónico como la acción se hayan configurado correctamente
- asegúrese de que el usuario que creó tenga al menos permisos de lectura en el host que generó el evento, como se indica en el paso *Adding user*. El usuario, al formar parte del grupo de usuarios 'Zabbix administrators', debe tener al menos acceso de lectura al grupo de hosts 'Linux servers' al que pertenece nuestro host.
- Además, puede consultar el registro de acciones yendo a *Reports* > *Action log*.

Nueva template

Descripción general

En esta sección aprenderá a configurar un template.

Anteriormente aprendimos a configurar un item y un trigger, así como a obtener una notificación de problema para el host.

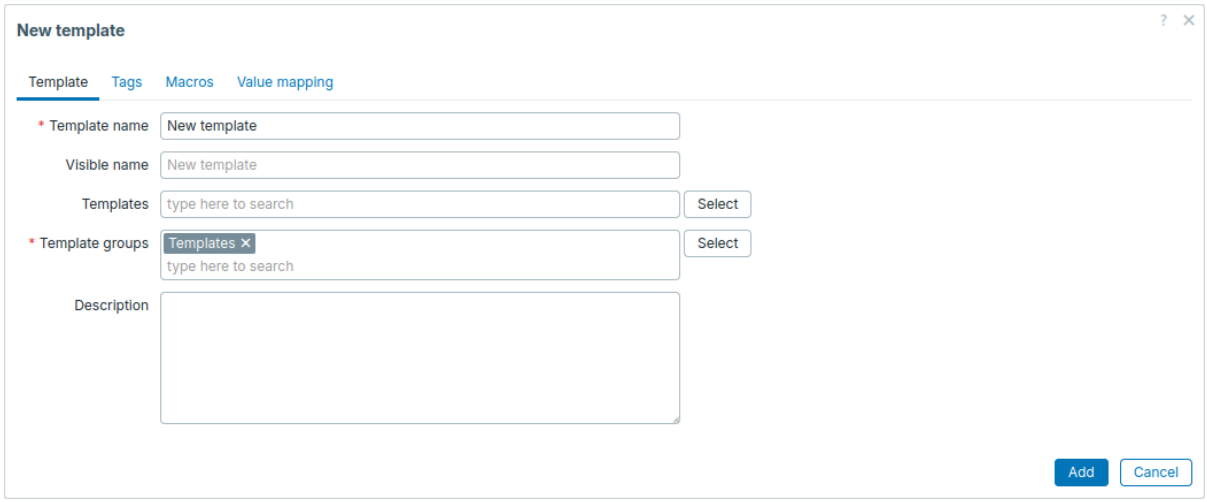
Aunque todos estos pasos ofrecen mucha flexibilidad por sí mismos, puede parecer que son demasiados pasos si es necesario realizarlos, por ejemplo, para mil hosts. Sería útil contar con algún tipo de automatización.

Aquí es donde los templates resultan útiles.
Los templates permiten agrupar items, triggers y otras entidades útiles para reutilizarlos una y otra vez al aplicarlos a los hosts en un solo paso.

Cuando un template se vincula a un host, este hereda todas las entidades del template. Por lo tanto, básicamente se puede aplicar muy rápidamente un conjunto de comprobaciones preparado de antemano.

Añadir template

Para empezar a trabajar con templates, primero debemos crear uno. Para ello, en *Data collection* > *Templates* haga clic en *Create template*. Esto abrirá un formulario de configuración de template.



Todos los campos de entrada obligatorios están marcados con un asterisco rojo.

Los parámetros requeridos que debe introducir aquí son:

Template name

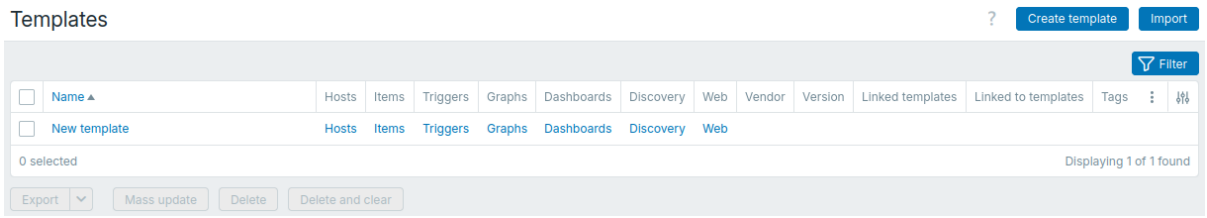
- Introduzca un nombre de template. Se permiten caracteres alfanuméricos, espacios y guiones bajos.

Template groups

- Seleccione uno o varios grupos haciendo clic en el botón *Select*. El template debe pertenecer a un grupo.

Note:
Los permisos de acceso a los grupos de templates se asignan en la configuración de user group en la pestaña **Template permissions** de la misma manera que los permisos de host. Todos los permisos de acceso se asignan a grupos, no a templates individuales, por eso es obligatorio incluir el template en al menos un grupo.

Cuando termine, haga clic en *Add*. Su nuevo template debería ser visible en la lista de templates. También puede usar el filter para encontrar su template.

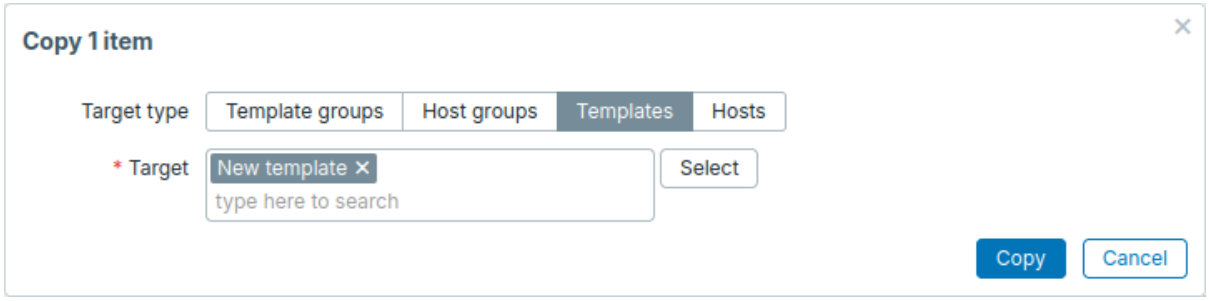


Como puede ver, el template está ahí, pero no contiene nada: no tiene items, triggers ni otras entidades.

Añadir un item a un template

Para añadir un item al template:

1. Vaya a *Recopilación de datos* > *Hosts*.
2. Localice «New host» en la lista y haga clic en el hipervínculo *Items*.
3. Marque la casilla del item «CPU Load» en la lista.
4. Haga clic en *Copy* debajo de la lista.
5. Seleccione la pestaña *Templates*.
6. Seleccione el template al que desea copiar el item.



Todos los campos de entrada obligatorios están marcados con un asterisco rojo. 7. Haga clic en *Copy*.

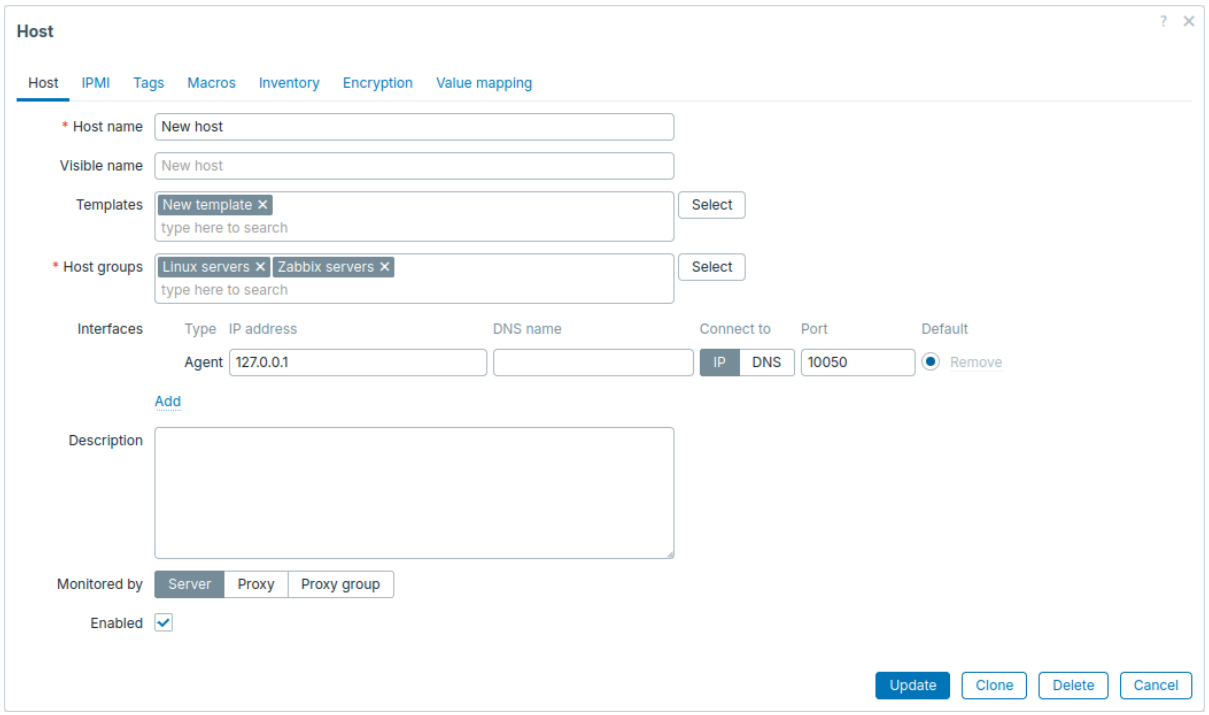
Si ahora va a *Recopilación de datos* > *Templates*, «New template» debería contener un item nuevo.

Por ahora nos detendremos en un solo item, pero de forma similar puede añadir cualquier otro item, trigger u otra entidad al template hasta obtener un conjunto de entidades bastante completo para un propósito determinado (supervisión del sistema operativo o de una aplicación específica).

Vinculación de un template a un host

Con el template listo, solo queda añadirlo a un host. Para ello:

1. Vaya a *Recopilación de datos* > *Hosts*.
2. Haga clic en «Nuevo host» para abrir su formulario de configuración y busque el campo *Templates*.
3. Empiece a escribir *New template* en el campo *Templates*. El nombre del template recién creado debería aparecer en la lista desplegable.
4. Desplácese hacia abajo para seleccionarlo. Compruebe que aparece en el campo *Templates*.



5. Haga clic en *Actualizar* en el formulario para guardar los cambios.

El template ya se ha añadido al host, junto con todas las entidades que contiene.

De esta forma, también se puede aplicar a cualquier otro host. Cualquier cambio realizado en los items, triggers y otras entidades en el nivel del template se propagará a los hosts a los que esté vinculado el template.

Vinculación de templates predefinidos a hosts

Como habrá observado, Zabbix incluye un conjunto de templates predefinidos para distintos sistemas operativos, dispositivos y aplicaciones. Para empezar a supervisar muy rápidamente, puede vincular el adecuado a un host, pero tenga en cuenta que estos templates deben ajustarse a su entorno. Es posible que algunas comprobaciones no sean necesarias y que los intervalos de sondeo sean demasiado frecuentes.

Hay más información disponible sobre templates.

Supervisión del Apache web server

Para comenzar a supervisar el Apache web server, siga este proceso:

1. Asegúrese de tener acceso a las estadísticas de Apache.
2. Aplique el template de supervisión de Apache.
3. Consulte los datos más recientes.

Para usar esta guía, debe tener Zabbix server y la web interface instalados. Debe haber iniciado sesión en Zabbix como el usuario predeterminado *Admin*.

Acceso a las estadísticas de Apache

El objetivo de este paso es asegurarse de que las estadísticas del servidor web Apache puedan ser leídas por Zabbix.

En Apache, la información en tiempo real sobre el rendimiento y la actividad del servidor la proporciona el módulo integrado `mod_status`. Cuando está habilitado, crea una página accesible desde la web (normalmente en `/server-status`) que muestra datos de rendimiento y actividad.

Para verificar que `mod_status` está habilitado para el servidor web Apache, ejecute:

En Debian/Ubuntu	En sistemas basados en RHEL
`apache2ctl -M | grep status`	`httpd -M | grep status`

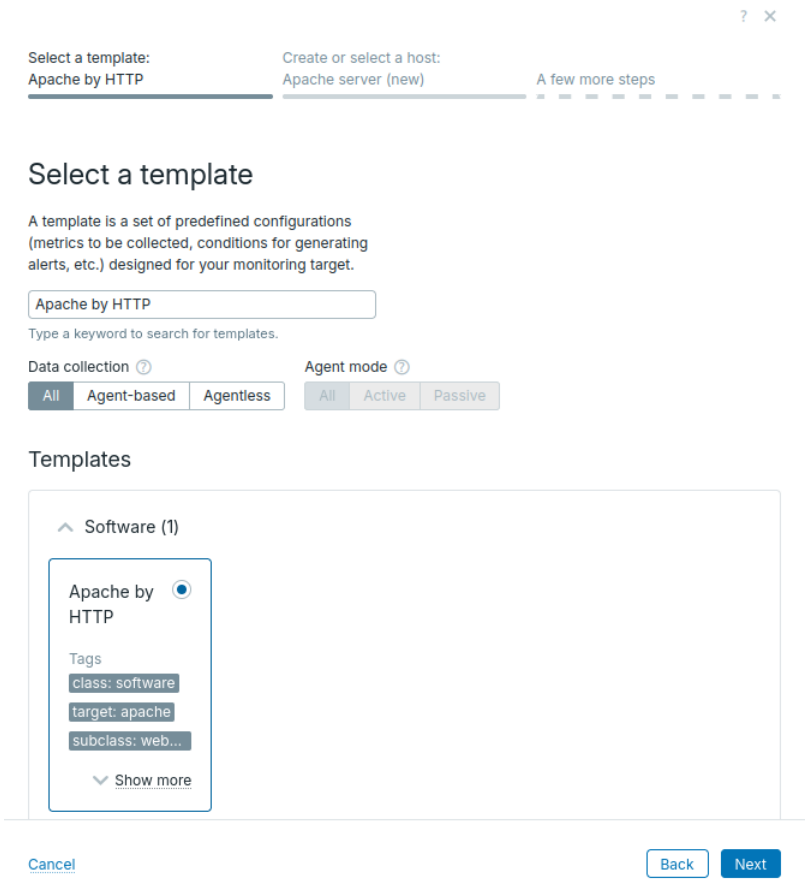
Si ve "`status_module (shared)`" en la salida, entonces el módulo de estado ya está habilitado. Si no, deberá habilitar el módulo de estado.

Aplicar la template de monitoreo de Apache

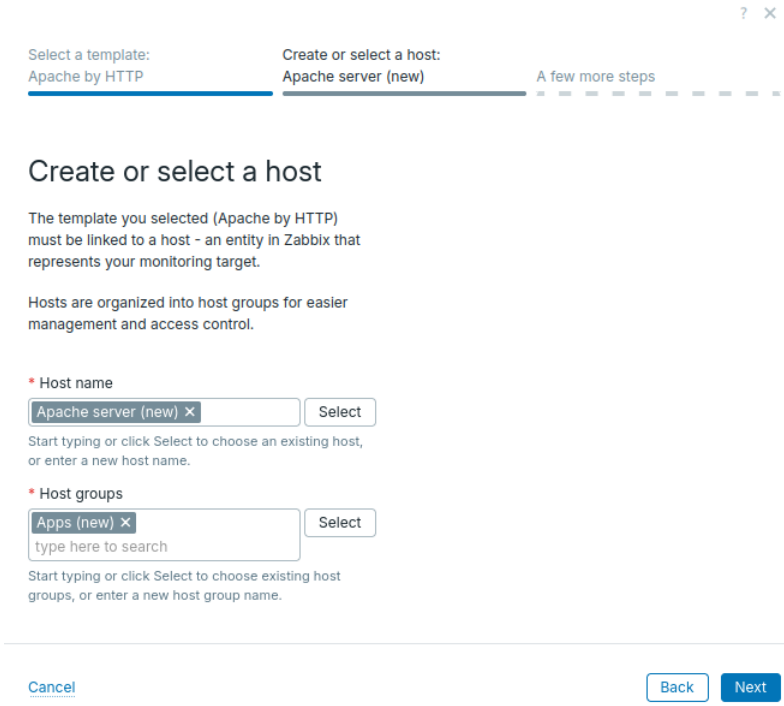
En Zabbix, esta template está lista para usar. Contiene elementos preconfigurados listos para la recopilación de datos, la visualización y el análisis.

Para aplicar la template:

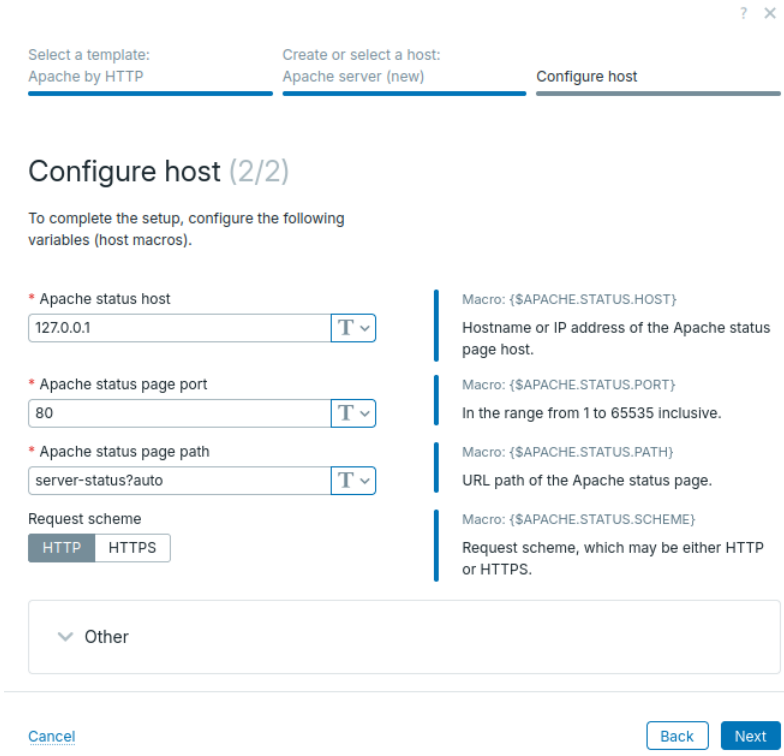
1. Inicie el Host Wizard en *Data collection* > *Hosts*.
2. Seleccione la template *Apache by HTTP*.



3. Cree un host de Apache y añádale un grupo. Este host virtual contendrá datos relacionados con Apache.



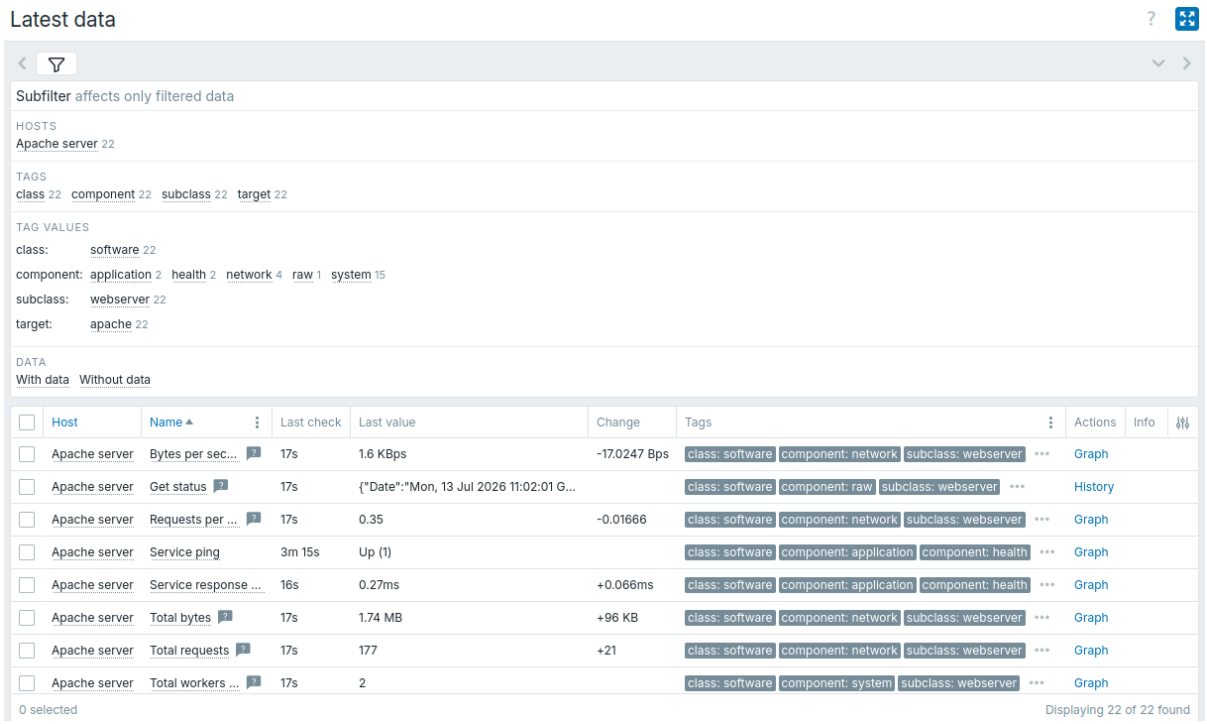
4. Configure el host. Es importante que introduzca la dirección del servidor web de Apache (127.0.0.1 en una máquina local, o la dirección IP en una máquina remota).



Puede dejar el resto de los ajustes con sus valores predeterminados.

5. Para completar el Host Wizard, haga clic en *Create* y *Finish*.

Ver los datos más recientes



¡Enhorabuena, la monitorización del servidor web Apache ha comenzado!

Zabbix ofrece opciones de visualización y envía alertas si surge algún problema. Para configurar alertas por correo electrónico, consulte Receiving problem notification.

Supervisión de un servidor Linux

Para empezar a supervisar un servidor Linux, siga este proceso:

1. Aplique la template de supervisión de Linux.
2. Instale Zabbix agent en el server.
3. Consulte los últimos datos.

Para empezar a supervisar un servidor Linux no local, necesita conocer su dirección IP y la dirección IP del server de Zabbix.

Se utilizará un Zabbix agent en el servidor Linux para supervisarlo. Las instrucciones para instalar el agent se proporcionarán durante el proceso.

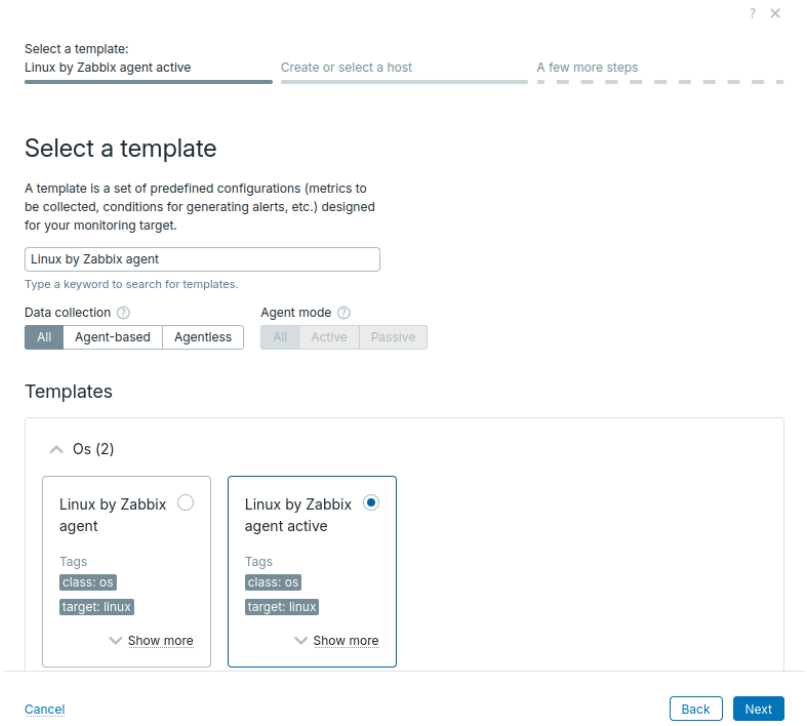
Para usar esta guía, debe tener Zabbix server y la interfaz web instalados. Debe haber iniciado sesión en Zabbix como el usuario predeterminado *Admin*.

Aplicar template de monitorización de Linux

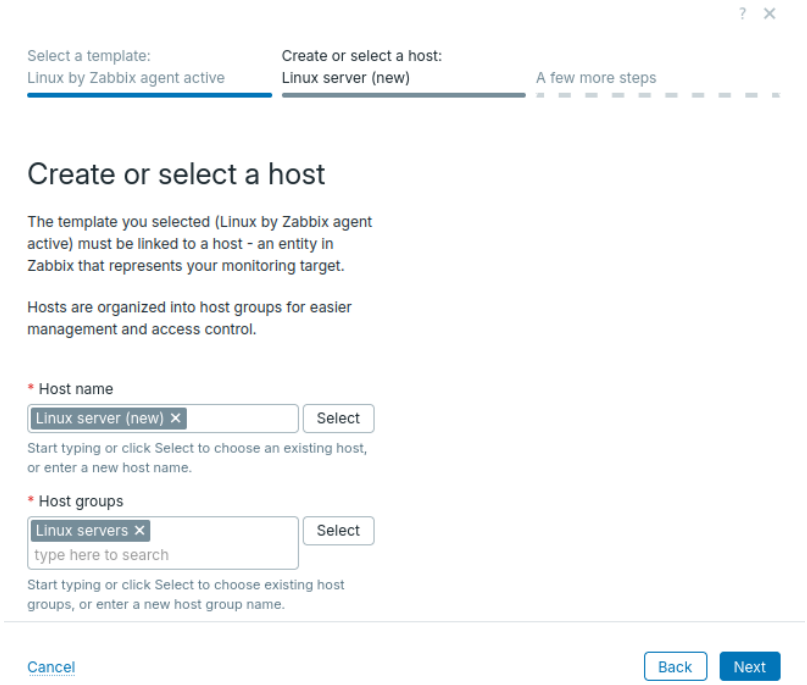
En Zabbix, este template está listo para usar. Contiene elementos preconfigurados listos para la recopilación de datos, la visualización y el análisis.

Para aplicar el template:

1. Inicie el Asistente de host en *Recopilación de datos* > *Hosts*.
2. Seleccione el template *Linux by Zabbix agent active*.



3. Cree un host para el servidor Linux y añádale un grupo. Este host virtual contendrá todos los datos relacionados con la monitorización del servidor Linux.



4. Configure e instale el agent.

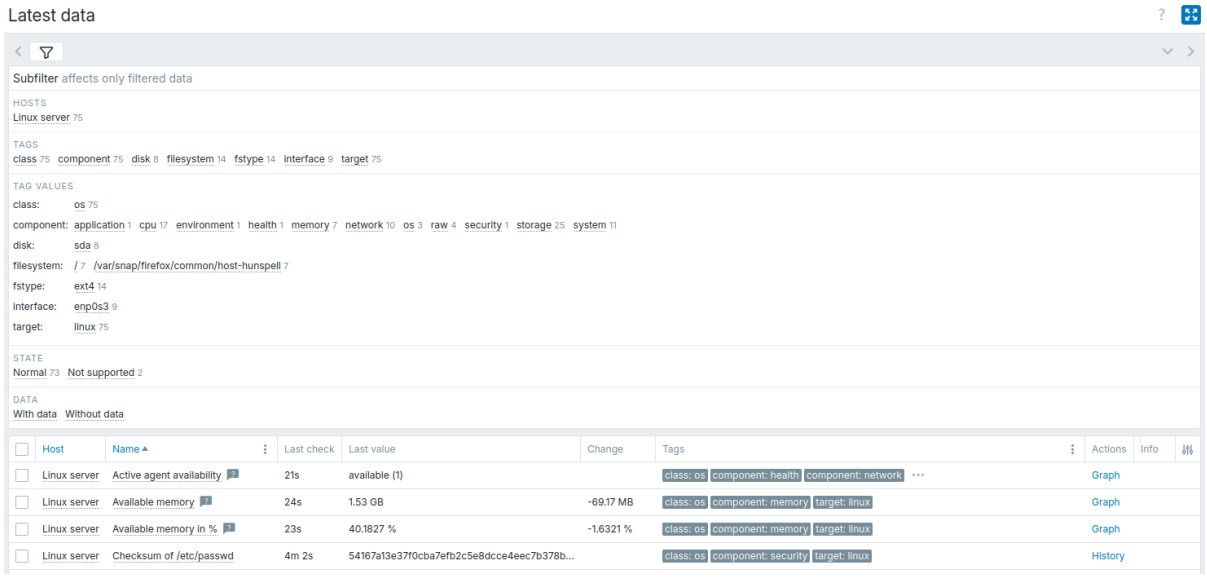


Para instalar el agent, ejecute el comando proporcionado en el servidor Linux. Esto instalará el agent y configurará la conexión del agent con el server de Zabbix mediante cifrado.
 Si ya tiene el agent instalado, puede omitir este paso (haga clic en *Siguiente*). Sin embargo, se le pedirá que proporcione la identidad de la clave precompartida y la clave precompartida para conectarse al agent.

Puede dejar el resto de los ajustes con sus valores predeterminados.

5. Finalice la configuración del host: haga clic en *Crear* y *Finalizar*.

Ver los datos más recientes



¡Enhorabuena, la supervisión del server Linux ha comenzado!

Zabbix ofrece opciones de visualización y envía alertas si surge algún problema. Para configurar alertas por correo electrónico, consulte Receiving problem notification.

Supervisión del servidor MySQL

Para empezar a supervisar un servidor MySQL, siga este proceso:

1. Cree un usuario MySQL con permisos limitados.
2. Aplique la template de supervisión de MySQL.
3. Consulte los datos más recientes.

Para empezar a supervisar un servidor MySQL, necesita conocer su DSN y la dirección IP del server de Zabbix.

Se utilizará un agent de Zabbix para supervisar el servidor MySQL. Las instrucciones para instalar el agent se proporcionarán durante el proceso.

Para usar esta guía, debe tener instalados Zabbix server y la interfaz web instalados. Debe haber iniciado sesión en Zabbix como el usuario predeterminado *Admin*.

Crear un usuario de MySQL con permisos limitados

Se recomienda crear un usuario de MySQL con permisos limitados únicamente a los necesarios para supervisar MySQL. Reemplace `<password>` por una contraseña segura y única.

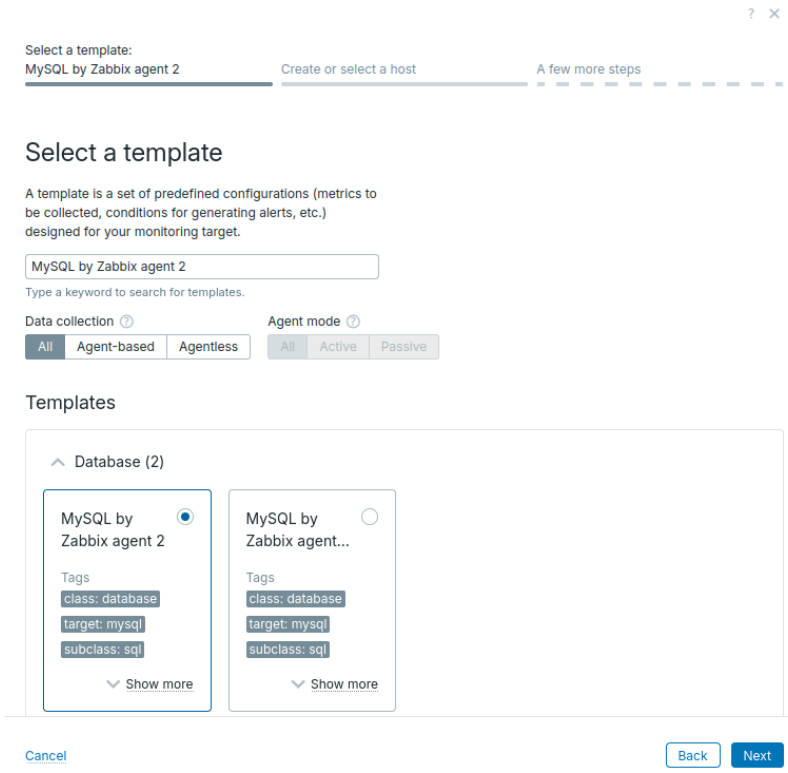
```
mysql> CREATE USER 'zbx_monitor'@'%' IDENTIFIED BY '<password>';
mysql> GRANT REPLICATION CLIENT,PROCESS,SHOW DATABASES,SHOW VIEW ON *.* TO 'zbx_monitor'@'%';
```

Aplicar template de monitorización de MySQL

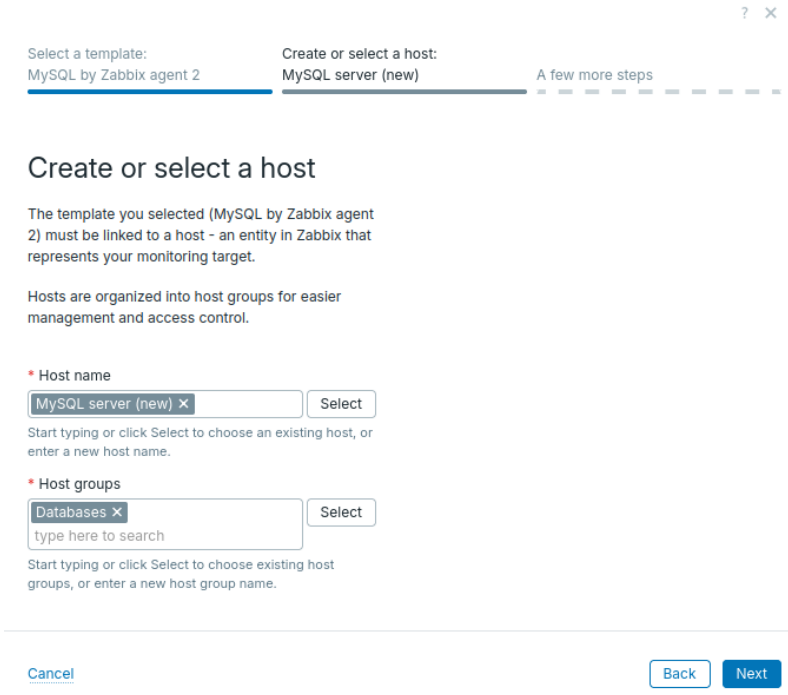
En Zabbix, este template está listo para usar. Contiene elementos preconfigurados listos para la recopilación de datos, la visualización y el análisis.

Para aplicar el template:

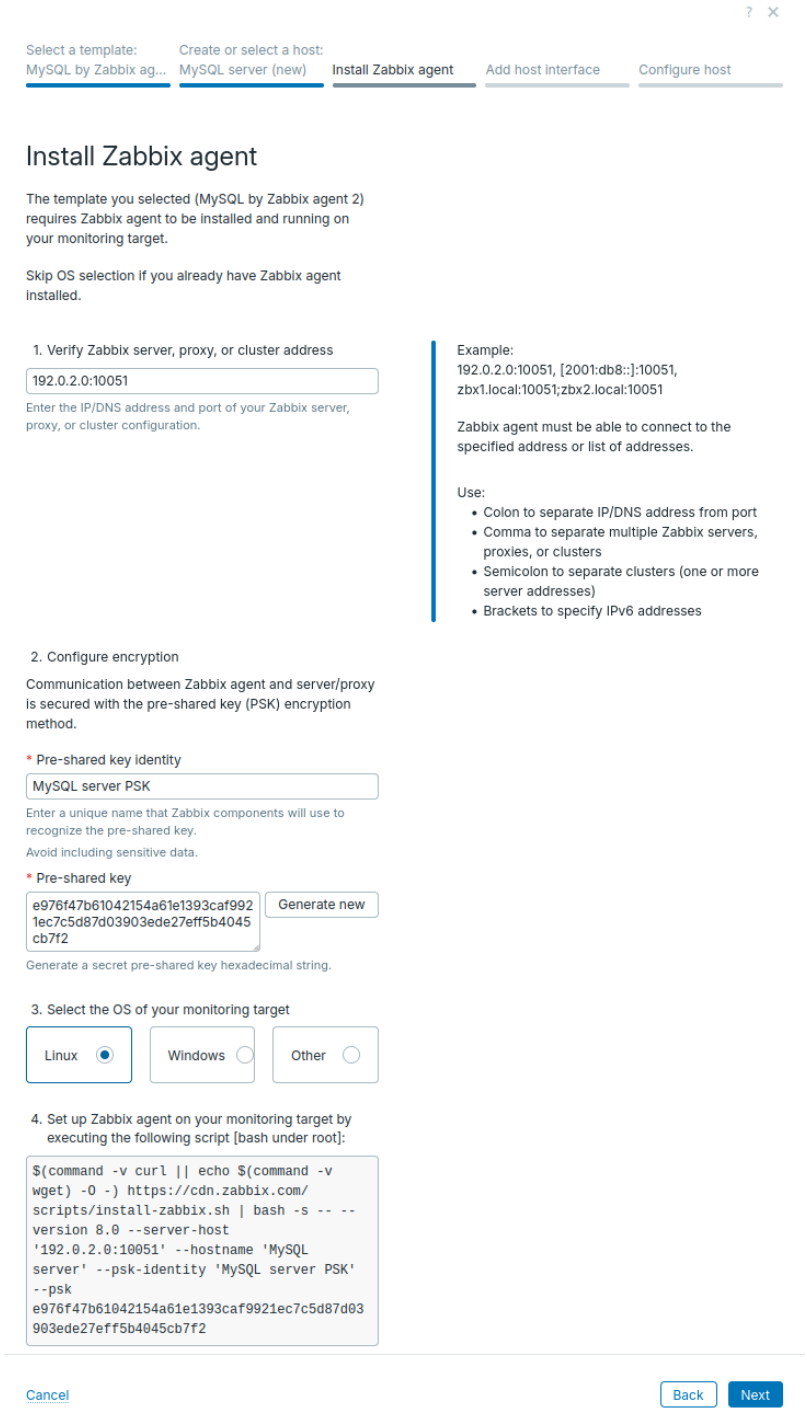
1. Inicie el Host Wizard en *Data collection* > *Hosts*.
2. Seleccione el template *MySQL by Zabbix agent 2*.



3. Cree un host de servidor MySQL y añádale un grupo. Este host virtual contendrá todos los datos relacionados con la monitorización del servidor MySQL.

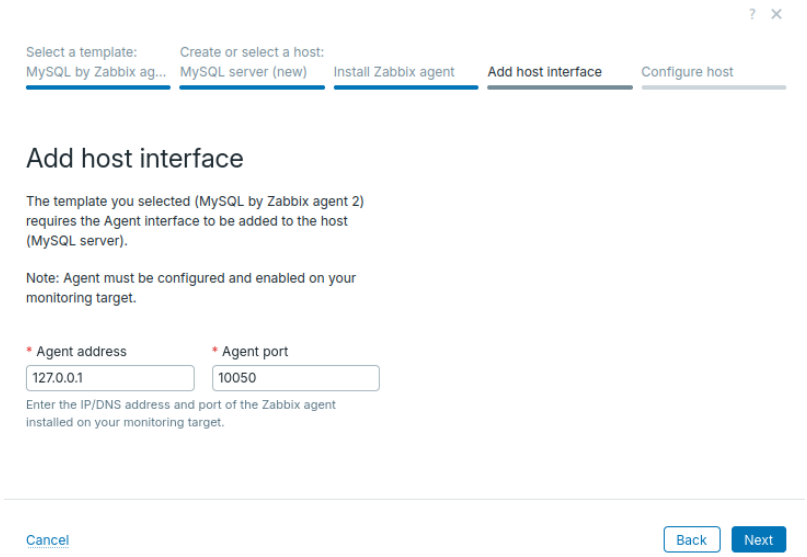


4. Configure e instale agent 2 (Zabbix agent 2 proporciona monitorización de MySQL de forma inmediata).

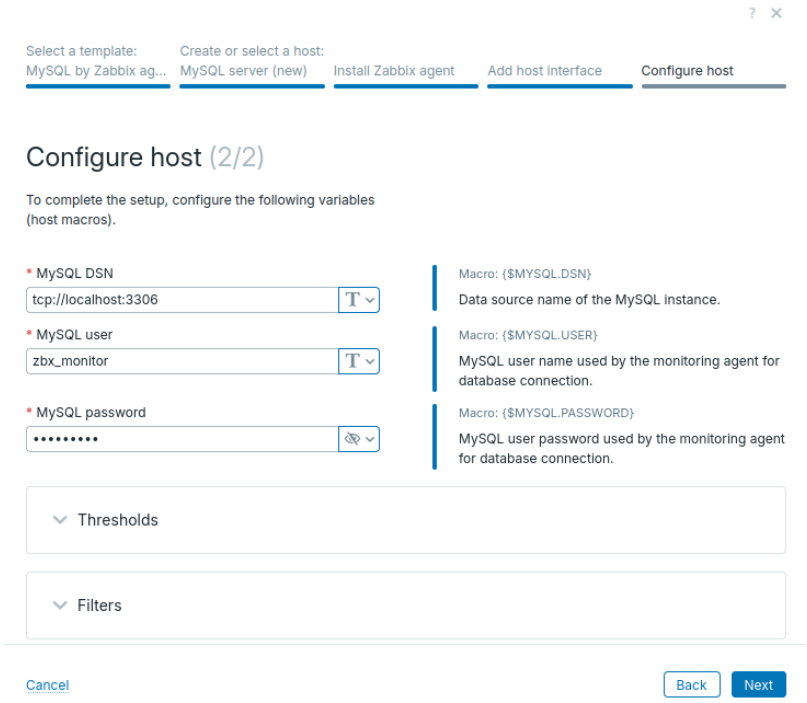


Para instalar el agent, ejecute el comando proporcionado en la máquina que aloja el servidor MySQL. Esto instalará el agent y configurará la conexión del agent con Zabbix server mediante cifrado.
 Si ya tiene agent 2 instalado, puede omitir este paso (haga clic en *Next*). Sin embargo, se le pedirá que proporcione la identidad de la clave precompartida y la clave precompartida para conectarse al agent.

5. Introduzca la dirección IP y el puerto del agent (servidor Linux).



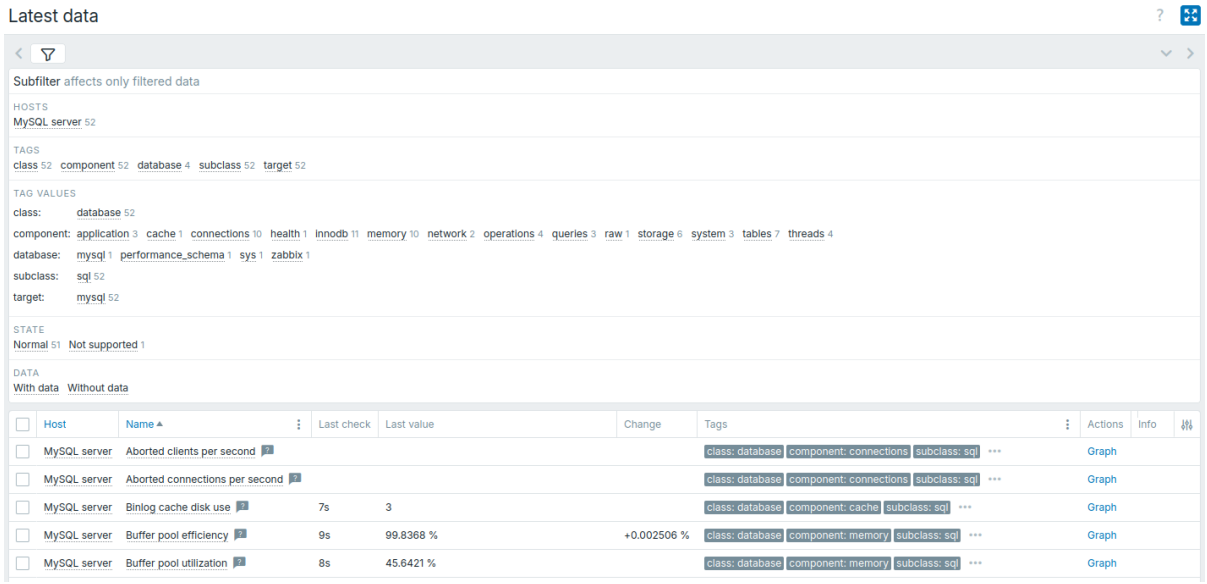
6. Termine de configurar el host. Importante: se le pedirá que introduzca el DSN del servidor MySQL, el usuario MySQL y la contraseña. Proporcione el usuario `zbx_monitor` con la contraseña que creó.



Puede dejar el resto de los ajustes con sus valores predeterminados.

6. Para completar el Host Wizard, haga clic en *Create* y *Finish*.

Ver los datos más recientes



¡Enhorabuena, la monitorización del server MySQL ha comenzado!

Zabbix ofrece opciones de visualización y envía alertas si surge algún problema. Para configurar alertas por correo electrónico, consulte Receiving problem notification.

Más guías

Esta sección de la documentación contiene recetas rápidas para configurar Zabbix para algunos objetivos de monitorización de uso común.

Está diseñada pensando en el nuevo usuario de Zabbix y puede utilizarse como un navegador a través de otras secciones de la documentación que contienen la información necesaria para resolver la tarea.

Las siguientes guías de referencia rápida están disponibles:

- Monitorizar Linux con Zabbix agent
- Monitorizar Windows con Zabbix agent
- Monitorizar Apache mediante HTTP
- Monitorizar MySQL con Zabbix agent 2
- Monitorizar VMware con Zabbix
- Monitorizar el tráfico de red con Zabbix
- Monitorizar el tráfico de red con Zabbix mediante comprobaciones activas
- Monitorizar sitios web con elementos Browser
- Monitorizar certificados de sitios web con Zabbix agent 2 (pasivo)
- Monitorizar un switch o router de red con Zabbix
- Monitorizar el registro de eventos de Windows mediante comprobaciones activas

Monitorizar Linux con agent de Zabbix

Introducción

Esta página le guía a través de los pasos necesarios para comenzar la monitorización básica de máquinas Linux con Zabbix. Los pasos descritos en este tutorial pueden aplicarse a cualquier sistema operativo basado en Linux.

Para quién está dirigida esta guía

Esta guía está diseñada para nuevos usuarios de Zabbix y contiene el conjunto mínimo de pasos necesarios para habilitar la monitorización básica de su máquina Linux. Si busca opciones de personalización avanzadas o necesita una configuración más compleja, consulte la sección Configuration del manual de Zabbix.

Requisitos previos

Antes de continuar con esta guía, debe descargar e instalar Zabbix server y Zabbix frontend de acuerdo con las instrucciones para su sistema operativo.

Instalar Zabbix agent

Zabbix agent es el proceso responsable de recopilar datos.

Compruebe la versión de su server Zabbix:

```
zabbix_server -V
```

Instale Zabbix agent de la misma versión (recomendado) en la máquina Linux que desea supervisar. Según sus necesidades de supervisión, puede ser la misma máquina donde está instalado el server Zabbix, o una máquina completamente diferente.

Elija el método de instalación más adecuado:

- Ejecutar como un contenedor Docker - consulte la lista de imágenes disponibles en Zabbix Docker repository.
- Instalar desde los packages de Zabbix (disponibles para Alma Linux, CentOS, Debian, Oracle Linux, Raspberry Pi OS, RHEL, Rocky Linux, SUSE Linux Enterprise Server, Ubuntu).
- Compilar from sources.

Configurar Zabbix para la monitorización

Zabbix agent puede recopilar métricas en modo activo o pasivo (simultáneamente).

> **Note:**
> Una comprobación pasiva es una solicitud simple de datos. Zabbix server o proxy solicita algunos datos (por ejemplo, la carga de CPU) y Zabbix agent devuelve el resultado al server. Las comprobaciones activas requieren un procesamiento más complejo. El agent primero debe recuperar del server(s) una lista de items para su procesamiento independiente y luego enviar los datos de vuelta en bloque. Consulte Comprobaciones pasivas y activas del agent para obtener más información.

Las templates de monitorización proporcionadas por Zabbix suelen ofrecer dos alternativas: una template para Zabbix agent y una template para Zabbix agent (active). Con la primera opción, el agent recopilará métricas en modo pasivo. Estas templates proporcionarán resultados de monitorización idénticos, pero utilizando distintos protocolos de comunicación.

La configuración adicional de Zabbix depende de si selecciona una template para comprobaciones de Zabbix agent activas o pasivas.

Comprobaciones pasivas

Zabbix agent

1. Abra el archivo de configuración del agent en la máquina donde está instalado el agent.

```
sudo vi /etc/zabbix/zabbix_agentd.conf
```

2. Añada la dirección IP o el nombre DNS (y el puerto opcional) de su server de Zabbix al parámetro Server. Por ejemplo:

```
Server=192.0.2.1:10051
```

El agent de Zabbix utilizará esta dirección para aceptar conexiones entrantes únicamente de los servers o proxies de Zabbix especificados para recuperar datos.

3. Reinicie el agent de Zabbix.

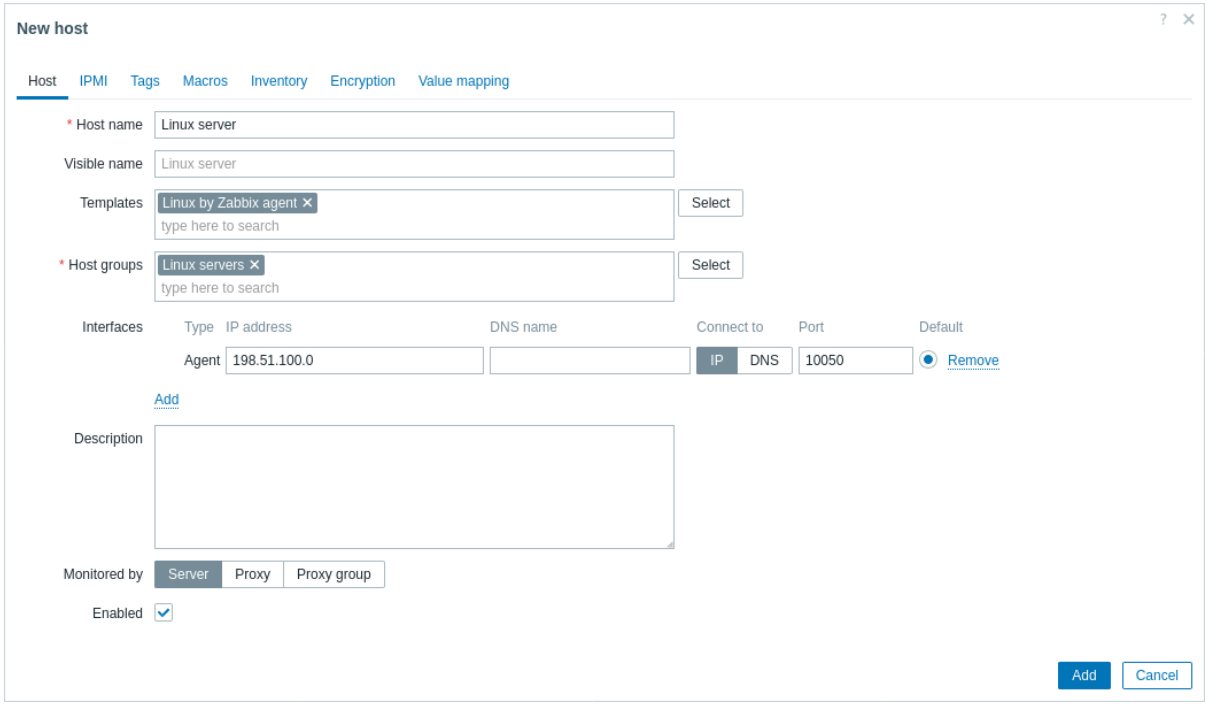
```
systemctl restart zabbix-agent
```

frontend de Zabbix

1. Inicie sesión en el frontend de Zabbix.

2. Cree un host en la interfaz web de Zabbix.

- En el campo *Host name*, introduzca un nombre de host (por ejemplo, ”Linux server”).
- En el campo *Templates*, escriba o seleccione el template ”Linux by Zabbix agent” que se vinculará al host.
- En el campo *Host groups*, escriba o seleccione un grupo de hosts (por ejemplo, ”Linux servers”).
- En el parámetro *Interfaces*, añada la interfaz *Agent* y especifique la dirección IP o el nombre DNS de la máquina Linux donde está instalado el agent.



3. Haga clic en *Add* para añadir el host. Este host representará la máquina Linux monitorizada.

Comprobaciones activas

Zabbix agent

1. Abra el archivo de configuración del agent en la máquina donde está instalado el agent:

```
sudo vi /etc/zabbix/zabbix_agentd.conf
```

2. Añada la dirección IP o el nombre DNS (y el puerto opcional) de su Zabbix server al parámetro ServerActive. Por ejemplo:

```
ServerActive=192.0.2.1:10051
```

Zabbix agent utilizará esta dirección para conectarse al puerto trapper de Zabbix server (predeterminado: 10051) y solicitar los datos de configuración para las comprobaciones activas.

3. Defina el parámetro Hostname, que debe coincidir con el nombre del host que se definirá en Zabbix frontend. En nuestro ejemplo, es:

```
Hostname=Linux server
```

El valor de Hostname debe coincidir porque, para las comprobaciones activas, Zabbix agent lo utiliza para recuperar la configuración correcta del host desde el server. En concreto, el agent inicia una conexión con el server y se identifica mediante el valor de Hostname. A continuación, el server proporciona la configuración de monitorización para ese host. Si estos valores difieren, el agent no recibirá la configuración adecuada, lo que provocará la ausencia de métricas o problemas de monitorización.

4. Reinicie Zabbix agent.

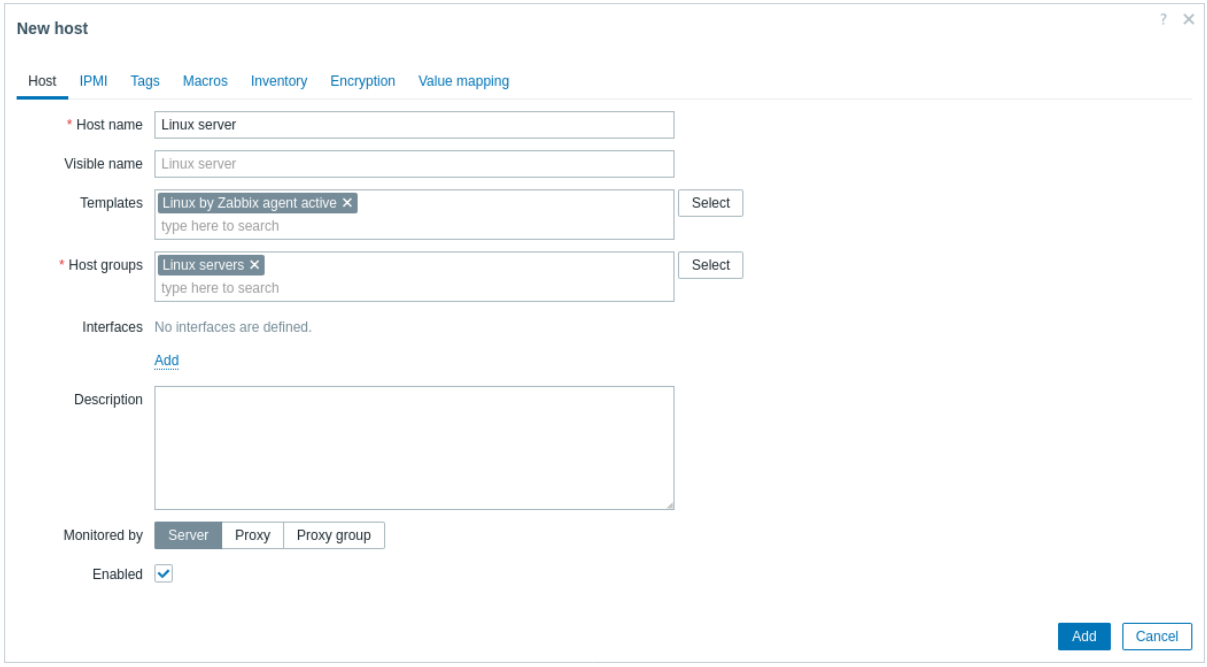
```
systemctl restart zabbix-agent
```

frontend de Zabbix

1. Inicie sesión en el frontend de Zabbix.

2. Cree un host en la interfaz web de Zabbix.

- En el campo *Host name*, introduzca un nombre de host (por ejemplo, ”Linux server”) que coincida con el valor del parámetro *Hostname* definido anteriormente en el archivo de configuración del agent.
- En el campo *Templates*, escriba o seleccione el template ”Linux by Zabbix agent active” que se vinculará al host.
- En el campo *Host groups*, escriba o seleccione un grupo de hosts (por ejemplo, ”Linux servers”).
- Deje *Interfaces* sin definir; no es necesaria una interfaz para las comprobaciones activas porque el agent inicia la conexión al server en lugar de esperar una conexión desde el server.



3. Haga clic en *Add* para añadir el host. Este host representará la máquina Linux supervisada.

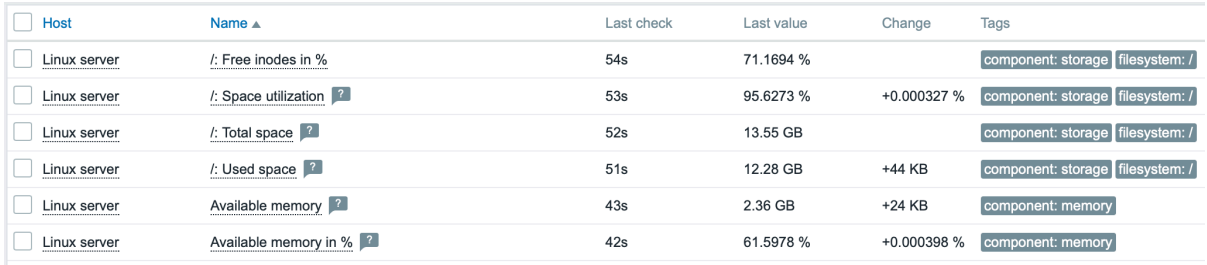
Ver métricas recopiladas

¡Enhorabuena! En este punto, Zabbix ya está supervisando su máquina Linux.

Para ver las métricas recopiladas, abra la sección de menú *Monitoring* > *Hosts* menu section y haga clic en *Latest data* junto al host.



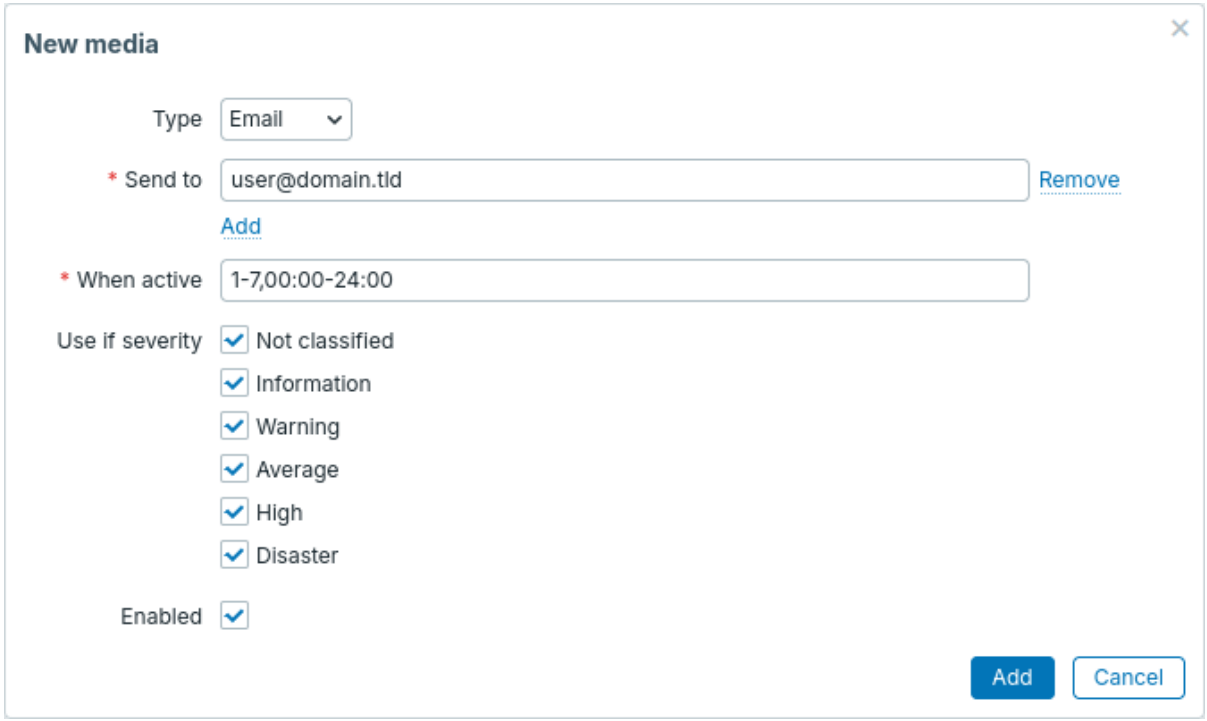
Esta acción abrirá una lista de todas las métricas más recientes recopiladas del host del server Linux.



Configurar alertas de problemas

Zabbix puede notificarle sobre un problema en su infraestructura mediante varios métodos.
Esta guía proporciona los pasos de configuración para enviar alertas por correo electrónico.

1. Vaya a *User settings* > *Profile*, cambie a la pestaña *Media* y
agregue su correo electrónico.



2. Siga la guía para Receiving problem notification.

La próxima vez que Zabbix detecte un problema, debería recibir una alerta por correo electrónico.

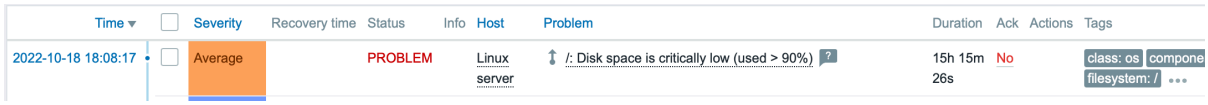
Pruebe su configuración

En Linux, puede simular una alta carga de CPU y, como resultado, recibir una alerta de problema ejecutando:

```
cat /dev/urandom | md5sum
```

Es posible que deba ejecutar varios procesos md5sum para que la carga de CPU supere el umbral.

Cuando Zabbix detecte el problema, aparecerá en la sección *Monitoring* > *Problems*.



Si las alertas están configuradas, también recibirá la notificación del problema.

Ver también:

- Creación de un item - cómo empezar a supervisar métricas adicionales (supervisión personalizada sin templates).
- Items del agent de Zabbix, Items del agent de Zabbix para Windows - lista completa de métricas que puede supervisar usando el agent de Zabbix en Windows.
- Escalaciones de problemas - cómo crear escenarios de alertas de varios pasos (p. ej., primero enviar un mensaje al administrador del sistema y luego, si un problema no se resuelve en 45 minutos, enviar un mensaje al responsable del centro de datos).
- Instalación desde paquetes - cómo instalar los componentes de Zabbix usando los paquetes oficiales RPM y DEB para varias distribuciones de Linux, garantizando el acceso a las funciones más recientes y a las correcciones de errores.

Monitor Windows con Zabbix agent

Introducción

Esta página le guía por los pasos necesarios para comenzar a realizar una monitorización básica de máquinas Windows con Zabbix.

A quién está dirigida esta guía

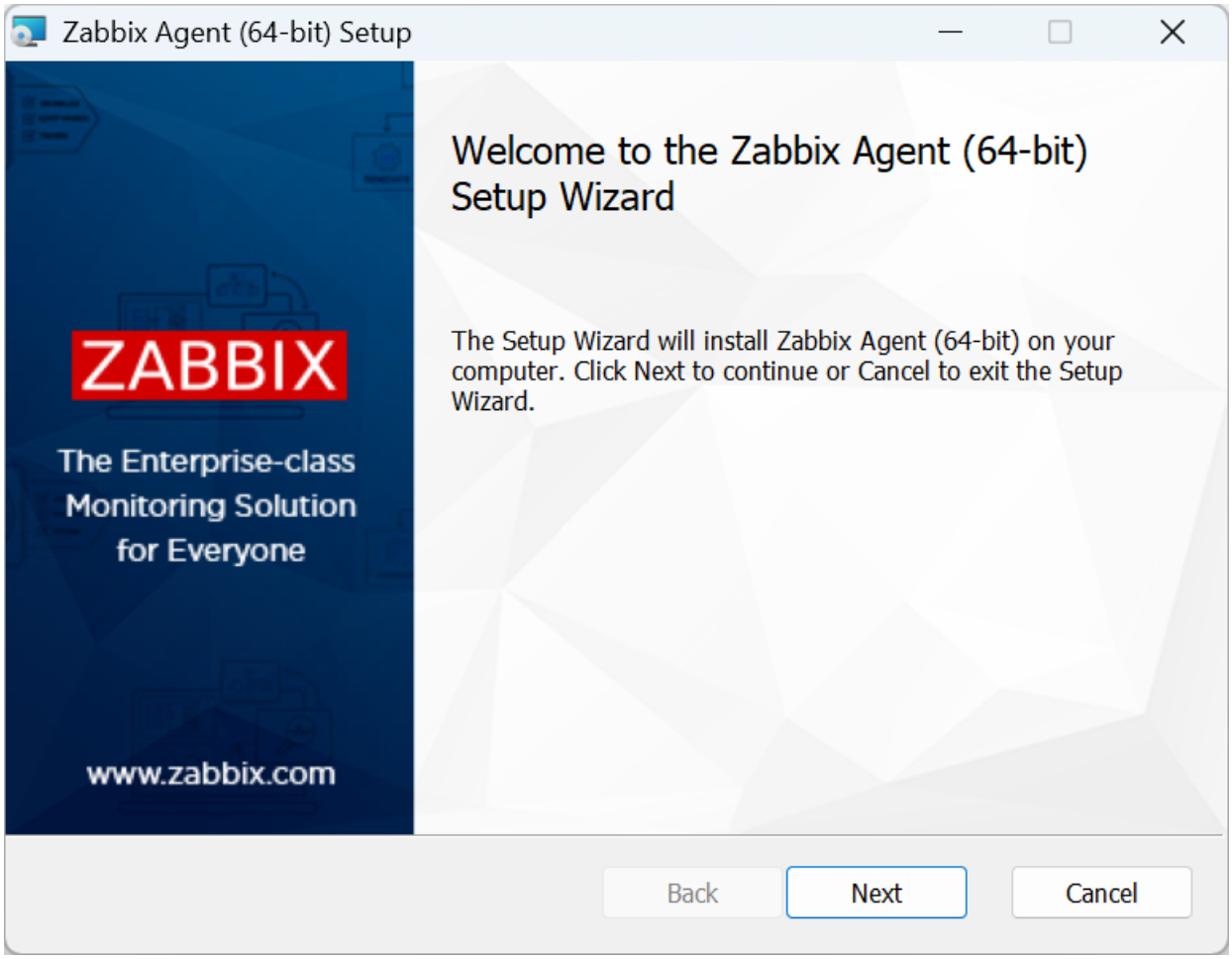
Esta guía está diseñada para nuevos usuarios de Zabbix y contiene el conjunto mínimo de pasos necesarios para habilitar la monitorización básica de su máquina Windows. Si busca opciones de personalización avanzadas o necesita una configuración más compleja, consulte la sección Configuración del manual de Zabbix.

Requisitos previos

Antes de continuar con esta guía de instalación, debe descargar e instalar el server y el frontend de Zabbix siguiendo las instrucciones correspondientes a su sistema operativo.

Instalar Zabbix agent

Zabbix agent es el proceso responsable de recopilar datos. Debe instalarlo en la máquina Windows que desea supervisar. Siga las instrucciones de instalación de Zabbix agent para Windows.



Configurar Zabbix para la supervisión

Zabbix agent puede recopilar métricas en modo activo o pasivo (simultáneamente).

Note:
Una comprobación pasiva es una simple solicitud de datos. Zabbix server o proxy solicita algunos datos (por ejemplo, la carga de CPU) y Zabbix agent envía el resultado de vuelta al server. Las comprobaciones activas requieren un procesamiento más complejo. El agent primero debe recuperar del server(s) una lista de items para su procesamiento independiente y luego enviar los datos en bloque de vuelta. Consulte Passive and active agent checks para obtener más información.

Las templates de supervisión proporcionadas por Zabbix suelen ofrecer dos alternativas: una template para Zabbix agent y una template para Zabbix agent (active). Con la primera opción, el agent recopilará métricas en modo pasivo. Estas templates ofrecerán resultados de supervisión idénticos, pero utilizando distintos protocolos de comunicación.

La configuración adicional de Zabbix depende de si selecciona una template para comprobaciones de Zabbix agent activas o pasivas.

Comprobaciones pasivas

Frontend de Zabbix

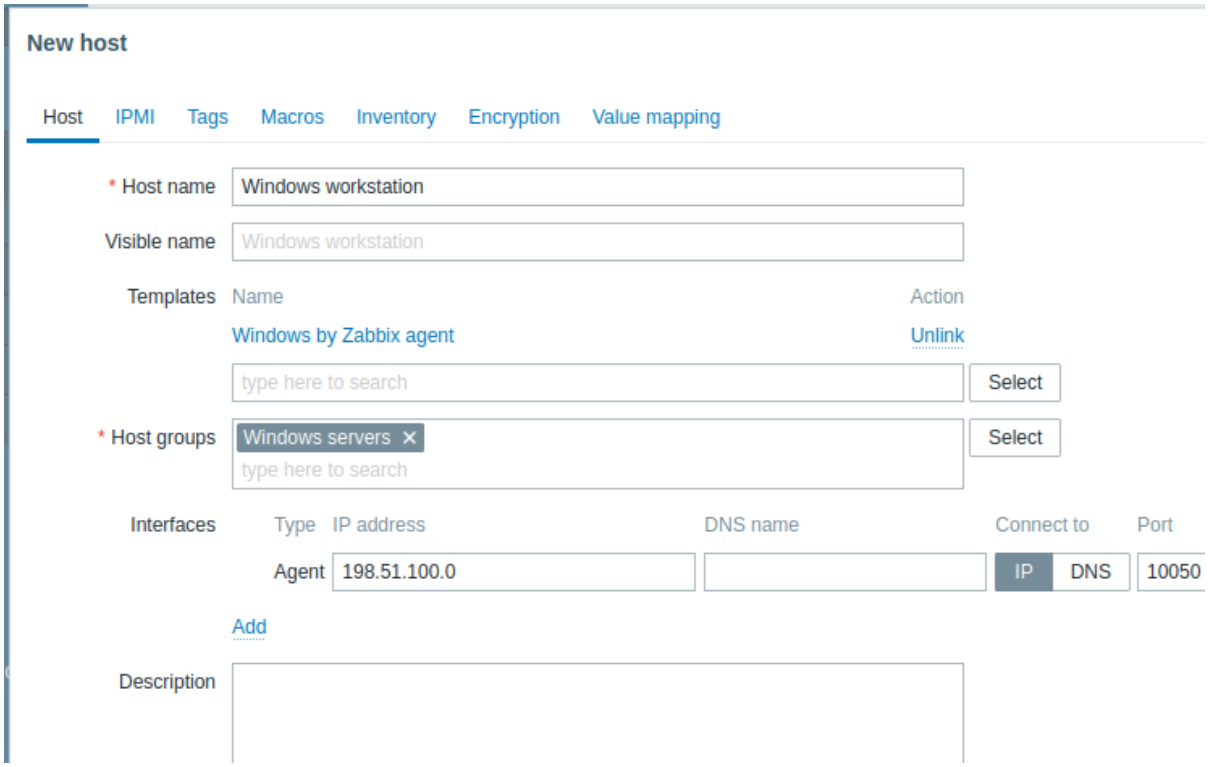
1. Inicie sesión en el frontend de Zabbix.

2. Cree un host en la interfaz web de Zabbix.

Este host representará su máquina Windows.

3. En el parámetro *Interfaces*, agregue la interfaz *Agent* y especifique la dirección IP o el nombre DNS de la máquina Windows donde está instalado el agent.

4. En el parámetro *Templates*, escriba o seleccione *Windows by Zabbix agent*.



Zabbix agent

Para las comprobaciones pasivas, el Zabbix agent necesita conocer la dirección IP o el nombre DNS del Zabbix server. Si ha proporcionado la información correcta durante la instalación del agent, el archivo de configuración ya se ha actualizado. De lo contrario, debe especificarla manualmente. Abra el archivo de configuración del agent y añada la IP/DNS de su Zabbix server al parámetro `Server`.

Ejemplo:

```
Server=192.0.2.22
```

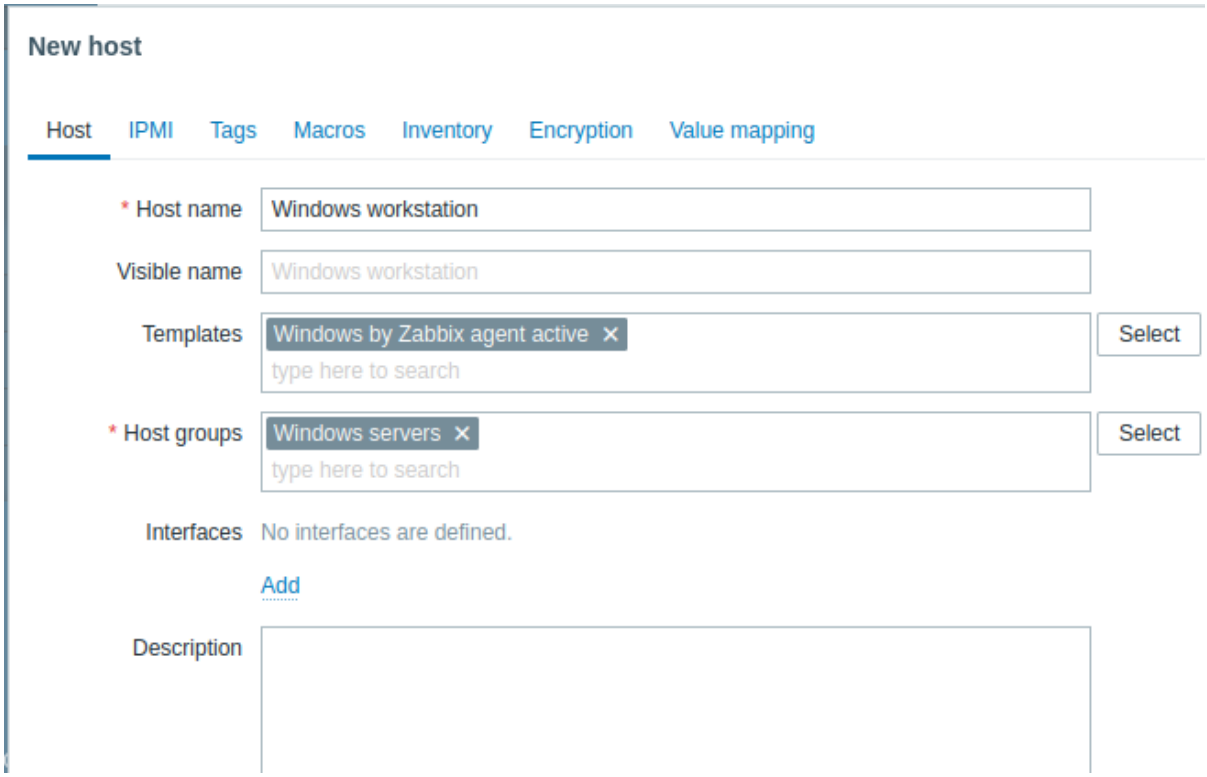
Comprobaciones activas

frontend de Zabbix

1. Inicie sesión en el frontend de Zabbix.
2. Cree un host en la interfaz web de Zabbix.

Este host representará su máquina Windows.

3. En el parámetro *Templates*, escriba o seleccione *Windows by Zabbix agent active*.



Zabbix agent

En la carpeta `C:\Program files\Zabbix Agent` abra el archivo `zabbix_agentd.conf` y agregue:

- El nombre del host que creó en la interfaz web de Zabbix al parámetro *Hostname*.
- La dirección IP o el nombre DNS de su máquina Zabbix server al parámetro *ServerActive* (podría estar rellenado previamente si lo proporcionó durante la configuración de Zabbix agent).

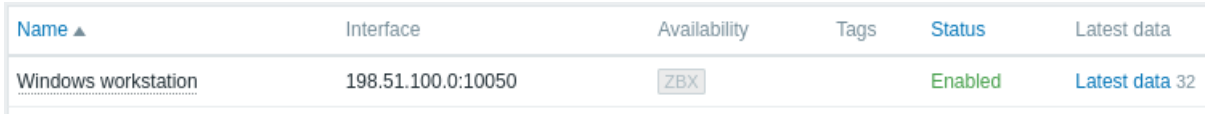
Ejemplo:

```
ServerActive= 192.0.2.22
Hostname=Windows workstation
```

Ver métricas recopiladas

¡Enhorabuena! En este punto, Zabbix ya está supervisando su máquina Windows.

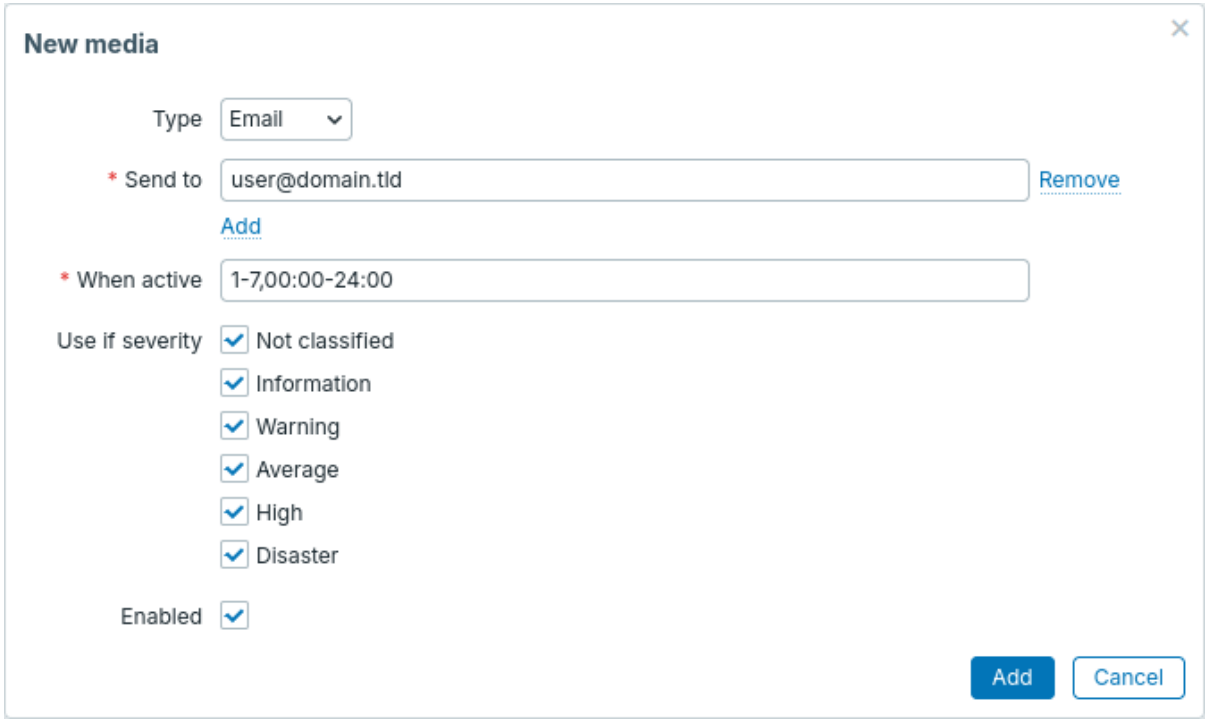
Para ver las métricas recopiladas, abra la sección de menú *Monitoring* > *Hosts* menu section y haga clic en *Latest data* junto al host.



Configurar alertas de problemas

Zabbix puede notificarle sobre un problema en su infraestructura mediante varios métodos. Esta guía proporciona los pasos de configuración para enviar alertas por correo electrónico.

1. Vaya a *User settings* > *Profile*, cambie a la pestaña *Media* y añada su correo electrónico.



2. Siga la guía para Receiving problem notification.

La próxima vez que Zabbix detecte un problema, debería recibir una alerta por correo electrónico.

Note:
En Windows, puede usar la utilidad CpuStres para simular una carga alta de CPU y, como resultado, recibir una alerta de problema.

Vea también:

- Creating an item - cómo empezar a supervisar métricas adicionales (supervisión personalizada sin templates).
- Zabbix agent items, Zabbix agent items for Windows - lista completa de métricas que puede supervisar usando Zabbix agent en Windows.
- Problem escalations - cómo crear escenarios de alertas de varios pasos (por ejemplo, primero enviar un mensaje al administrador del sistema y luego, si un problema no se resuelve en 45 minutos, enviar un mensaje al responsable del centro de datos).

Supervisar Apache mediante HTTP

Introducción

Esta página muestra una forma rápida y sencilla de comenzar a supervisar un servidor web Apache sin instalar ningún software adicional.

Para quién está dirigida esta guía

Esta guía está diseñada para nuevos usuarios de Zabbix y contiene el conjunto mínimo de pasos necesarios para habilitar la monitorización básica de su instalación de Apache. Si busca opciones de personalización avanzadas o necesita una configuración más compleja, consulte la sección Configuration del manual de Zabbix.

Requisitos previos

Antes de continuar con esta guía de instalación, debe descargar e instalar Zabbix server y Zabbix frontend según las instrucciones para su sistema operativo.

Preparar Apache

1. Compruebe qué versión de Apache está utilizando:

En un sistema basado en RHEL, ejecute:

```
httpd -v
```

En Debian/Ubuntu, ejecute:

```
apache2 -v
```

2. Asegúrese de que el módulo Status esté habilitado en su instancia de Apache.

En un sistema basado en RHEL, ejecute:

```
httpd -M | grep status
status_module (shared)
```

En Debian/Ubuntu, ejecute:

```
apache2ctl -M | grep status
status_module (shared)
```

Si no ve status_module en la lista, habilite el módulo ejecutando:

En un sistema basado en RHEL, ejecute:

```
LoadModule status_module /usr/lib/apache2/modules/mod_status.so
```

En Debian/Ubuntu, ejecute:

```
sudo /usr/sbin/a2enmod status
```

3. Edite el archivo de configuración de Apache para permitir el acceso a los informes de estado desde la IP del server de Zabbix.

En un sistema basado en RHEL: `/etc/httpd/conf.modules.d/status.conf`:

```
sudo vi /etc/httpd/conf.modules.d/status.conf
```

En Debian/Ubuntu: `/etc/apache2/mods-enabled/status.conf`:

```
sudo vi /etc/apache2/mods-enabled/status.conf
```

Añada las siguientes líneas al archivo (**reemplace 198.51.100.255** por la dirección IP de su server de Zabbix):

- Para Apache 2.2:
  <Location /server-status> SetHandler server-status

  Order Deny,Allow Deny from all Allow from 198.51.100.255 </Location>
- Para Apache 2.4:
  <Location ”/server-status”> SetHandler server-status Require ip 198.51.100.255 </Location>

4. Reinicie Apache

En un sistema basado en RHEL, ejecute:

```
sudo systemctl restart httpd
```

En Debian/Ubuntu, ejecute:

```
sudo systemctl restart apache2
```

5. Para comprobar si todo está configurado correctamente, ejecute (**reemplace 198.51.100.255** por la dirección IP de su server de Zabbix):

```
curl 198.51.100.255/server-status
```

La respuesta debería contener estadísticas del servidor web Apache.

Configurar Zabbix para la monitorización

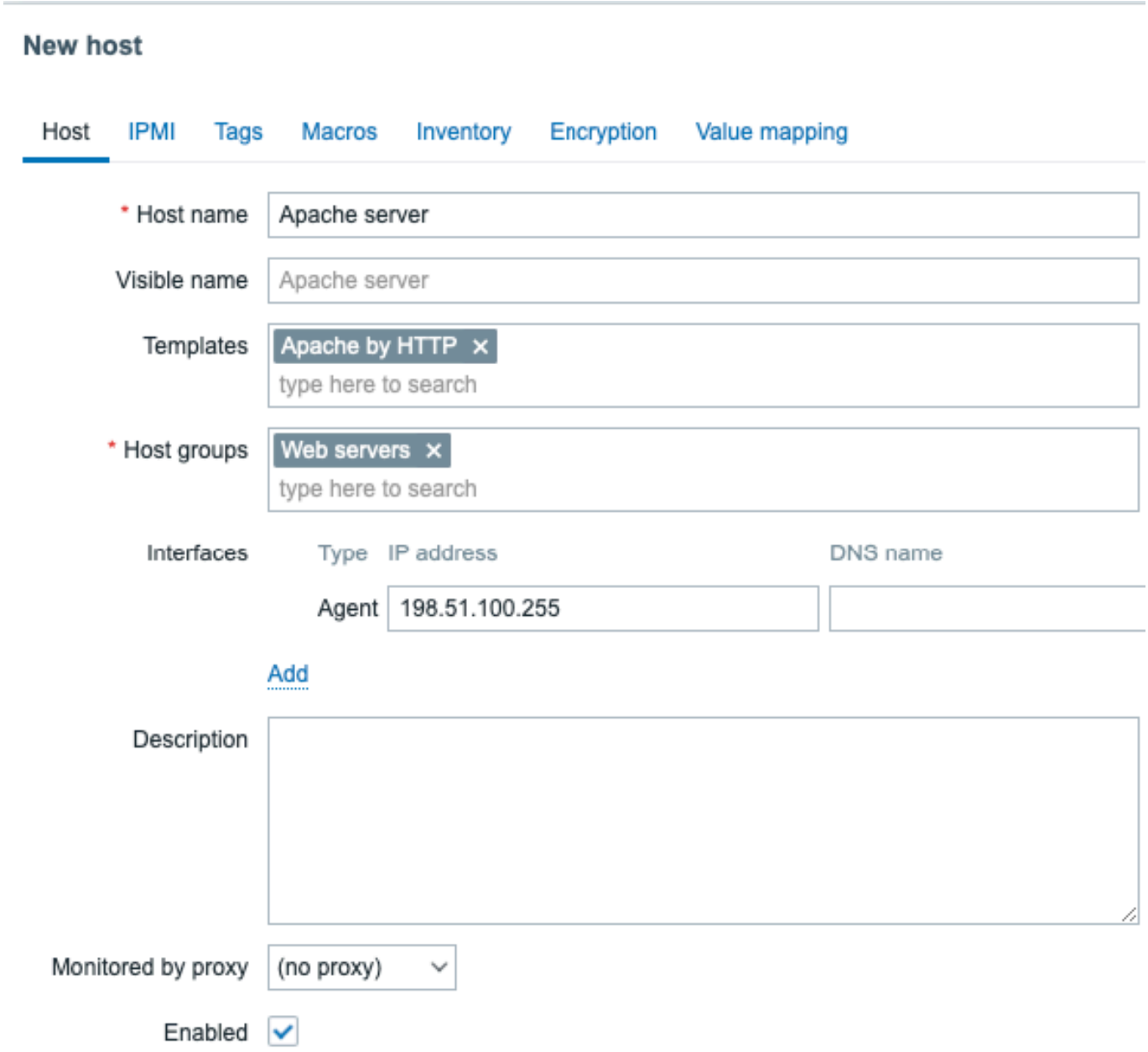
1. Inicie sesión en el frontend de Zabbix.

2. Cree un host en la interfaz web de Zabbix.

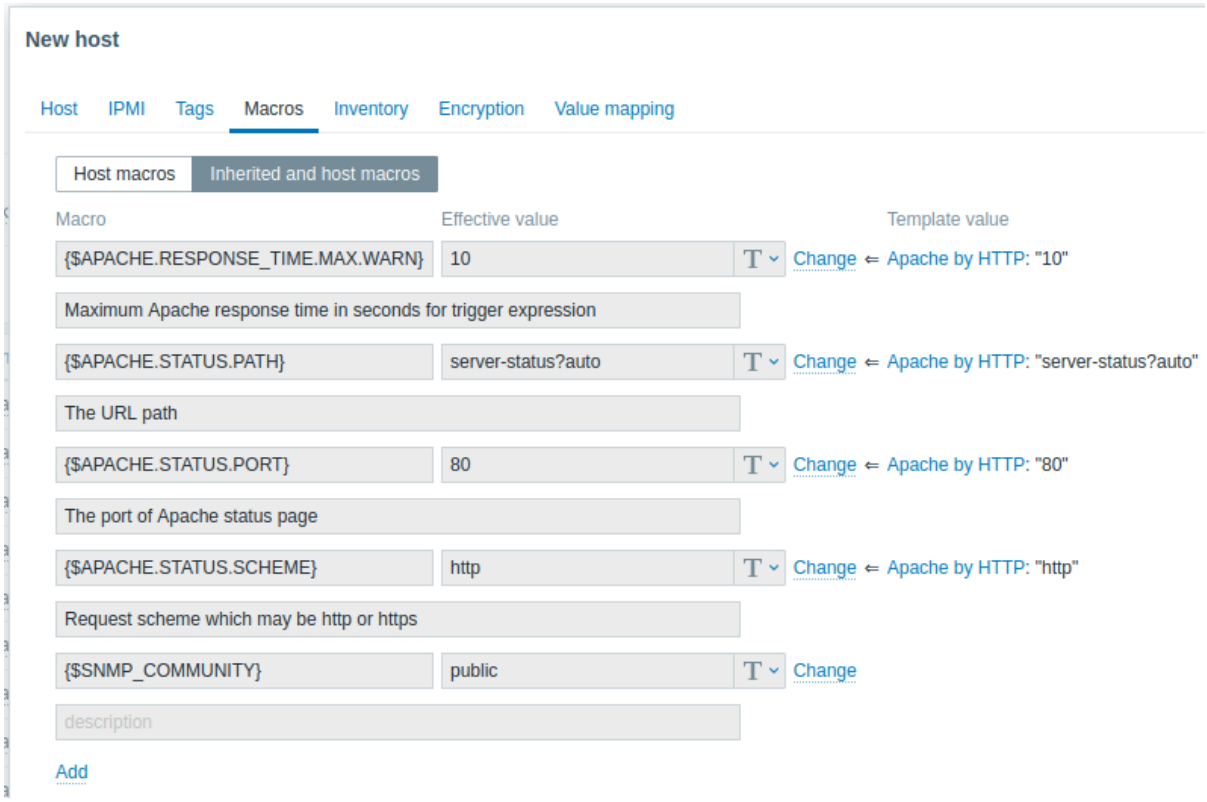
Este host representará su servidor Apache.

3. En el parámetro *Interfaces*, añada una interfaz *Agent* y especifique la dirección IP de su instancia de Apache. **No necesita instalar Zabbix agent en la máquina**, la interfaz solo se usará para resolver la macro {HOST.CONN}. Esta macro se usa en los items de template para localizar la instancia de Apache.

4. En el parámetro *Templates*, escriba o seleccione *Apache by HTTP*.



5. Cambie a la pestaña **Macros** y seleccione el modo *Inherited and host macros*. Compruebe que los valores de las macros {$APACHE.STATUS.PORT} y {$APACHE.STATUS.SCHEME} se ajustan a la configuración de su instalación. De forma predeterminada, el puerto es 80 y el esquema es http. Cambie los valores de las macros si utiliza un puerto y/o esquema diferentes.



Ver métricas recopiladas

¡Enhorabuena! En este punto, Zabbix ya está supervisando su servidor web Apache.

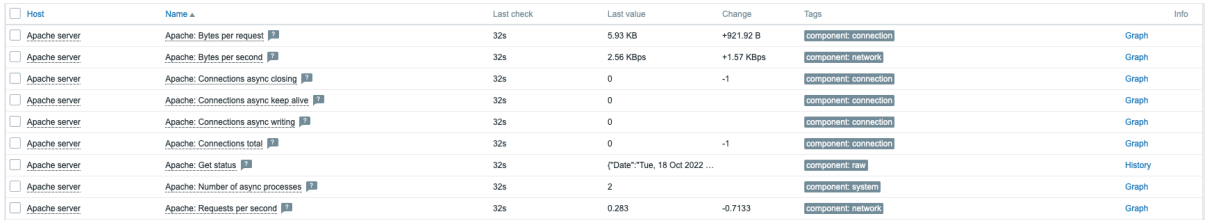
Para ver las métricas recopiladas, abra la sección de menú *Monitoring* > *Hosts* menu section y haga clic en *Dashboards* junto al host.



Esta acción le llevará al dashboard del host con las métricas más importantes recopiladas de la página /server-status de Apache.



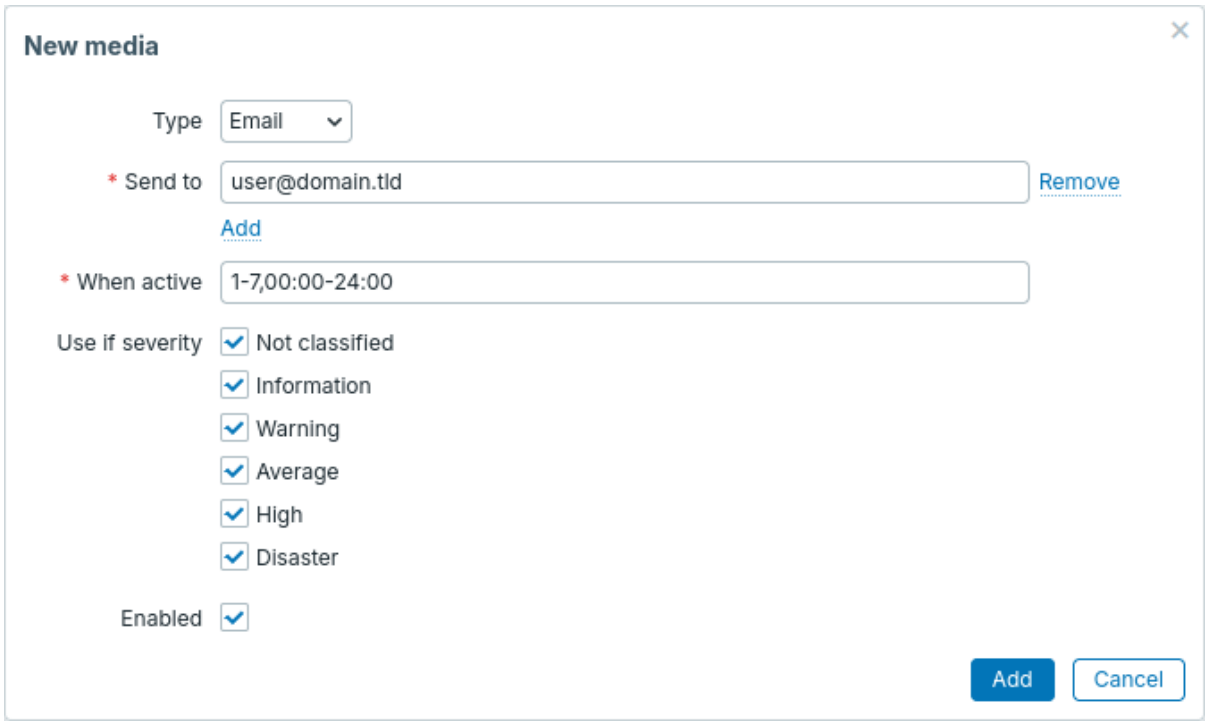
Como alternativa, desde *Monitoring* > *Hosts*, puede hacer clic en *Latest data* para ver todas las métricas recopiladas más recientes en una lista.



Configurar alertas de problemas

Zabbix puede notificarle sobre un problema en su infraestructura mediante varios métodos.
Esta guía proporciona los pasos de configuración para enviar alertas por correo electrónico.

1. Vaya a *User settings* > *Profile*, cambie a la pestaña *Media* y añada su correo electrónico.



2. Siga la guía para Receiving problem notification.

La próxima vez que Zabbix detecte un problema, debería recibir una alerta por correo electrónico.

Pruebe su configuración

Para simular un problema real y recibir una alerta de prueba de problema:

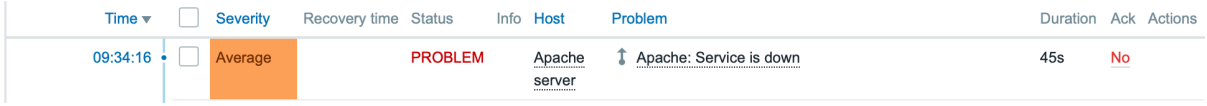
1. Abra la configuración del host *Apache server* en Zabbix.

2. Cambie a la pestaña Macros y seleccione *Inherited and host macros*.

3. Haga clic en *Change* junto a la macro {$APACHE.STATUS.PORT} y establezca un puerto diferente.

4. Haga clic en *Update* para guardar la configuración del host.

5. En unos minutos, Zabbix detectará el problema *Apache service is down*, porque ahora no puede conectarse a la instancia. Aparecerá en la sección *Monitoring* > *Problems*.



Si las alertas están configuradas, también recibirá la notificación del problema.

6. Vuelva a cambiar el valor de la macro para resolver el problema y continuar supervisando Apache.

Ver también:

- Endurecimiento del web server - configuración recomendada para una mayor seguridad del web server.
- Creación de un item - cómo empezar a supervisar métricas adicionales.
- Items HTTP - cómo supervisar métricas personalizadas usando HTTP agent.
- Escalaciones de problemas - cómo crear escenarios de alertas de varios pasos (por ejemplo, primero enviar un mensaje al administrador del sistema y, luego, si un problema no se resuelve en 45 minutos, enviar un mensaje al responsable del centro de datos).

Monitor MySQL con Zabbix agent 2

Introducción

Esta página le guía a través de los pasos necesarios para comenzar la monitorización básica de un server MySQL.

Para monitorizar un server MySQL, existen varios enfoques: Zabbix agent, Zabbix agent 2 o el estándar Open Database Connectivity (ODBC). El enfoque principal de esta guía es la monitorización de un server MySQL con Zabbix agent 2, que es el enfoque **recomendado** debido a su configuración fluida en distintos entornos. Sin embargo, esta página también ofrece instrucciones para los otros enfoques, así que puede elegir el que mejor se adapte a sus requisitos.

Para quién está dirigida esta guía

Esta guía está diseñada para nuevos usuarios de Zabbix y contiene el conjunto mínimo de pasos necesarios para habilitar la monitorización básica de un server MySQL. Si busca opciones de personalización avanzadas o necesita una configuración más compleja, consulte la sección Configuration del manual de Zabbix.

Requisitos previos

Antes de continuar con esta guía, debe descargar e instalar Zabbix server, Zabbix frontend y Zabbix agent 2 según las instrucciones para su sistema operativo.

Según su entorno, algunos pasos de esta guía pueden diferir ligeramente. Esta guía se basa en un entorno que ejecuta Ubuntu.

Crear usuario de MySQL

Para supervisar un server MySQL, Zabbix requiere acceso a él y a sus procesos. Su instalación de MySQL ya tiene un usuario con el nivel de acceso requerido (el usuario "zabbix" que se creó al instalar Zabbix), sin embargo, este usuario tiene más privilegios de los necesarios para una supervisión simple (privilegios para DROP databases, DELETE entradas de tablas, etc.). Por lo tanto, es necesario crear un usuario de MySQL con el propósito de *solo* supervisar el server MySQL.

1. Conéctese al cliente de MySQL, cree un usuario "zbx_monitor" (reemplace <*password*> para el usuario "zbx_monitor" por una contraseña de su elección), y GRANT los privilegios necesarios al usuario:

```
mysql -u root -p
# Enter password:

mysql> CREATE USER 'zbx_monitor'@'%' IDENTIFIED BY '<password>';
mysql> GRANT REPLICATION CLIENT,PROCESS,SHOW DATABASES,SHOW VIEW ON *.* TO 'zbx_monitor'@'%';
mysql> quit;
```

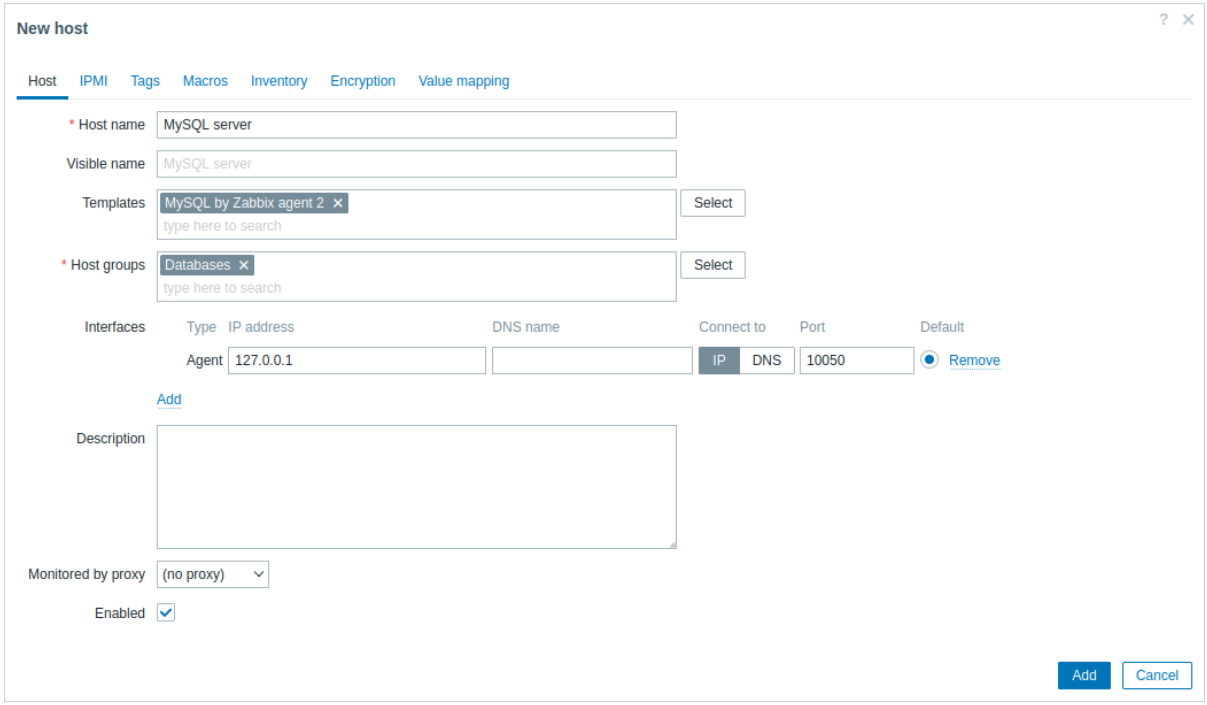
Una vez creado el usuario, puede pasar al siguiente paso.

Configurar el frontend de Zabbix

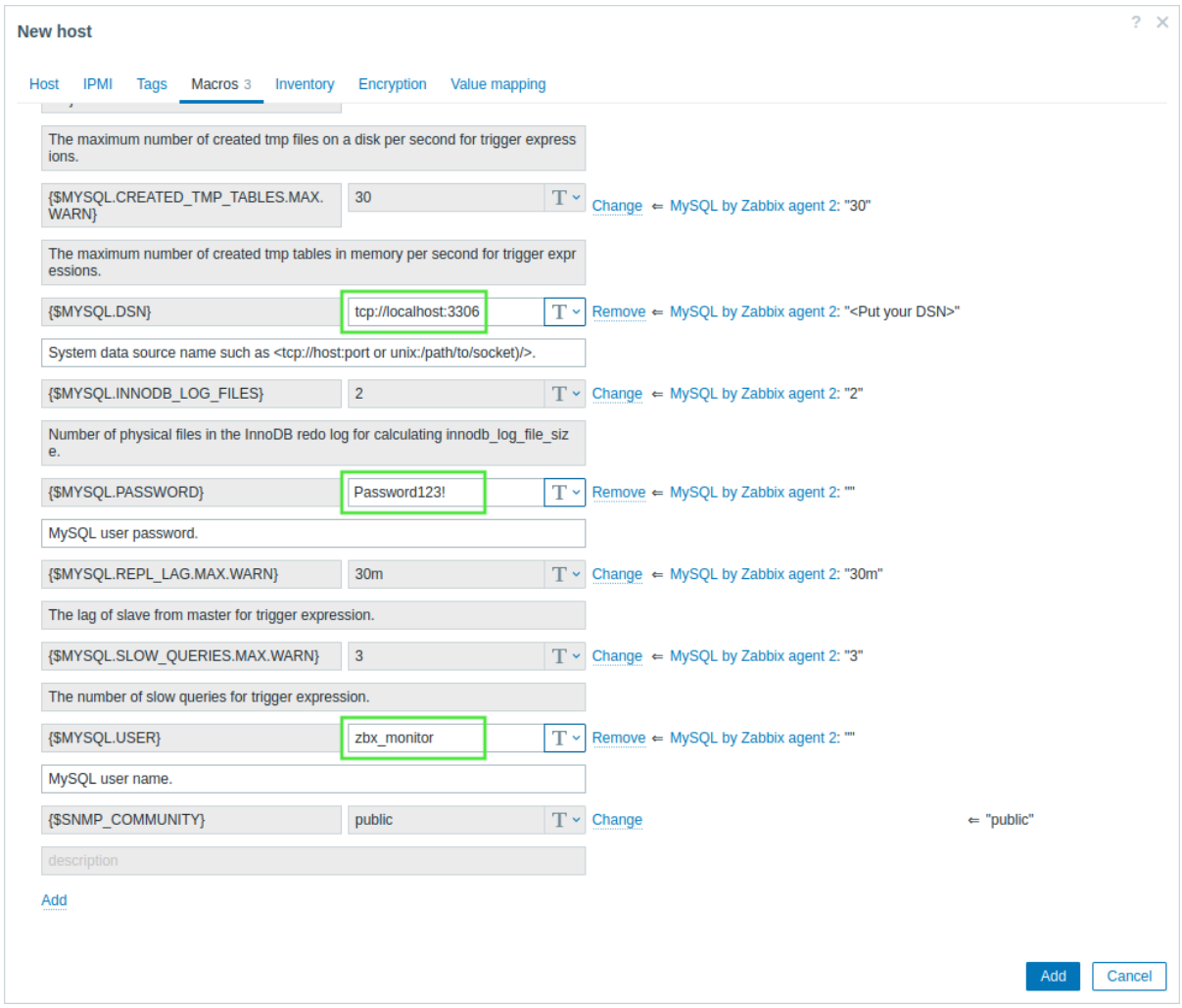
1. Inicie sesión en el frontend de Zabbix.

2. Cree un host en la interfaz web de Zabbix:

- En el campo *Host name*, introduzca un nombre de host (por ejemplo, "MySQL server").
- En el campo *Templates*, escriba o seleccione el template "MySQL by Zabbix agent 2" que se vinculará al host.
- En el campo *Host groups*, escriba o seleccione un grupo de hosts (por ejemplo, "Databases").
- En el campo *Interfaces*, agregue una interfaz de tipo "Agent" y especifique la dirección IP de su servidor MySQL. Esta guía usa "127.0.0.1" (localhost) para supervisar un servidor MySQL que está instalado en la misma máquina que el server de Zabbix y el agent 2 de Zabbix.



- En la pestaña *Macros*, cambie a *Inherited and host macros*, busque las siguientes macros y haga clic en *Change* junto al valor de la macro para actualizarlo:
  - {$MYSQL.DSN} - establezca la fuente de datos del servidor MySQL (la cadena de conexión de una sesión con nombre del archivo de configuración del plugin MySQL del agent 2 de Zabbix). Esta guía usa la fuente de datos predeterminada "tcp://localhost:3306" para supervisar un servidor MySQL que está instalado en la misma máquina que el server de Zabbix y el agent 2 de Zabbix.
  - {$MYSQL.PASSWORD} - establezca la contraseña del usuario MySQL creado previamente "zbx_monitor".
  - {$MYSQL.USER} - establezca el nombre del usuario MySQL creado previamente "zbx_monitor".

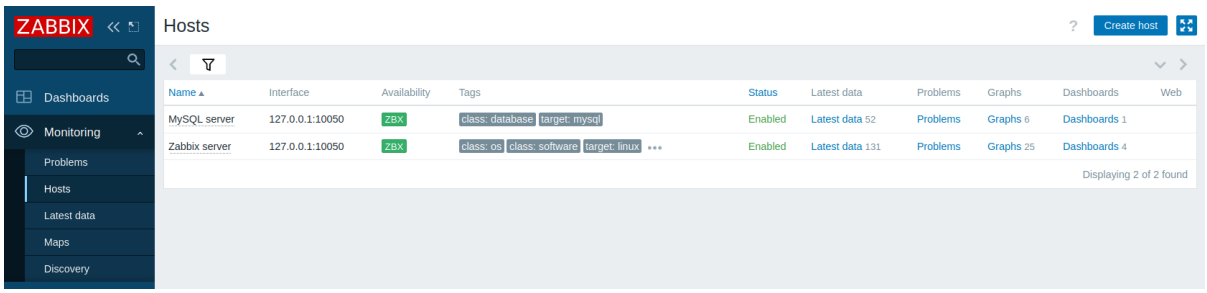


3. Haga clic en *Add* para agregar el host. Este host representará su servidor MySQL.

Ver métricas recopiladas

¡Felicidades! En este punto, Zabbix ya está supervisando su servidor MySQL.

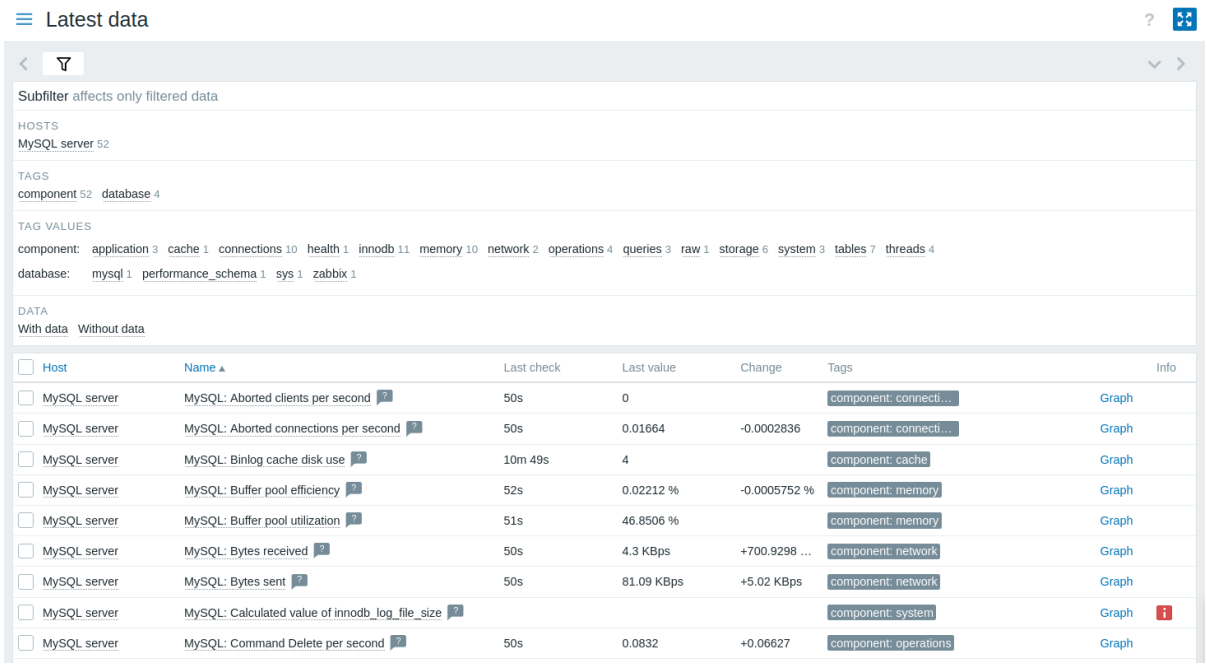
Para ver las métricas recopiladas, vaya a la sección de menú *Monitoring > Hosts* y haga clic en *Dashboards* junto al host.



Esta acción le llevará al dashboard del host (configurado a nivel de template) con las métricas más importantes recopiladas del servidor MySQL.



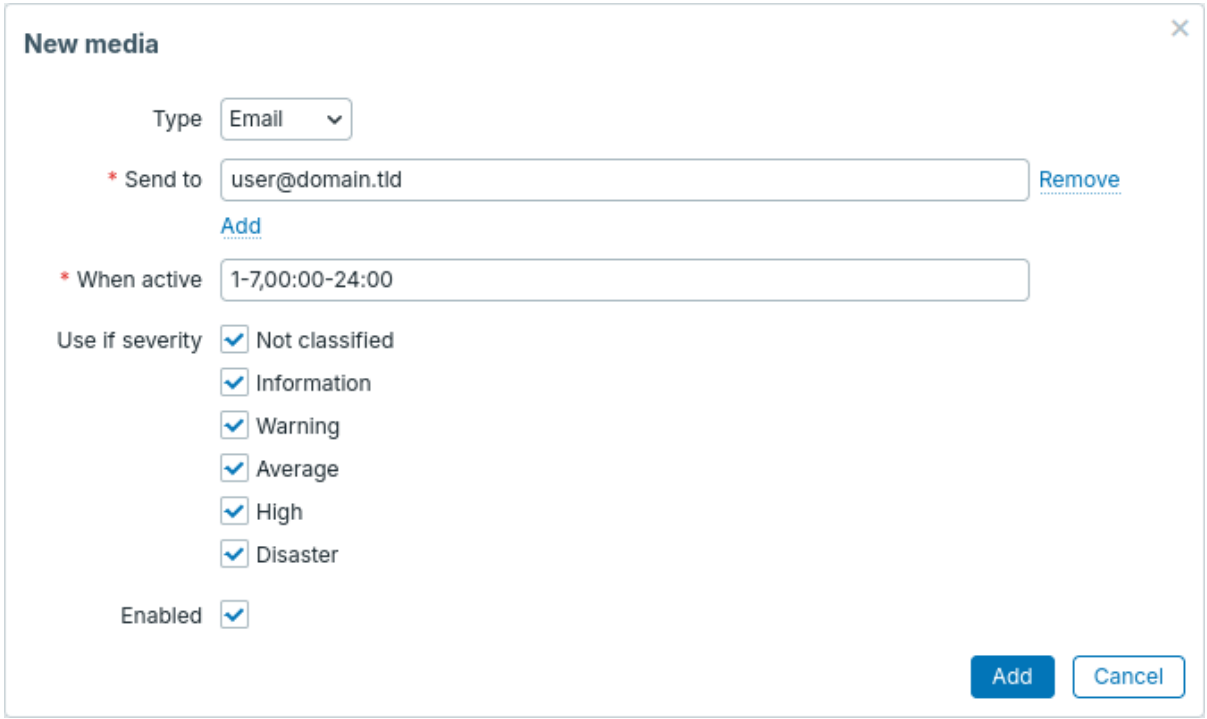
Como alternativa, desde la sección de menú *Monitoring > Hosts*, puede hacer clic en *Latest data* para ver en una lista todas las métricas recopiladas más recientes. Tenga en cuenta que se espera que el item *MySQL: Calculated value of innodb_log_file_size* no tenga datos, ya que el valor se calculará a partir de los datos de la última hora.



Configurar alertas de problemas

Zabbix puede notificarle sobre un problema con su infraestructura mediante varios métodos. Esta guía proporciona pasos básicos de configuración para enviar alertas por correo electrónico.

1. Vaya a *User settings > Profile*, cambie a la pestaña *Media* y añada su correo electrónico.



2. Siga la guía para Receiving a problem notification.

La próxima vez que Zabbix detecte un problema, debería recibir una alerta por correo electrónico.

Pruebe su configuración

Para probar su configuración, podemos simular un problema real actualizando la configuración del host en el frontend de Zabbix.

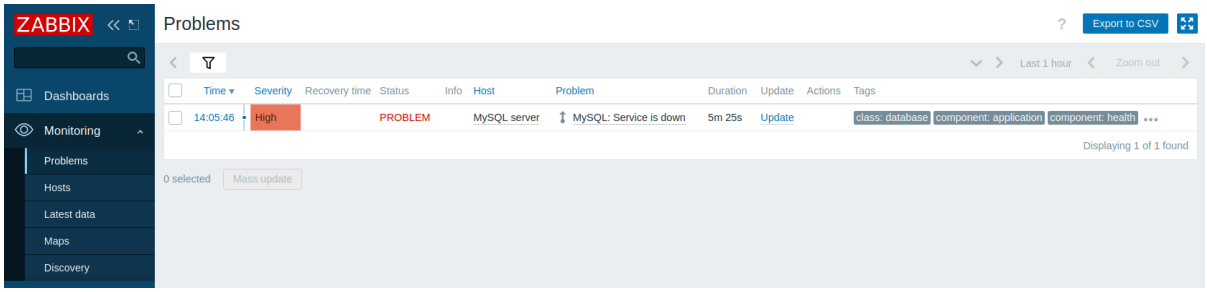
1. Abra la configuración del host de su servidor MySQL en Zabbix.

2. Vaya a la pestaña *Macros* y seleccione *Inherited and host macros*.

3. Haga clic en *Change* junto al valor de la macro {$MYSQL.USER} configurado previamente, por ejemplo, y establezca un nombre de usuario de MySQL diferente.

4. Haga clic en *Update* para actualizar la configuración del host.

5. En unos momentos, Zabbix detectará el problema ”MySQL: Service is down”, porque no podrá conectarse al servidor MySQL. El problema aparecerá en *Monitoring > Problems*.



Si las alertas están configuradas, también recibirá la notificación del problema.

6. Vuelva a cambiar el valor de la macro a su valor anterior para resolver el problema y continuar supervisando el servidor MySQL.

Otros enfoques para monitorizar MySQL

En lugar de monitorizar un server MySQL con Zabbix agent 2, también puede usar Zabbix agent o el estándar Open Database Connectivity (ODBC). Aunque se recomienda usar Zabbix agent 2, puede haber algunas configuraciones que no admitan Zabbix agent 2 o que requieran un enfoque personalizado.

La diferencia clave entre Zabbix agent y ODBC radica en el método de recopilación de datos: Zabbix agent se instala directamente en el server MySQL y recopila datos mediante su funcionalidad integrada, mientras que ODBC depende de un controlador ODBC para establecer una conexión con el server MySQL y recuperar datos mediante consultas SQL.

Aunque muchos de los pasos de configuración son similares a la monitorización de un server MySQL con Zabbix agent 2, existen algunas diferencias importantes: debe configurar Zabbix agent u ODBC para poder monitorizar un server MySQL. Las siguientes instrucciones le guiarán a través de estas **diferencias**.

Monitor MySQL con Zabbix agent

Para monitorizar un servidor MySQL con Zabbix agent, debe descargar e instalar Zabbix server, Zabbix frontend y Zabbix agent según las instrucciones para su sistema operativo.

Una vez que haya instalado correctamente los componentes de Zabbix necesarios, debe crear un usuario de MySQL como se describe en la sección *Crear usuario de MySQL*.

Después de haber creado el usuario de MySQL, debe configurar Zabbix agent para que pueda establecer una conexión con el servidor MySQL y monitorizarlo. Esto incluye configurar varios parámetros de usuario para ejecutar comprobaciones personalizadas de agent, así como proporcionar a Zabbix agent las credenciales necesarias para conectarse al servidor MySQL como el usuario ”zbx_monitor” creado previamente.

Configurar el agente de Zabbix

1. Vaya al directorio de configuraciones adicionales del agente de Zabbix.

```
cd /etc/zabbix/zabbix_agentd.d
```

> **Attention:**
> El directorio de configuraciones adicionales del agente de Zabbix debe encontrarse en el mismo directorio que su archivo de configuración del agente de Zabbix; la ruta predeterminada depende del sistema operativo y del método de instalación de Zabbix.

En lugar de definir todos los parámetros de usuario necesarios para supervisar el server MySQL en el archivo de configuración del agente de Zabbix, estos parámetros se definirán en un archivo independiente dentro del directorio de configuraciones adicionales.

2. Cree un archivo `template_db_mysql.conf` en el directorio de configuraciones adicionales del agente de Zabbix.

```
vi template_db_mysql.conf
```

3. Copie el contenido del archivo `template_db_mysql.conf` (ubicado en el repositorio de Zabbix) en el archivo `template_db_mysql.conf` que creó y guárdelo.

4. Reinicie el agente de Zabbix para actualizar su configuración.

```
systemctl restart zabbix-agent
```

Una vez configurados los parámetros de usuario del agente de Zabbix, puede continuar con la configuración de las credenciales que permitirán al agente de Zabbix acceder al server MySQL.

5. Navegue al directorio de inicio del agent de Zabbix (si no existe en su sistema, debe crearlo; valor predeterminado: */var/lib/zabbix*).

```
cd /var/lib/zabbix
```

6. Cree un archivo *.my.cnf* en el directorio de inicio del agent de Zabbix.

```
vi .my.cnf
```

7. Copie el siguiente contenido en el archivo *.my.cnf* (reemplace *<password>* por la contraseña del usuario "zbx_monitor").

```
[client]
user='zbx_monitor'
password='<password>'
```

Configure Zabbix frontend y pruebe su configuración

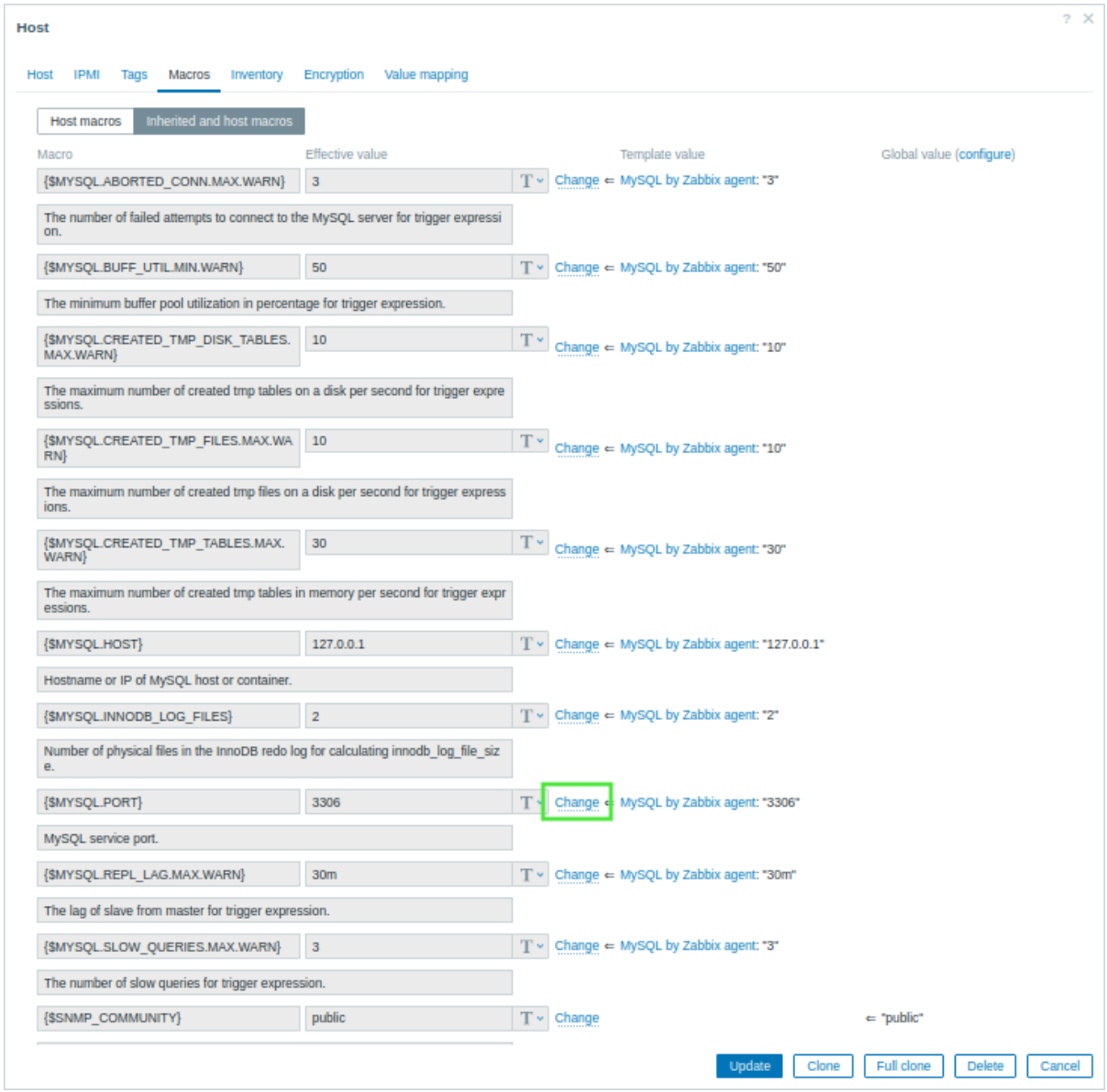
Para configurar Zabbix frontend, siga las instrucciones de la sección *Configure Zabbix frontend* con los siguientes ajustes:

- En el campo *Templates*, escriba o seleccione el template "MySQL by Zabbix agent" que se vinculará al host.
- No es necesario configurar *Macros*.

Una vez que haya configurado Zabbix frontend, puede ver las métricas recopiladas y configurar alertas de problemas.

Para probar su configuración, siga las instrucciones de la sección *Test your configuration* con los siguientes ajustes:

- En la sección *Inherited and host macros* de la configuración del host del servidor MySQL, haga clic en *Change* junto al valor de la macro {$MYSQL.PORT} y establezca un puerto diferente (por ejemplo, "6033").



Monitor MySQL con ODBC

Para monitorizar un server MySQL con ODBC, necesita descargar e instalar Zabbix server y Zabbix frontend.

Una vez que haya instalado correctamente los componentes de Zabbix requeridos, debe crear un usuario de MySQL como se describe en la sección *Crear usuario de MySQL*.

Después de haber creado el usuario de MySQL, debe configurar ODBC. Esto incluye instalar una de las implementaciones de la API ODBC de código abierto más utilizadas: unixODBC - y un controlador unixODBC, así como editar el archivo de configuración del controlador ODBC.

Configurar ODBC

1. Instale unixODBC. La forma recomendada de instalar unixODBC es usar los repositorios de paquetes predeterminados del sistema operativo Linux.

```
apt install unixodbc
```

2. Instale el controlador de base de datos MariaDB unixODBC. Aunque tenga una base de datos MySQL, se usa el controlador MariaDB unixODBC por motivos de compatibilidad.

```
apt install odbc-mariadb
```

3. Compruebe la ubicación de los archivos de configuración de ODBC *odbcinst.ini* y *odbc.ini*.

```
odbcinst -j
```

El resultado de ejecutar este comando debería ser similar al siguiente.

```
unixODBC 2.3.9
DRIVERS............: /etc/odbcinst.ini
SYSTEM DATA SOURCES: /etc/odbc.ini
FILE DATA SOURCES..: /etc/ODBCDataSources
...
```

4. Para configurar el controlador ODBC para supervisar una base de datos MySQL, necesita el nombre del controlador, que se encuentra en el archivo *odbcinst.ini*. En el siguiente ejemplo de archivo *odbcinst.ini*, el nombre del controlador es "MariaDB Unicode".

```
[MariaDB Unicode]
Driver=libmaodbc.so
Description=MariaDB Connector/ODBC(Unicode)
Threading=0
UsageCount=1
```

5. Copie el siguiente contenido en el archivo *odbc.ini* (reemplace <*password*> por la contraseña del usuario "zbx_monitor"). Esta guía usa "127.0.0.1" (localhost) como dirección del server MySQL para supervisar un server MySQL instalado en la misma máquina que el controlador ODBC. Tenga en cuenta el nombre de origen de datos (DSN) "test", que será necesario al configurar el frontend de Zabbix.

```
[test]
Driver=MariaDB Unicode
Server=127.0.0.1
User=zbx_monitor
Password=<password>
Port=3306
Database=zabbix
```

Configure Zabbix frontend y pruebe su configuración

Para configurar Zabbix frontend, siga las instrucciones de la sección *Configure Zabbix frontend* con los siguientes ajustes:

- En el campo *Templates*, escriba o seleccione el template "MySQL by ODBC" que se vinculará al host.
- No es necesario configurar *Interfaces*.
- El valor de la macro {$MYSQL.DSN} en la sección *Inherited and host macros* de la configuración del host del servidor MySQL debe establecerse en el nombre DSN del archivo *odbc.ini*.

Una vez que haya configurado Zabbix frontend, puede ver las métricas recopiladas, configurar alertas de problemas y probar su configuración.

Ver también

- Creación de un item - cómo empezar a supervisar métricas adicionales.
- Escalaciones de problemas - cómo crear escenarios de alertas de varios pasos (por ejemplo, primero enviar un mensaje al administrador del sistema y luego, si un problema no se resuelve en 45 minutos, enviar un mensaje al responsable del centro de datos).

- Supervisión ODBC - cómo configurar ODBC en otras distribuciones de Linux y cómo empezar a supervisar métricas adicionales relacionadas con bases de datos con ODBC.
- template *MySQL by Zabbix agent* - información adicional sobre el template *MySQL by Zabbix agent*.
- template *MySQL by Zabbix agent 2* - información adicional sobre el template *MySQL by Zabbix agent 2*.
- template *MySQL by ODBC* - información adicional sobre el template *MySQL by ODBC*.

Monitorizar VMware con Zabbix

Introducción

Esta página le guía a través de los pasos necesarios para comenzar la monitorización básica de VMware.

Para quién está dirigida esta guía

Esta guía está diseñada para nuevos usuarios de Zabbix y contiene el conjunto mínimo de pasos necesarios para habilitar la supervisión básica de VMware. Si busca opciones de personalización avanzadas o necesita una configuración más compleja, consulte la sección Virtual machine monitoring o la sección Configuration del manual de Zabbix.

Requisitos previos

Antes de continuar con esta guía, debe descargar e instalar Zabbix server y Zabbix frontend siguiendo las instrucciones correspondientes a su sistema operativo.

Según su entorno, algunos pasos de esta guía pueden diferir ligeramente. Esta guía se basa en un entorno que ejecuta Ubuntu.

Se asume que VMware ya está configurado. Esta guía no cubre la configuración de VMware.

Configurar el server de Zabbix

Para supervisar VMware, es necesario habilitar los procesos de Zabbix *vmware collector*. Para obtener más información sobre cómo se realiza la supervisión de VMware, consulte Supervisión de máquinas virtuales.

1. Abra el archivo de configuración del server de Zabbix.

```
vi /etc/zabbix/zabbix_server.conf
```

2. Busque y establezca el parámetro `StartVMwareCollectors` en el archivo de configuración del server de Zabbix en 2 o más (el valor predeterminado es 0).

```
###### Option: StartVMwareCollectors
#####       Number of pre-forked vmware collector instances.
#####
##### Mandatory: no
##### Range: 0-250
##### Default:
##### StartVMwareCollectors=0

StartVMwareCollectors=2
```

3. Reinicie el server de Zabbix.

```
systemctl restart zabbix-server
```

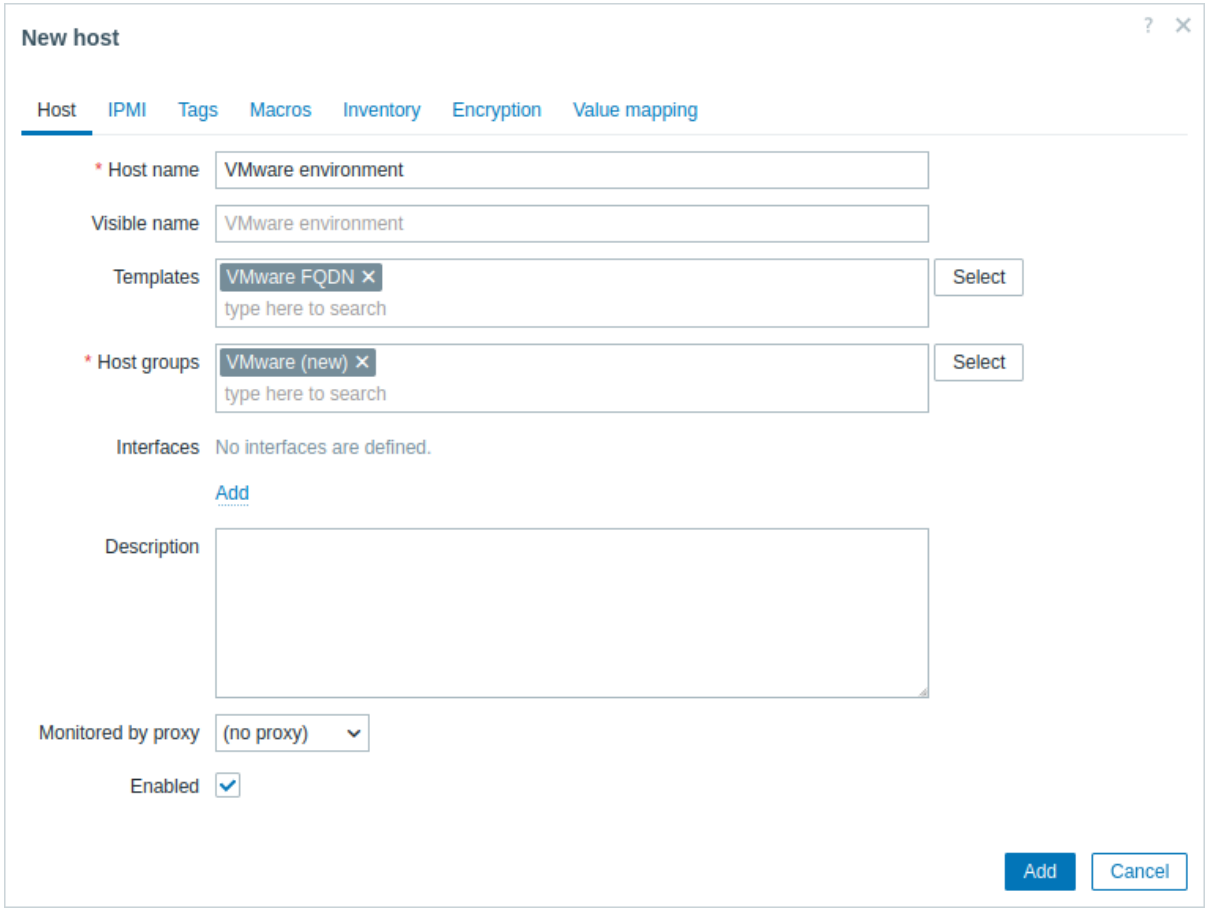
Una vez iniciados los procesos de *vmware collector*, continúe con el siguiente paso.

Configurar frontend de Zabbix

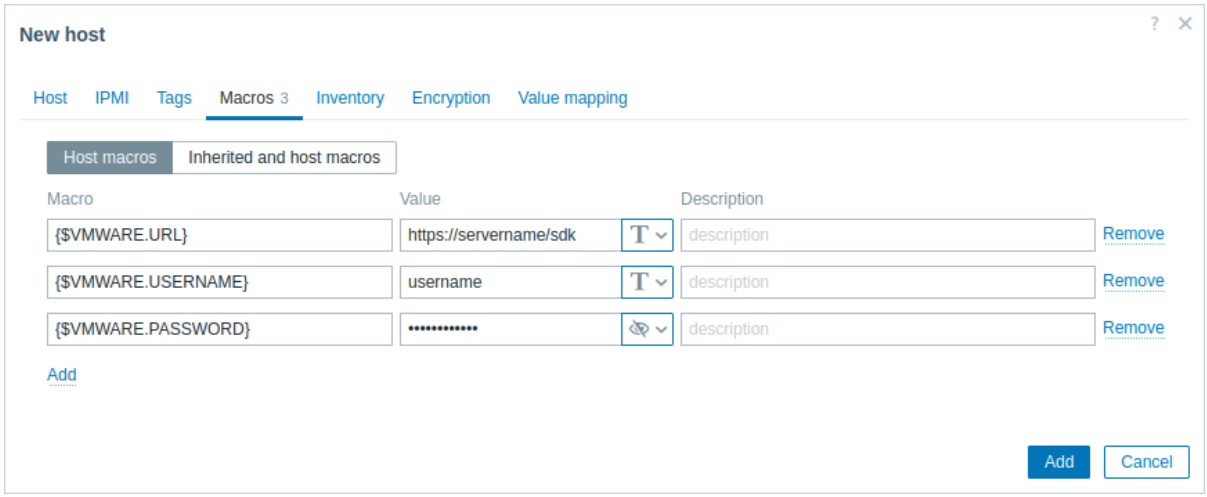
1. Inicie sesión en el frontend de Zabbix.

2. Cree un host en la interfaz web de Zabbix:

- En el campo *Host name*, introduzca un nombre de host (por ejemplo, ”VMware environment”).
- En el campo *Templates*, escriba o seleccione la template ”VMware FQDN” (o ”VMware”). Para obtener más información sobre estas templates, consulte Virtual machine monitoring.
- En el campo *Host groups*, escriba o seleccione un grupo de hosts (por ejemplo, un nuevo grupo de hosts ”VMware”).



- En la pestaña *Macros*, configure las siguientes macros del host:
  - {$VMWARE.URL} - URL SDK del servicio VMware (vCenter o hipervisor ESXi) (https://servername/sdk)
  - {$VMWARE.USERNAME} - nombre de usuario del servicio VMware
  - {$VMWARE.PASSWORD} - contraseña del usuario {$VMWARE.USERNAME} del servicio VMware



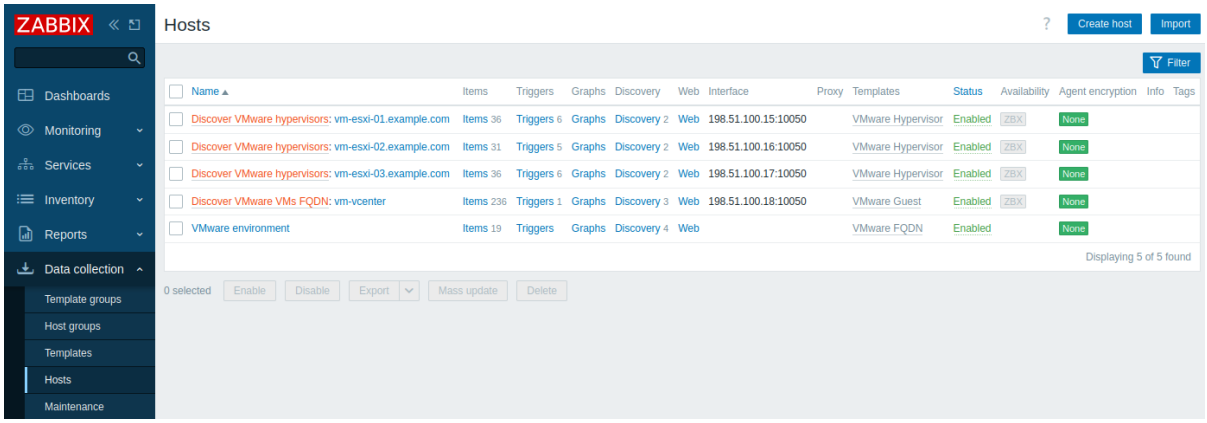
3. Haga clic en el botón *Add* para crear el host. Este host representará su entorno VMware.

Ver métricas recopiladas

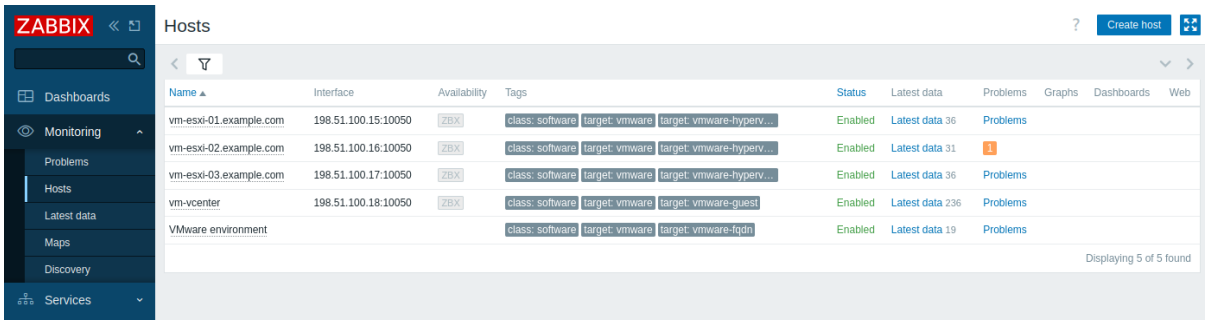
¡Felicidades! En este punto, Zabbix ya está supervisando su entorno VMware.

Según la configuración de su entorno VMware, Zabbix puede descubrir y luego crear hosts para las entidades descubiertas. Tenga en cuenta que el descubrimiento y la creación de hosts también pueden ejecutarse manualmente, si es necesario.

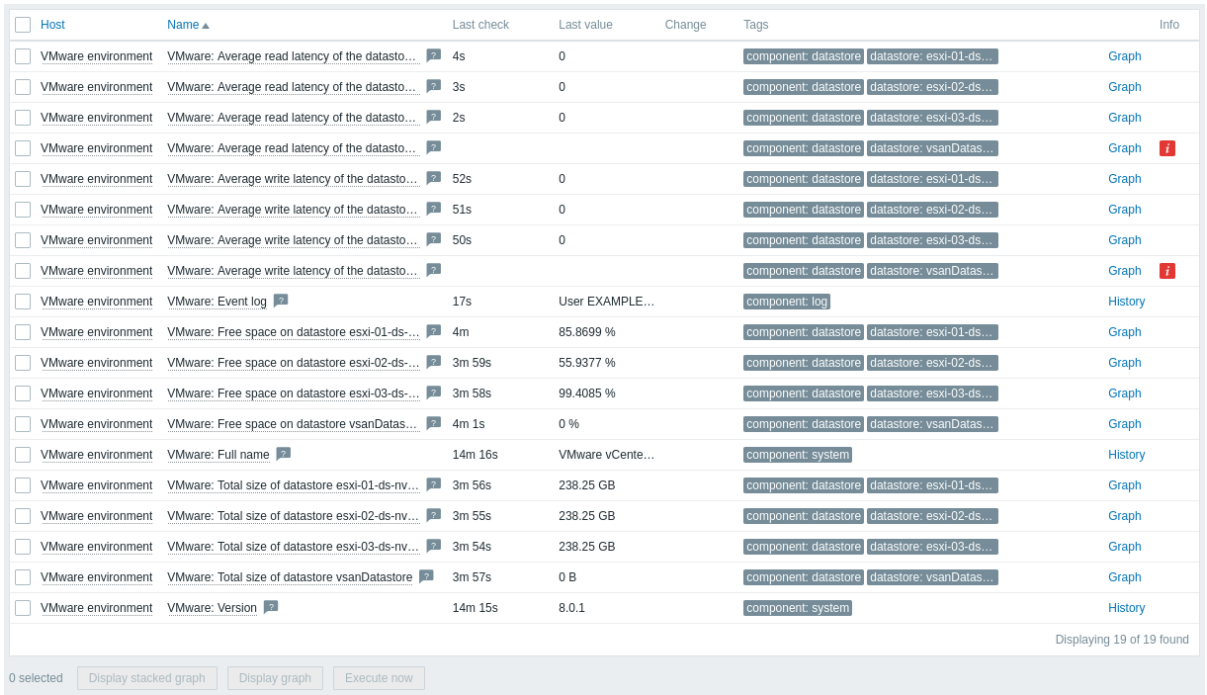
Para ver los hosts creados, vaya a la sección de menú *Data collection > Hosts*.



Para ver las métricas recopiladas, vaya a la sección de menú *Monitoring > Hosts* y haga clic en *Latest data* junto al host "VMware environment" creado o a uno de los hosts que se crearon para las entidades descubiertas.



Esta acción abrirá una lista de todas las métricas más recientes recopiladas del host seleccionado.

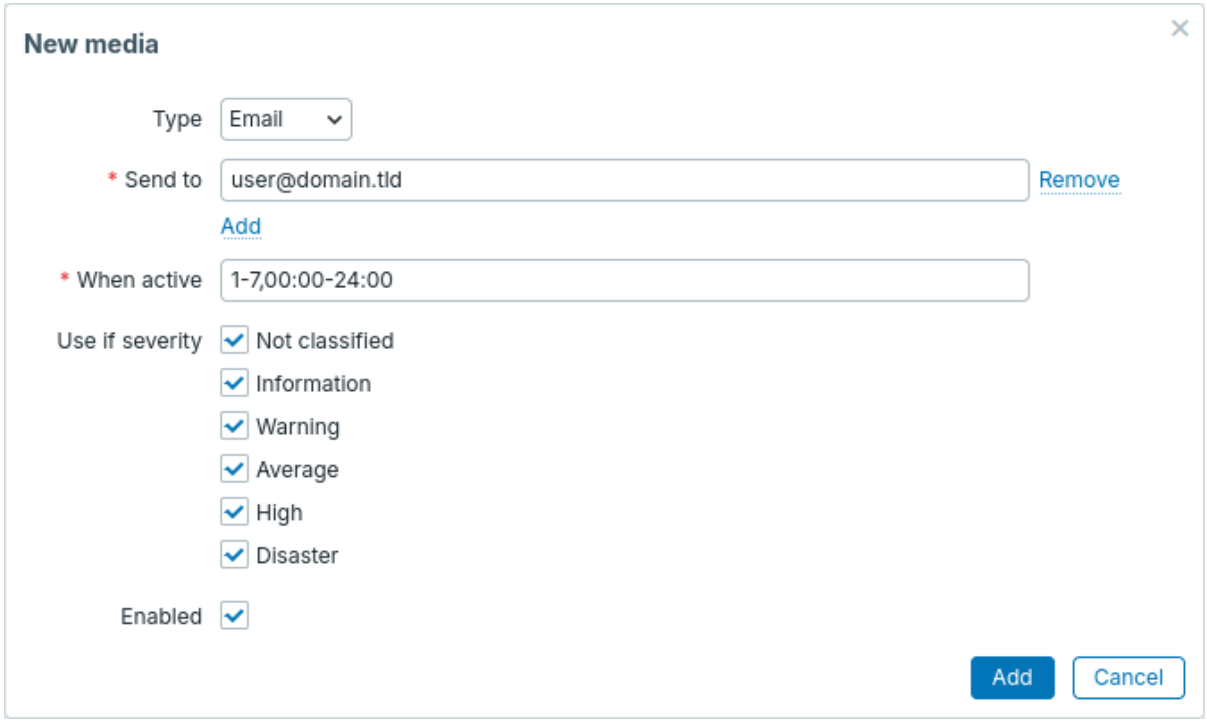


Tenga en cuenta que algunos items no tienen datos y presentan el estado *Not supported*. Esto se debe a que Zabbix no puede encontrar contadores de rendimiento válidos en el datastore específico, ya que no está habilitado en el entorno VMware que se está supervisando.

Configurar alertas de problemas

Zabbix puede notificarle sobre un problema con su infraestructura mediante varios métodos. Esta guía proporciona pasos básicos de configuración para enviar alertas por correo electrónico.

1. Vaya a *User settings > Profile*, cambie a la pestaña *Media* y añada su correo electrónico.



2. Siga la guía para Receiving a problem notification.

La próxima vez que Zabbix detecte un problema, debería recibir una alerta por correo electrónico.

Véase también

- Creación de un item - cómo empezar a supervisar métricas adicionales.
- Escalaciones de problemas - cómo crear escenarios de alertas de varios pasos (por ejemplo, primero enviar un mensaje al administrador del sistema y luego, si un problema no se resuelve en 45 minutos, enviar un mensaje al responsable del centro de datos).
- Supervisión de máquinas virtuales - información adicional sobre la supervisión de VMware (proceso de recopilación de datos, opciones de configuración del server, guía de solución de problemas, etc.).
- Claves de item para la supervisión de VMware - una lista completa de métricas de VMware que se pueden supervisar usando Zabbix.
- Template *VMware* - información adicional sobre el template *VMware*.
- Template *VMware FQDN* - información adicional sobre el template *VMware FQDN*.

Monitorizar el tráfico de red con Zabbix

Introducción

Esta página le guía a través de los pasos necesarios para comenzar la monitorización básica del tráfico de su red con Zabbix.

Para quién está dirigida esta guía

Esta guía está diseñada para nuevos usuarios de Zabbix y contiene el conjunto mínimo de pasos necesarios para habilitar la supervisión básica del tráfico de su red. Si busca opciones de personalización avanzadas o necesita una configuración más compleja, consulte la sección Configuration del manual de Zabbix.

Requisitos previos

Antes de continuar con esta guía, debe descargar e instalar Zabbix server, Zabbix frontend y Zabbix agent según las instrucciones para su sistema operativo.
Tenga en cuenta que debe instalar Zabbix agent en la máquina que requiere la supervisión del tráfico.
Puede ser el mismo host donde está instalado Zabbix server o un host diferente.

Esta guía proporcionará instrucciones para configurar la supervisión del tráfico de red de la interfaz *eth0* en una máquina separada llamada *Remote host*.

Configurar Zabbix para la supervisión

El agent de Zabbix puede recopilar métricas en modo activo o pasivo (simultáneamente). Consulte Passive and active agent checks para obtener más información. En esta guía, se describirá la supervisión mediante comprobaciones pasivas.

Configuración del agent de Zabbix

1. Abra el archivo de configuración del agent en la máquina donde está instalado el agent:

```
sudo vi /etc/zabbix/zabbix_agentd.conf
```

2. Añada la dirección IP o el nombre DNS de su server de Zabbix al parámetro *Server*. Por ejemplo:

```
Server=192.0.2.22
```

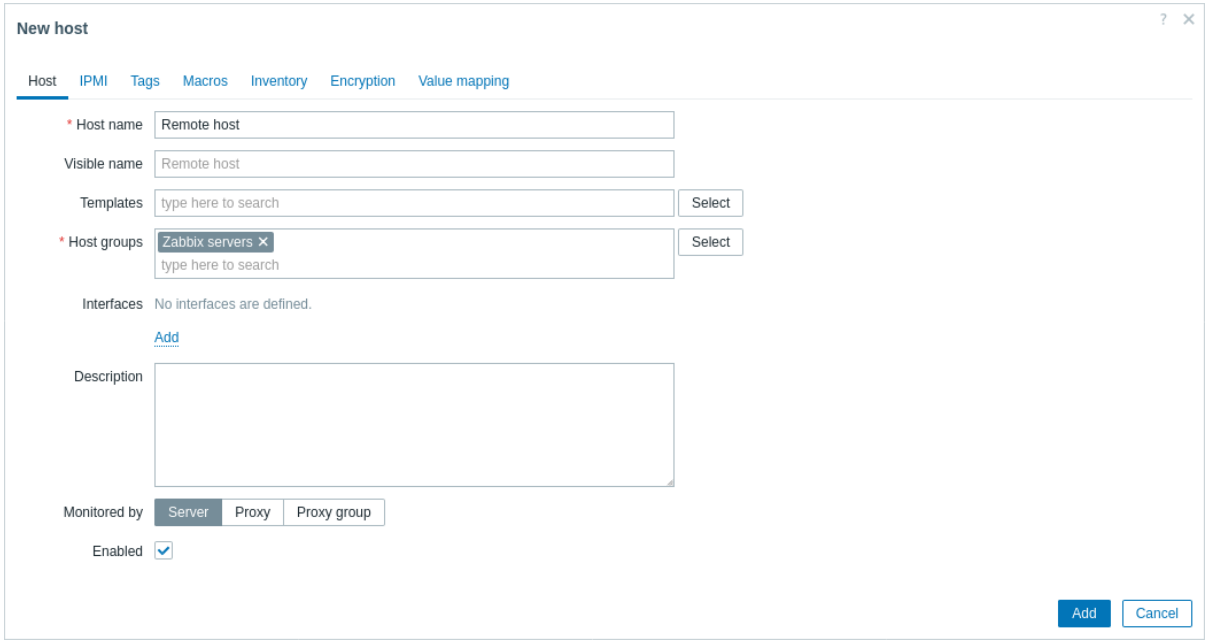
3. Reinicie el agent de Zabbix:

```
systemctl restart zabbix-agent
```

Frontend de Zabbix

1. Inicie sesión en el frontend de Zabbix.

2. Cree un host en la interfaz web de Zabbix, especificando la dirección IP o el nombre DNS de la máquina en la que está instalado el agent.

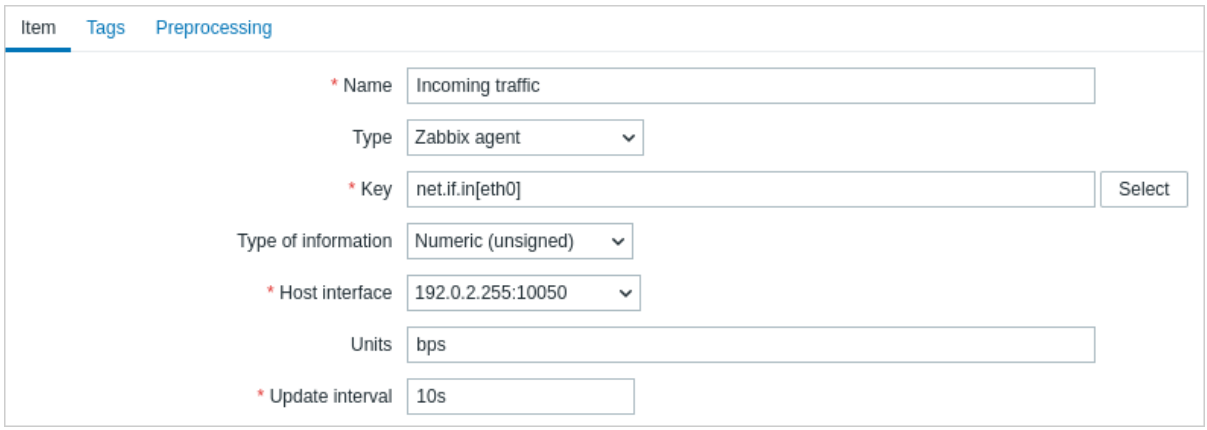


Crear items

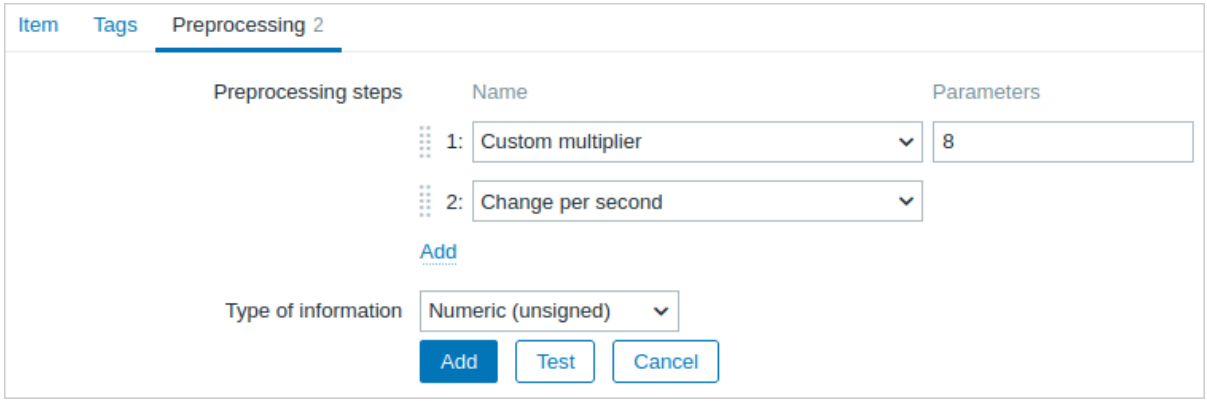
Siga las instrucciones sobre crear un item para añadir los items para la supervisión del tráfico, concretamente:

- Tráfico entrante
- Tráfico saliente
- Tráfico total

Una configuración sencilla para el item de supervisión del tráfico entrante tendría el siguiente aspecto:



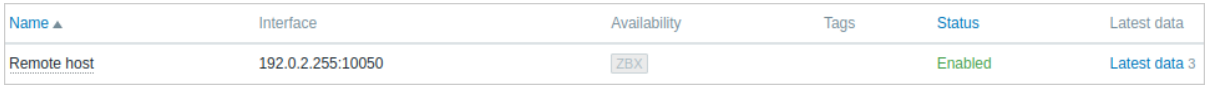
Para que los datos recopilados sean adecuados para un uso práctico, puede definir algunos pasos de preprocesamiento al crear los items. En este caso, pueden ser la multiplicación por 8 (para convertir bytes en bits) y la presentación como cambio por segundo.



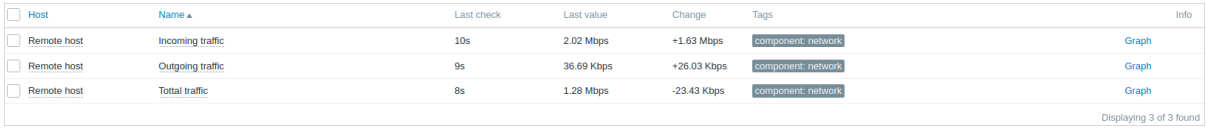
Ver datos recopilados

¡Felicidades! En este punto, Zabbix ya está supervisando el tráfico de su red.

Para ver las métricas recopiladas, abra la sección del menú *Monitoring > Hosts* y haga clic en *Latest data* en la fila del host.

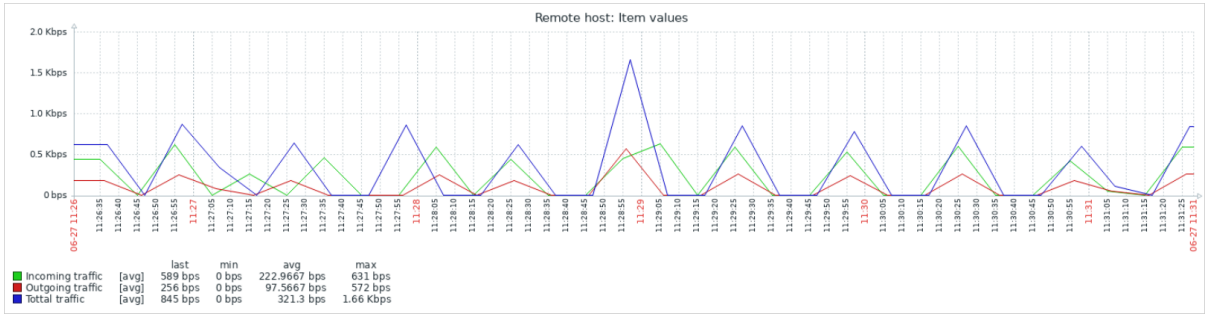


Verá los datos del tráfico.



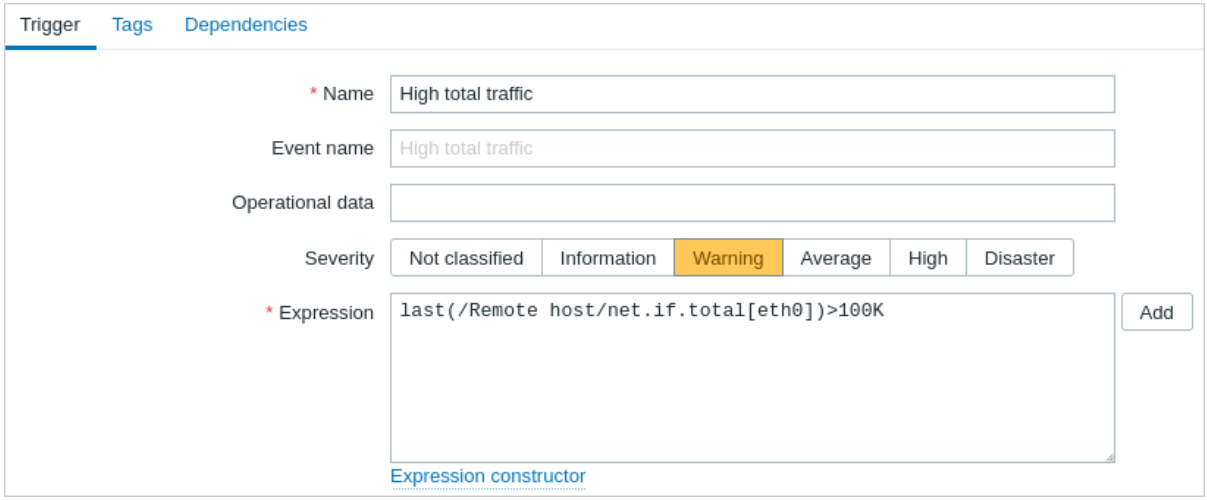
Ver gráficos

Los datos recopilados se pueden mostrar como gráficos. Para verlos, en la sección *Latest data* haga clic en *Graph* en la fila del item o seleccione los items necesarios y haga clic en *Display graph* debajo.

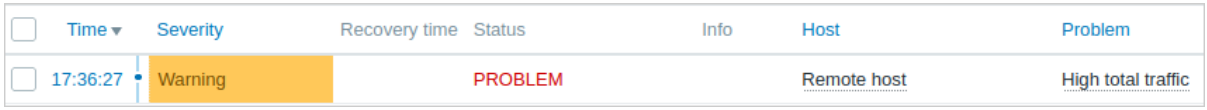


Configurar triggers

Puede configurar triggers para detectar tráfico de red anómalo. Consulte las instrucciones sobre cómo configurar un trigger y añada el trigger que indique que el tráfico total es demasiado alto, por ejemplo:



Ahora haga que el tráfico supere el umbral que ha establecido en la expresión del trigger y vaya a *Monitoring > Problems* para comprobar que el problema aparece allí.



Configurar alertas de problemas

Hay varias formas de recibir notificaciones sobre el problema. El correo electrónico es la más popular; siga las instrucciones para configurar una notificación de problema por correo electrónico. También puede elegir otros tipos de medios para utilizarlos en la entrega de notificaciones.

Véase también:

- Escalaciones de problemas - cómo crear escenarios de alertas de varios pasos (p. ej., primero enviar un mensaje al administrador del sistema y, si un problema no se resuelve en 45 minutos, enviar un mensaje al responsable del centro de datos).
- Reconocimiento de problemas - cómo indicar que el problema es conocido, hacer comentarios sobre su resolución, suprimir o cerrar el problema.
- Monitorizar Linux con Zabbix agent - cómo iniciar la monitorización básica de los items más importantes vinculándolos a un template preconfigurado.

Supervisar el tráfico de red mediante comprobaciones activas

Introducción

Esta página le guía a través de los pasos necesarios para comenzar la monitorización básica del tráfico de su red con Zabbix mediante comprobaciones activas.

Para quién está dirigida esta guía

Esta guía está diseñada para nuevos usuarios de Zabbix y contiene el conjunto mínimo de pasos necesarios para habilitar la supervisión básica del tráfico de su red mediante comprobaciones activas. Si busca opciones de personalización avanzadas o necesita una configuración más compleja, consulte la sección Configuration del manual de Zabbix.

Requisitos previos

Antes de continuar con esta guía, debe descargar e instalar Zabbix server, Zabbix frontend y Zabbix agent según las instrucciones para su sistema operativo.
Tenga en cuenta que debe instalar Zabbix agent en la máquina que requiere supervisión del tráfico.
Puede ser el mismo host donde está instalado Zabbix server o un host diferente.

Esta guía proporcionará instrucciones para configurar la supervisión del tráfico de red de la interfaz *eth0*
en una máquina separada llamada *Remote host*.

Configurar Zabbix para la supervisión

El agent de Zabbix puede recopilar métricas en modo activo o pasivo (simultáneamente). Consulte Passive and active agent checks para obtener más información. En esta guía, se describirá la supervisión mediante **active checks**.

Configuración de Zabbix agent

1. Abra el archivo de configuración del agent en la máquina donde está instalado el agent.

Si utiliza Zabbix agent:

```
sudo vi /etc/zabbix/zabbix_agentd.conf
```

Si utiliza Zabbix agent 2:

```
sudo vi /etc/zabbix/zabbix_agent2.conf
```

2. Añada la dirección IP o el nombre DNS (y el puerto opcional) de su Zabbix server al parámetro ServerActive. Por ejemplo:

```
ServerActive=192.0.2.1:10051
```

Zabbix agent utilizará esta dirección para conectarse al puerto trapper de Zabbix server (predeterminado: 10051) y solicitar los datos de configuración para las comprobaciones activas.

3. Defina el parámetro Hostname, que debe coincidir con el nombre del host que se definirá en Zabbix frontend. En nuestro ejemplo, es:

```
Hostname=Remote host
```

El valor de Hostname debe coincidir porque, para las comprobaciones activas, Zabbix agent lo utiliza para recuperar la configuración correcta del host desde el server. Específicamente, el agent inicia una conexión con el server y se identifica mediante el valor de Hostname. A continuación, el server proporciona la configuración de monitorización para ese host. Si estos valores difieren, el agent no recibirá la configuración adecuada, lo que provocará la ausencia de métricas o problemas de monitorización.

4. Reinicie Zabbix agent.

Si utiliza Zabbix agent:

```
systemctl restart zabbix-agent
```

Si utiliza Zabbix agent 2:

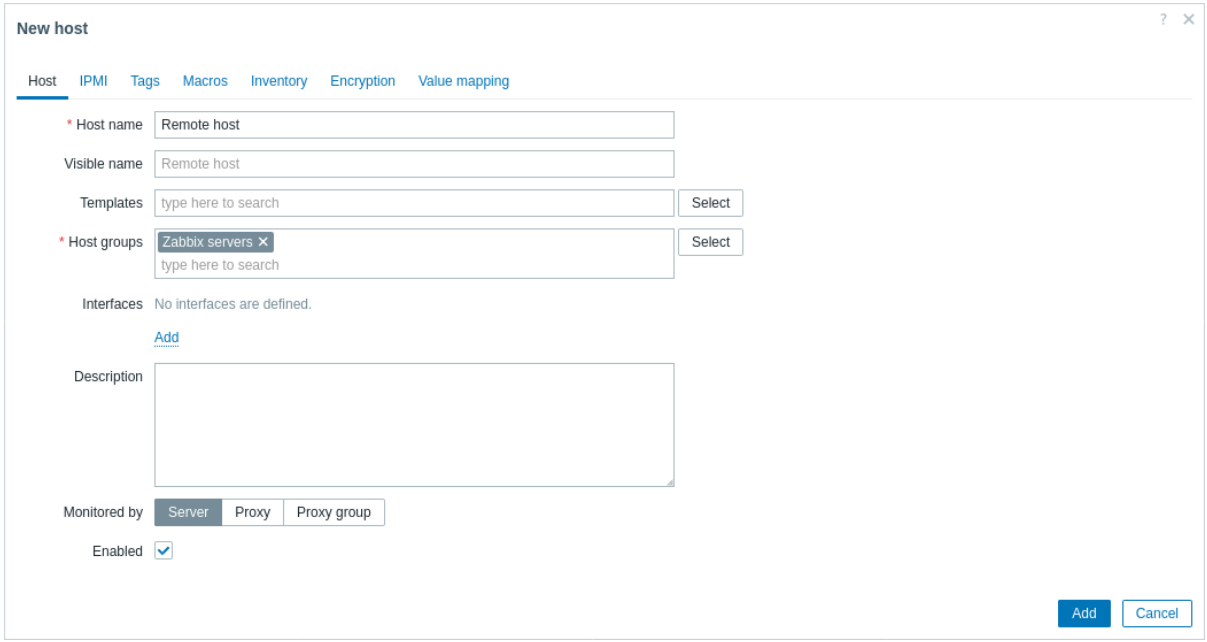
```
systemctl restart zabbix-agent2
```

frontend de Zabbix

1. Inicie sesión en el frontend de Zabbix.

2. Cree un host en la interfaz web de Zabbix.

- En el campo *Host name*, introduzca un nombre de host (por ejemplo, ”Remote host”) que coincida con el valor del parámetro Hostname definido anteriormente en el archivo de configuración del agent.
- En el campo *Host groups*, escriba o seleccione un grupo de hosts (por ejemplo, ”Zabbix servers”).
- Deje *Interfaces* sin definir; no es necesaria una interfaz para las comprobaciones activas porque el agent inicia la conexión al server en lugar de escuchar una conexión desde el server.



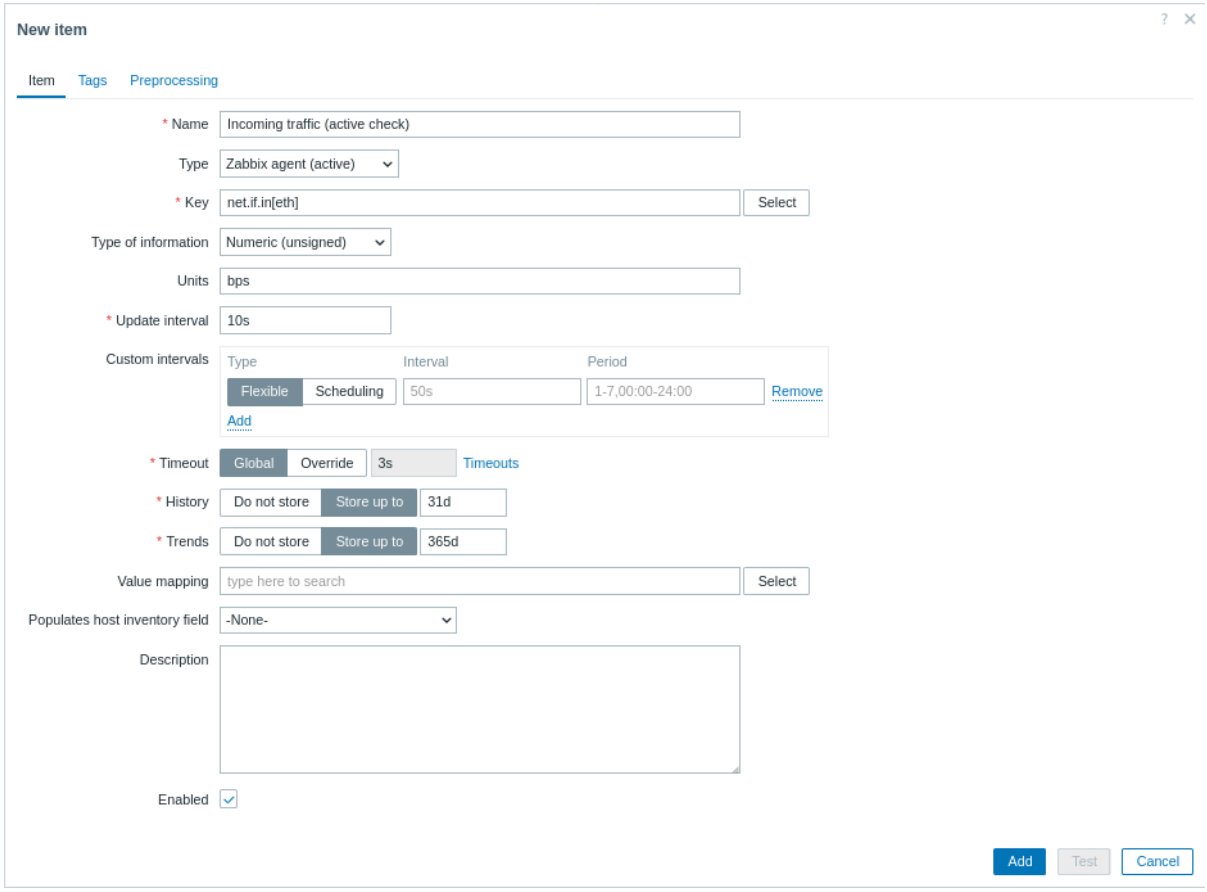
3. Haga clic en *Add* para añadir el host. Este host representará la máquina Linux supervisada.

Crear items

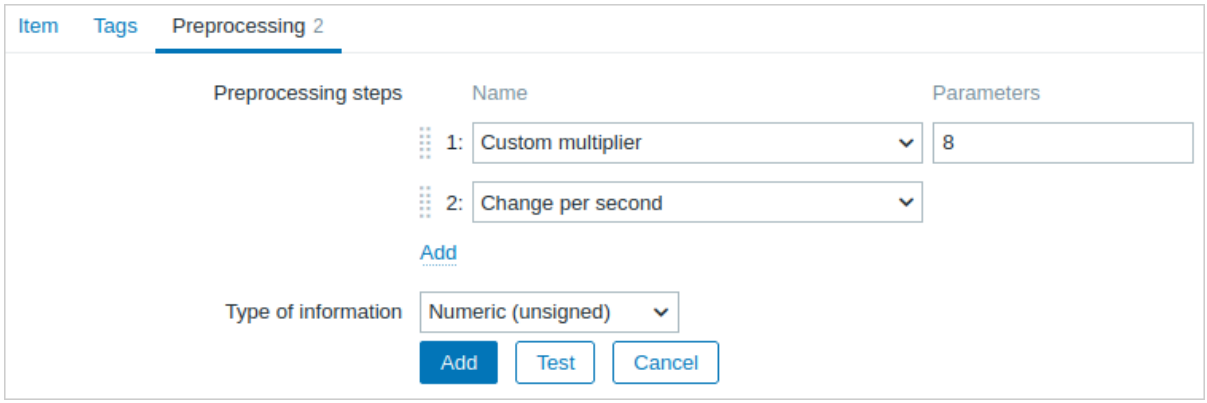
Siga las instrucciones sobre crear un item para añadir los items para la supervisión del tráfico, a saber:

- Tráfico entrante
- Tráfico saliente
- Tráfico total

Una configuración sencilla para el item de supervisión del tráfico entrante con una comprobación activa se vería así:



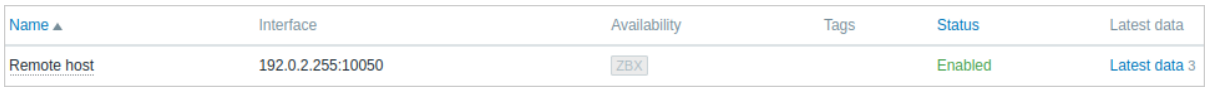
Para que los datos recopilados sean adecuados para un uso práctico, puede establecer algunos pasos de preprocesamiento al crear los items. En este caso, pueden ser la multiplicación por 8 (para convertir bytes en bits) y la presentación como cambio por segundo.



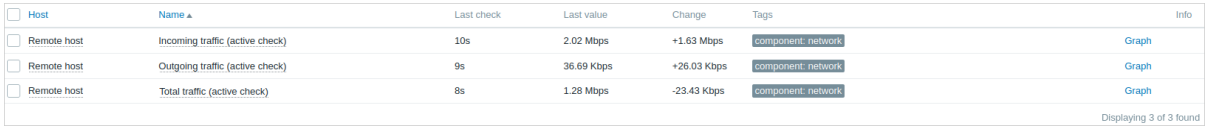
Ver datos recopilados

¡Enhorabuena! En este punto, Zabbix ya está supervisando el tráfico de su red.

Para ver las métricas recopiladas, abra la sección de menú *Monitoring > Hosts* y haga clic en *Latest data* en la fila del host.



Verá los datos de tráfico.



Ver también:

- Ver gráficos - cómo mostrar los datos recopilados como gráficos.
- Configurar triggers - cómo configurar triggers para detectar tráfico de red anómalo.
- Configurar alertas de problemas - cómo configurar notificaciones sobre situaciones problemáticas.

- Escalaciones de problemas - cómo crear escenarios de alertas de varios pasos (por ejemplo, primero enviar un mensaje al administrador del sistema y luego, si un problema no se resuelve en 45 minutos, enviar un mensaje al responsable del centro de datos).
- Reconocimiento de problemas - cómo indicar que el problema es conocido, hacer comentarios sobre su resolución, suprimir o cerrar el problema.
- Monitorizar Linux con Zabbix agent - cómo iniciar la monitorización básica de los items más importantes vinculándolos a un template preconfigurado.
- Instalación desde paquetes - cómo instalar los componentes de Zabbix usando los paquetes oficiales RPM y DEB para diversas distribuciones de Linux, garantizando acceso a las funciones más recientes y a las correcciones de errores.

Supervisar sitios web con items de navegador

Introducción

Esta página le guía a través de los pasos necesarios para comenzar la monitorización básica de sitios web con Browser items.

Para quién está dirigida esta guía

Esta guía está diseñada para nuevos usuarios de Zabbix y contiene el conjunto mínimo de pasos necesarios para habilitar la supervisión básica de sitios web con Browser items. Si busca opciones de personalización avanzadas o requiere una configuración más compleja, consulte la página Browser items o la sección Configuration del manual de Zabbix.

Requisitos previos

Antes de continuar con esta guía, debe descargar e instalar Zabbix server y Zabbix frontend según las instrucciones para su sistema operativo.

Según su entorno, algunos pasos de esta guía pueden diferir ligeramente. Esta guía se basa en un entorno que ejecuta Ubuntu.

Configurar WebDriver

Los items de navegador requieren un framework de automatización (ya sea Selenium Server o un WebDriver simple, por ejemplo, ChromeDriver) como un punto de conexión de pruebas web que controla e interactúa con un navegador, ejecutando comandos de prueba como hacer clic en botones o introducir texto. Como ejemplo, esta guía utilizará Selenium Server con Chrome en un contenedor Docker.

Se asume que Docker ya está configurado. Esta guía no cubre la configuración de Docker. Para obtener instrucciones de instalación, consulte Install Docker Engine on Ubuntu.

1. Inicie Selenium Server con Chrome en un contenedor Docker con las siguientes opciones:

- **docker run --name browser** - ejecuta un nuevo contenedor Docker llamado "browser";
- **-p 4444:4444** - asigna el puerto 4444 de su máquina host al puerto 4444 del contenedor (este es el puerto que usa Selenium Server para aceptar comandos);
- **-p 7900:7900** - asigna el puerto 7900 de su máquina host al puerto 7900 del contenedor (este es el puerto que usa el servidor de Virtual Network Computing (VNC), lo que le permite ver la interfaz gráfica del navegador de forma remota; requiere un cliente VNC);
- **--shm-size="2g"** - asigna 2 GB de memoria compartida al contenedor (esto es importante para que Chrome funcione correctamente, ya que puede requerir una cantidad significativa de memoria compartida para evitar fallos);
- **-d** - ejecuta el contenedor en modo detached, lo que significa que se ejecutará en segundo plano;
- **selenium/standalone-chrome:latest** - especifica la imagen Docker que se usará; en este caso, la versión más reciente de Selenium Server with Chrome.

```
docker run --name browser \
-p 4444:4444 \
-p 7900:7900 \
--shm-size="2g" \
-d selenium/standalone-chrome:latest
```

2. Asegúrese de que el contenedor Docker `browser` se esté ejecutando y sea accesible.

- Obtenga la dirección IP del contenedor (en este ejemplo, 192.0.2.1):

```
ip addr

##### 1: lo: <LOOPBACK,UP,LOWER_UP>
#####     ...
##### 3: docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> ...
#####     inet 192.0.2.1/16 brd 192.0.255.255 scope global docker0
#####     ...
```

- Pruebe la conexión al contenedor con Ncat:

```
nc -zv 192.0.2.1 4444

##### Connection to 192.0.2.1 4444 port [tcp/*] succeeded!
```

- Obtenga el contenido de la página web desde Selenium Server con curl:

```
curl -L 192.0.2.1:4444

##### <!DOCTYPE html>
##### <html lang="en">
#####
##### <head>
#####   <meta charset="utf-8"/>
#####   <link href="favicon.svg" rel="icon" type="image/svg">
#####   <meta content="width=device-width, initial-scale=1" name="viewport"/>
#####   <link href="logo192.png" rel="apple-touch-icon"/>
#####   <link href="manifest.json" rel="manifest"/>
#####   <title>Selenium Grid</title>
##### </head>
#####
##### <body>
##### ...
```

Note:
Para la resolución de problemas, consulte la documentación de Docker.

Configurar el server de Zabbix

Los items del navegador se ejecutan y procesan mediante procesos *browser poller* de Zabbix, que deben habilitarse ajustando el parámetro de configuración del server StartBrowserPollers. Además, el parámetro WebDriverURL debe especificar el endpoint de pruebas web configurado previamente.

De forma predeterminada, el parámetro StartBrowserPollers está establecido en 1, por lo que solo es necesario especificar el endpoint de pruebas web.

1. Abra el archivo de configuración del server de Zabbix:

```
vi /etc/zabbix/zabbix_server.conf
```

2. Busque y establezca el parámetro WebDriverURL en el archivo de configuración del server de Zabbix:

```
###### Option: WebDriverURL
#####WebDriver interface HTTP(S) URL. For example, http://localhost:4444 used with Selenium WebDriver star
#####
##### Mandatory: no
##### Default:
##### WebDriverURL=

WebDriverURL=192.0.2.1:4444
```

3. Reinicie el server de Zabbix.

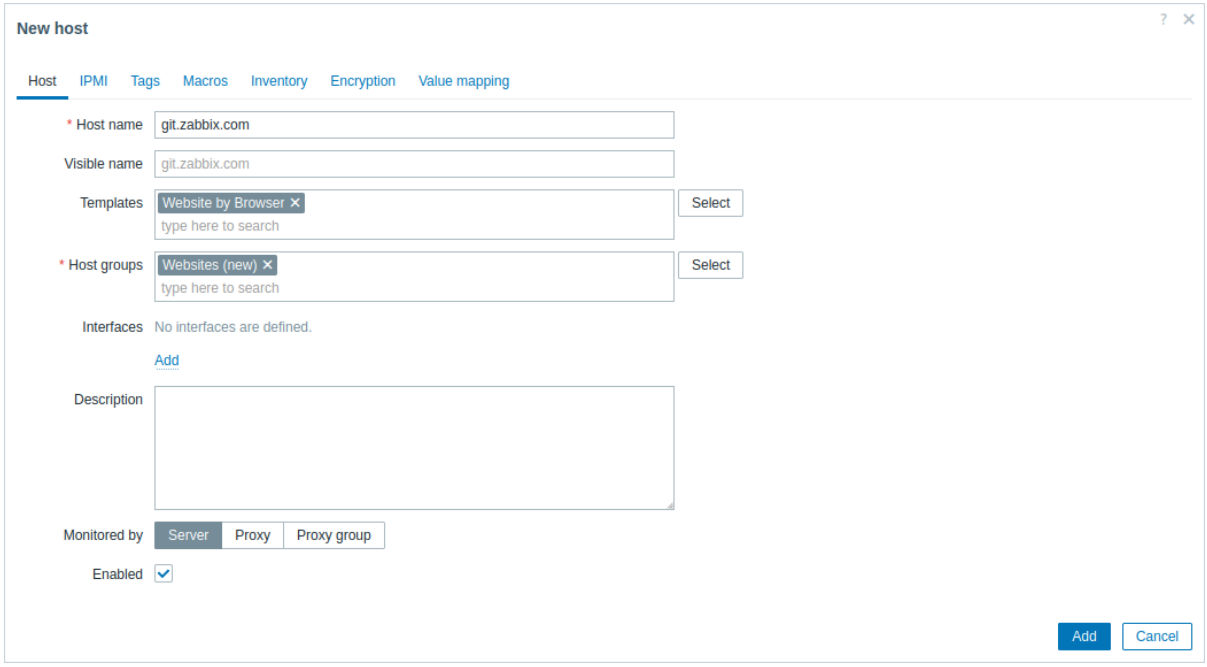
```
systemctl restart zabbix-server
```

Configurar el frontend de Zabbix

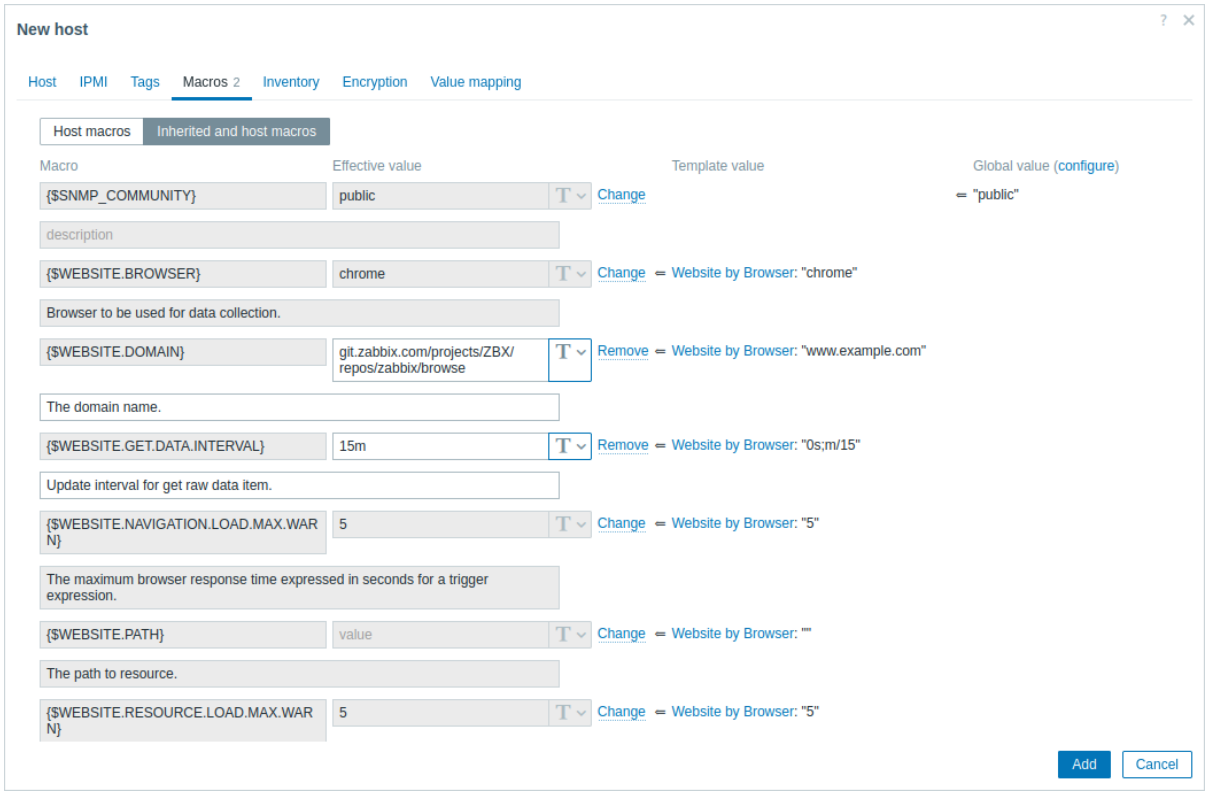
1. Inicie sesión en el frontend de Zabbix.

2. Cree un host en la interfaz web de Zabbix:

- En el campo *Host name*, introduzca un nombre de host (por ejemplo, ”git.zabbix.com”).
- En el campo *Templates*, escriba o seleccione el template ”Website by Browser”. Para obtener más información sobre este template, consulte Website by Browser.
- En el campo *Host groups*, escriba o seleccione un grupo de hosts (por ejemplo, un nuevo grupo de hosts ”Websites”).



- En la pestaña *Macros*, cambie a *Inherited and host macros*, busque las siguientes macros y haga clic en *Change* junto al valor de la macro para actualizarlo:
  - {$WEBSITE.DOMAIN} - Nombre de dominio (por ejemplo, git.zabbix.com/projects/ZBX/repos/zabbix/browse)
  - {$WEBSITE.GET.DATA.INTERVAL} - Intervalo de actualización de datos del item (por ejemplo, 15m)

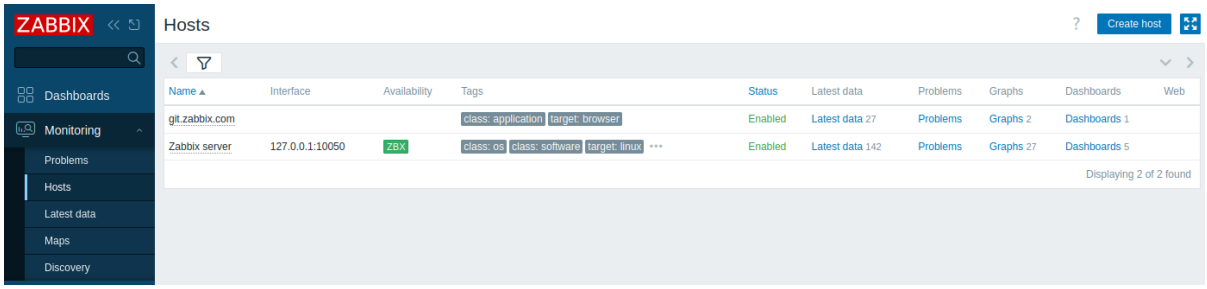


3. Haga clic en el botón *Add* para crear el host. Este host representará el sitio web que desea supervisar.

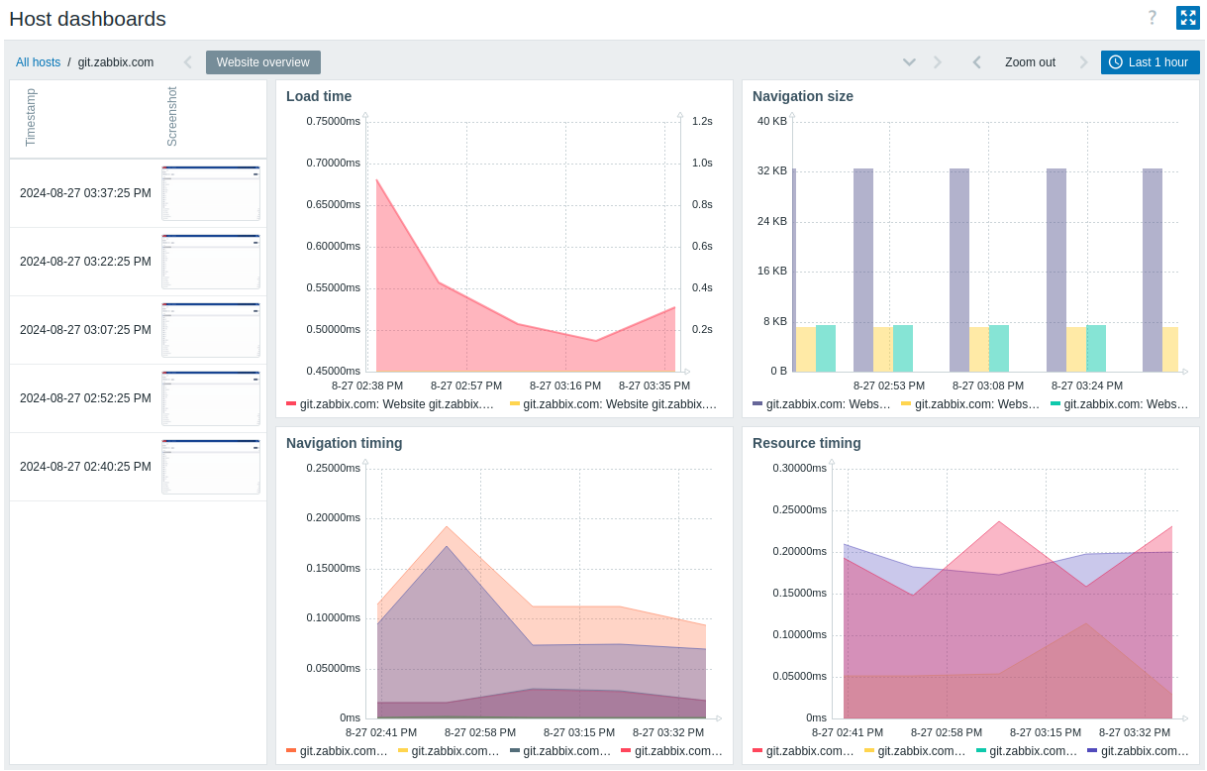
Ver métricas recopiladas

¡Enhorabuena! En este punto, Zabbix ya está supervisando el sitio web que especificó.

Para ver las métricas recopiladas, vaya a la sección de menú *Monitoring > Hosts* y haga clic en *Dashboards* junto al host.



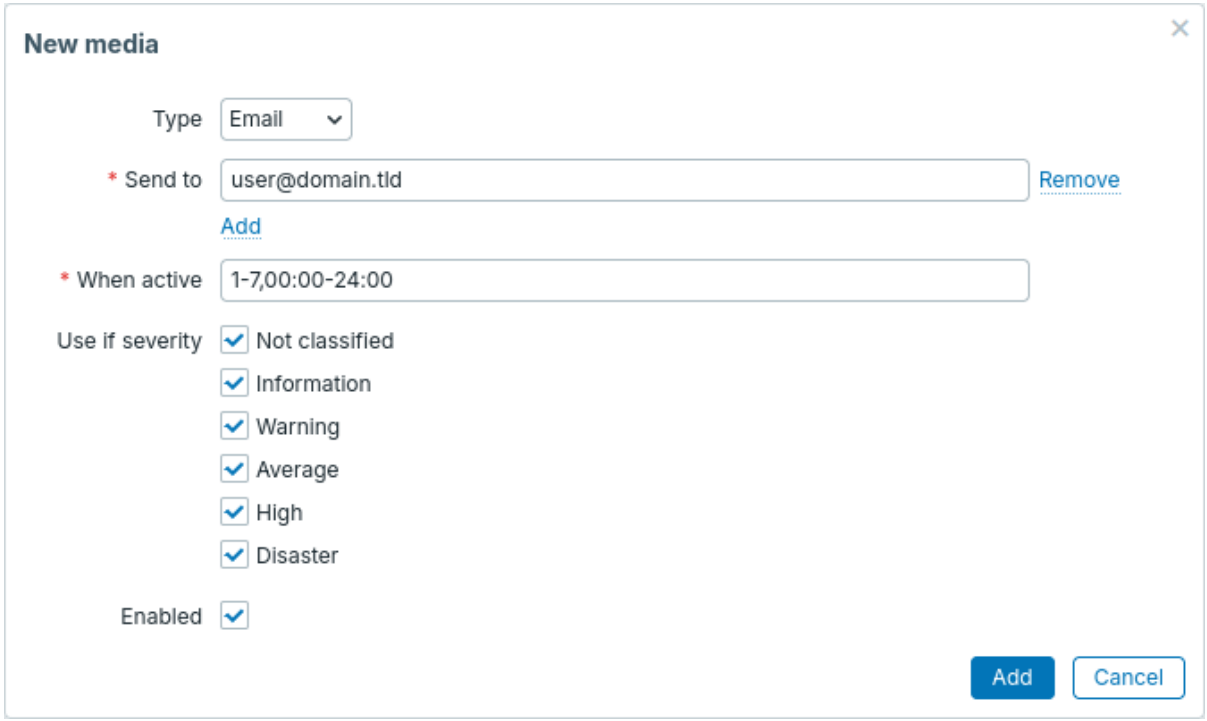
Esta acción le llevará al dashboard del host (configurado a nivel de template) con las métricas más importantes recopiladas del sitio web.



Configurar alertas de problemas

Zabbix puede notificarle sobre un problema con su infraestructura mediante varios métodos. Esta guía proporciona pasos básicos de configuración para enviar alertas por correo electrónico.

1. Vaya a *User settings > Profile*, cambie a la pestaña *Media* y añada su correo electrónico.



2. Siga la guía para Receiving a problem notification.

La próxima vez que Zabbix detecte un problema, debería recibir una alerta por correo electrónico.

Prueba tu configuración

Para probar tu configuración, podemos simular un problema real actualizando la configuración del host en el frontend de Zabbix.

1. Abre la configuración del host de tu sitio web en Zabbix.

2. Cambia a la pestaña *Macros* y selecciona *Inherited and host macros*.

3. Haz clic en *Change* junto al valor de la macro {$WEBSITE.DOMAIN} configurado anteriormente y establece un nombre de dominio incorrecto (por ejemplo, /git.zabbix.com/projects/ZBX/repos/zabbix/browse).

4. Haz clic en *Update* para actualizar la configuración del host.

5. En unos momentos, Zabbix detectará el problema "Failed to get JSON of the requested website", porque no podrá conectarse al sitio web especificado. El problema aparecerá en *Monitoring > Problems*.



Si las alertas están configuradas, también recibirás la notificación del problema.

6. Cambia el valor de la macro de nuevo a su valor anterior para resolver el problema y continuar supervisando el sitio web.

Ver también

- Creación de un item - cómo empezar a supervisar métricas adicionales.
- Escalaciones de problemas - cómo crear escenarios de alertas de varios pasos (por ejemplo, primero enviar un mensaje al administrador del sistema y luego, si un problema no se resuelve en 45 minutos, enviar un mensaje al gerente del centro de datos).
- Items de navegador - cómo configurar items de navegador.
- template Website by Browser - información adicional sobre el template *Website by Browser*.

Monitorizar certificados de sitios web con Zabbix agent 2 (pasivo)

Introducción

Esta guía proporciona una visión general completa de cómo configurar y supervisar certificados SSL/TLS utilizando la clave `web.certificate.get` en Zabbix agent 2. Está diseñada para simplificar la supervisión de certificados para uno o varios sitios web, garantizando que los administradores puedan identificar rápidamente posibles problemas, como certificados caducados o no válidos.

Para quién está dirigida esta guía

Esta guía está diseñada para nuevos usuarios de Zabbix y contiene el conjunto mínimo de pasos necesarios para habilitar la supervisión básica de certificados de sitios web. Si busca opciones de personalización avanzadas o requiere una configuración más compleja, consulte la sección Configuration del manual de Zabbix.

Prerrequisitos

Antes de continuar con esta guía, debe descargar e instalar Zabbix server, Zabbix frontend y Zabbix agent 2 según las instrucciones para su sistema operativo. Este tutorial supone que tanto el Zabbix server como el agent están instalados en la misma máquina; por lo tanto, se usa `127.0.0.1` en la configuración.

Configurar Zabbix agent 2

1. Abra el archivo de configuración del agent de Zabbix:

```
sudo vi /etc/zabbix/zabbix_agent2.conf
```

2. Establezca el parámetro Server en 127.0.0.1, ya que el agent y el server se ejecutan en la misma máquina:

```
Server=127.0.0.1
```

3. Guarde el archivo y reinicie el servicio de Zabbix agent 2:

```
sudo systemctl restart zabbix-agent2
```

4. Después de instalar y configurar Zabbix agent 2, compruebe su disponibilidad con:

```
zabbix_get -s 127.0.0.1 -k web.certificate.get[<website_DNS_name>]
```

Zabbix agent 2 incluye el plugin WebCertificate de forma predeterminada, por lo que no es necesario realizar ninguna instalación ni configuración adicional.

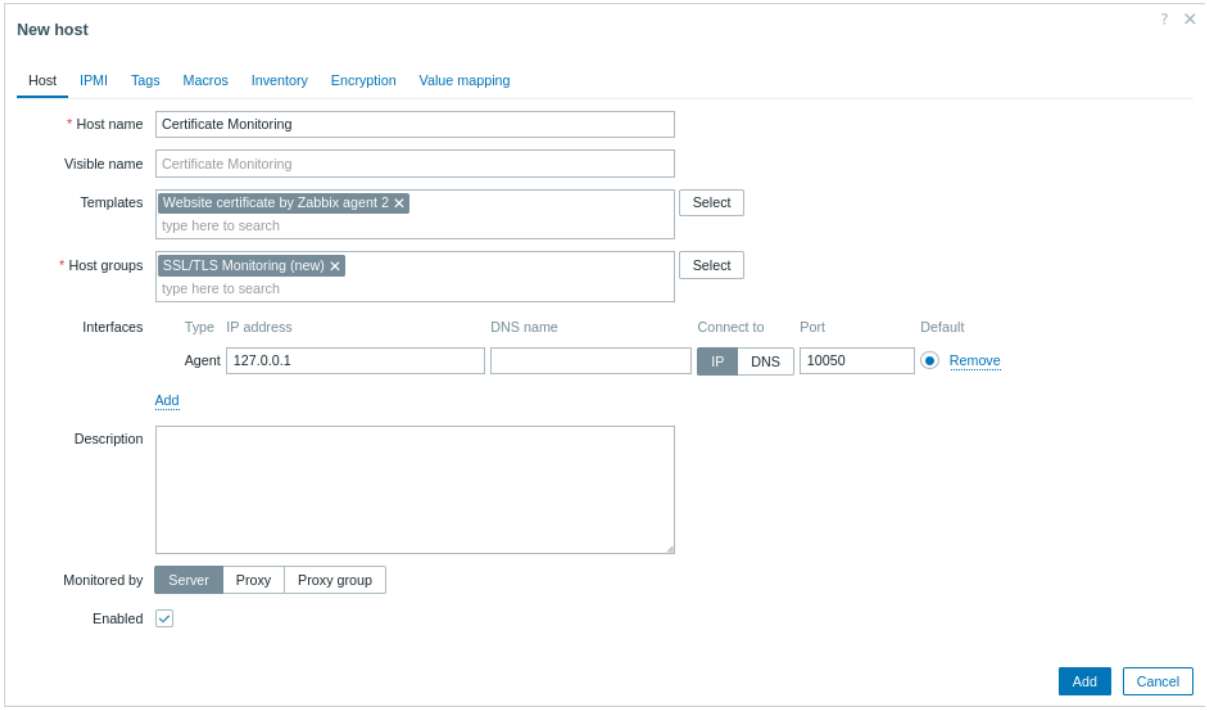
Configurar el frontend de Zabbix

1. Inicie sesión en el frontend de Zabbix.

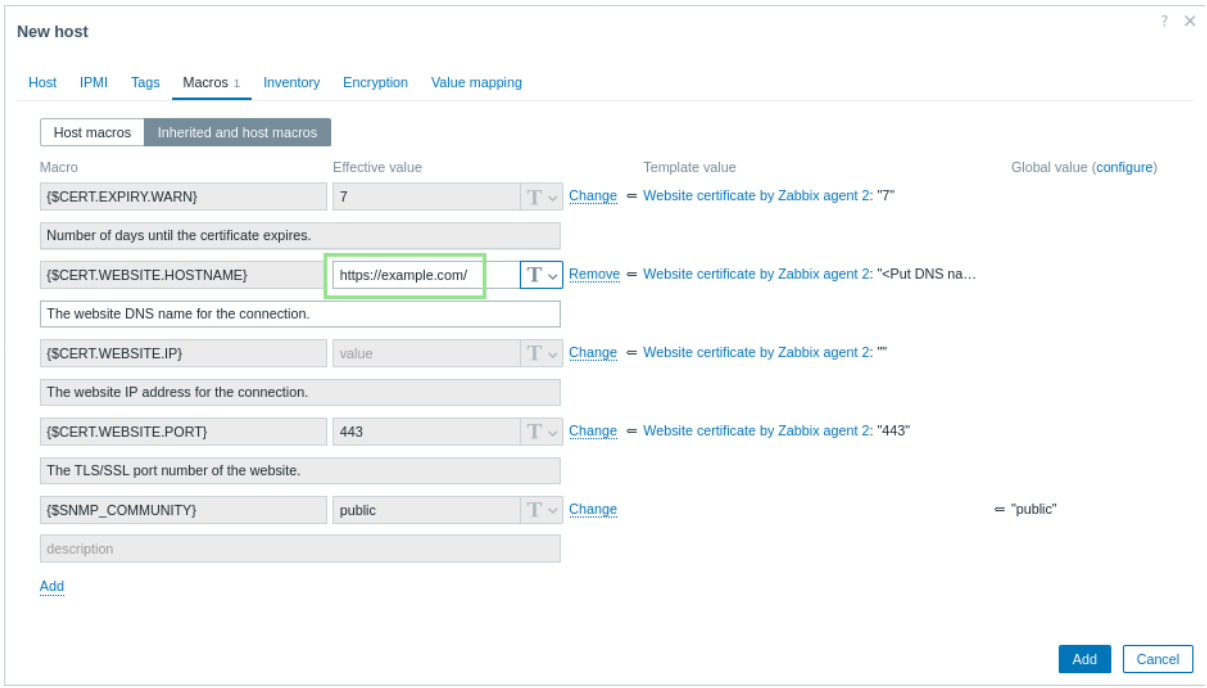
2. Vaya a *Monitoring* > *Hosts*.

3. Haga clic en un host existente donde desee supervisar los certificados del sitio web, o cree un host si es necesario:

- En el campo *Host name*, introduzca un nombre de host (por ejemplo, ”Certificate Monitoring”).
- En el campo *Templates*, escriba o seleccione el template ”Website certificate by Zabbix agent 2” que se vinculará al host.
- En el campo *Host groups*, escriba o seleccione un grupo de hosts (por ejemplo, ”SSL/TLS Monitoring”).
- En el campo *Interfaces*, agregue una interfaz de tipo ”Agent” y especifique una dirección IP. Para este ejemplo, se usa ”127.0.0.1”.



- En la pestaña *Macros*, cambie a *Inherited and host macros*, busque las siguientes macros y haga clic en *Change* junto al valor de la macro para actualizarlo:
  - `{$CERT.WEBSITE.HOSTNAME}` - introduzca como valor el nombre DNS del sitio web deseado.



4. Haga clic en *Add* para agregar el host.

Para supervisar varios sitios web, puede especificar una lista separada por comas de sus nombres de host en la macro `{$CERT.WEBSITE.HOSTNAME}`. Opcionalmente, también puede proporcionar una lista separada por comas de puertos en la macro `{$CERT.WEBSITE.PORT}`, donde cada puerto corresponde al nombre de host respectivo en orden. Por ejemplo, para supervisar example.com y example.org en el puerto 8443:

- `{$CERT.WEBSITE.HOSTNAME}`: example.com,example.org
- `{$CERT.WEBSITE.PORT}`: 443,8443

Para cada sitio web especificado en la macro `{$CERT.WEBSITE.HOSTNAME}`, Zabbix creará un conjunto correspondiente de items y triggers. Esto permite la supervisión y las alertas individuales para el certificado SSL de cada sitio web.

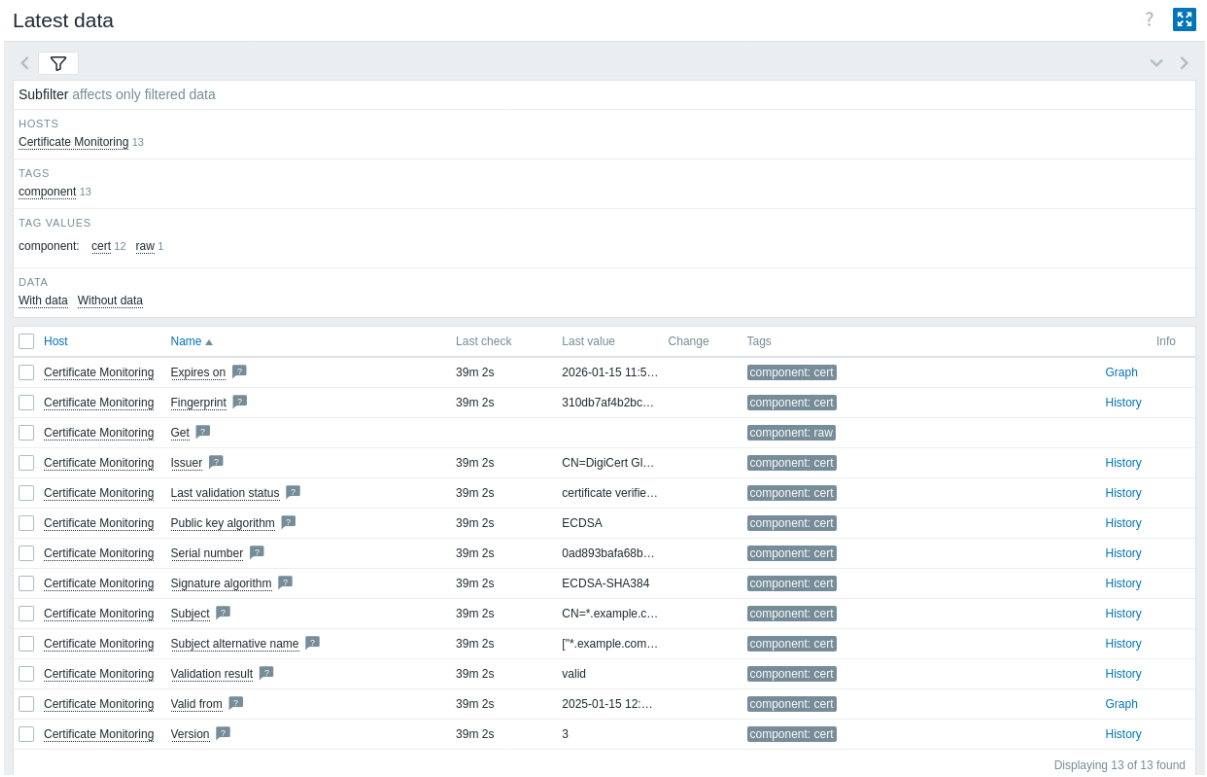


Host	Name ▲	Last check	Last value
Certificate Monitoring	Get data	11m 16s	{"error":{"message…
Certificate Monitoring	[example.com]: Expires on	11m 6s	2026-01-15 11:59:…
Certificate Monitoring	[example.com]: Fingerprint	11m 6s	310db7af4b2bc90…
Certificate Monitoring	[example.com]: Get		
Certificate Monitoring	[example.com]: Issuer	11m 6s	CN=DigiCert Glob…
Certificate Monitoring	[example.com]: Last validation status	11m 6s	certificate verified …
Certificate Monitoring	[example.com]: Public key algorithm	11m 6s	ECDSA
Certificate Monitoring	[example.com]: Serial number	11m 6s	0ad893bafa68b0b…
Certificate Monitoring	[example.com]: Signature algorithm	11m 6s	ECDSA-SHA384
Certificate Monitoring	[example.com]: Subject	11m 6s	CN=*.example.co…
Certificate Monitoring	[example.com]: Subject alternative name	11m 6s	["*.example.com","…
Certificate Monitoring	[example.com]: Validation result	11m 6s	valid
Certificate Monitoring	[example.com]: Valid from	11m 6s	2025-01-15 12:00:…
Certificate Monitoring	[example.com]: Version	11m 6s	3
Certificate Monitoring	[example.org]: Expires on	10m 52s	2026-01-15 11:59:…
Certificate Monitoring	[example.org]: Fingerprint	10m 52s	3b451cfce915637…
Certificate Monitoring	[example.org]: Get		
Certificate Monitoring	[example.org]: Issuer	10m 52s	CN=DigiCert Glob…
Certificate Monitoring	[example.org]: Last validation status	10m 52s	certificate verified …
Certificate Monitoring	[example.org]: Public key algorithm	10m 52s	ECDSA
Certificate Monitoring	[example.org]: Serial number	10m 52s	0722a749b558476…
Certificate Monitoring	[example.org]: Signature algorithm	10m 52s	ECDSA-SHA384
Certificate Monitoring	[example.org]: Subject	10m 52s	CN=*.example.org…
Certificate Monitoring	[example.org]: Subject alternative name	10m 52s	["*.example.org","e…
Certificate Monitoring	[example.org]: Validation result	10m 52s	valid
Certificate Monitoring	[example.org]: Valid from	10m 52s	2025-01-15 12:00:…

Ver métricas recopiladas

¡Enhorabuena! En este punto, Zabbix ya está supervisando el certificado web deseado.

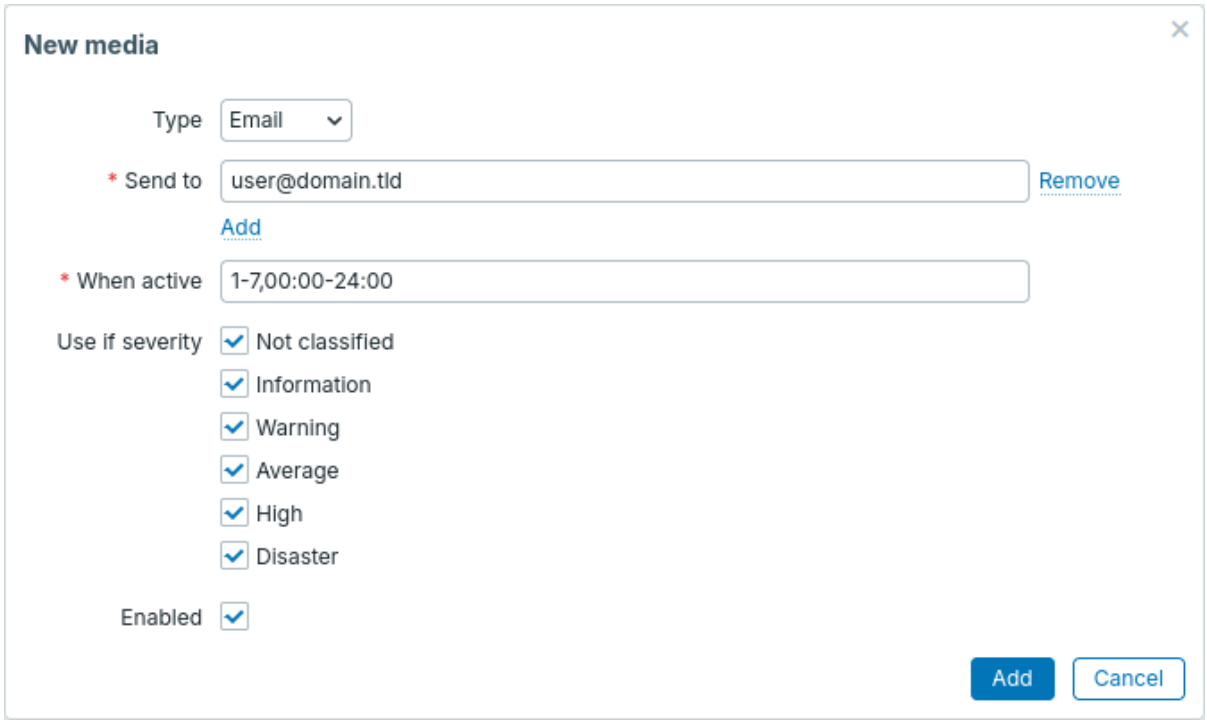
Para ver las métricas recopiladas, vaya a la sección de menú *Monitoring > Hosts* y haga clic en *Latest data* junto al host para ver todas las métricas recopiladas más recientes en una lista, como la fecha de expiración, el emisor y el sujeto.



Configurar alertas de problemas

Zabbix puede notificarle sobre problemas de infraestructura. Esta guía proporciona pasos básicos de configuración para enviar alertas por correo electrónico.

1. Vaya a *User settings > Profile*, cambie a la pestaña *Media* y agregue su correo electrónico.



2. Siga la guía para Receiving a problem notification.

La próxima vez que Zabbix detecte un problema, debería recibir una alerta por correo electrónico.

Pruebe su configuración

Para probar su configuración, podemos simular un problema real actualizando la configuración del host en el frontend de Zabbix.

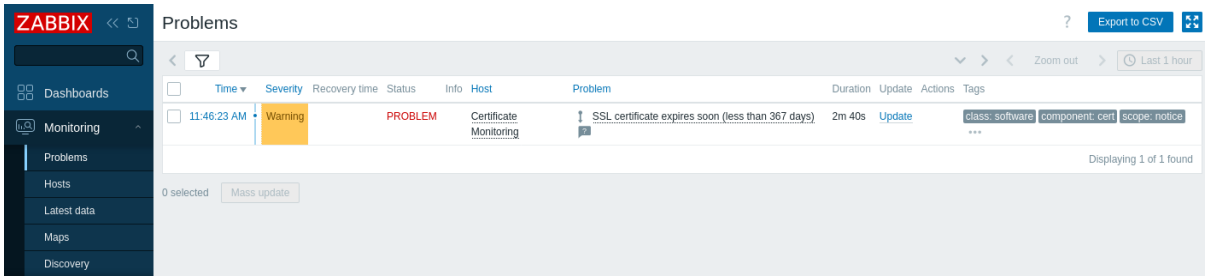
1. Abra la configuración del host ”Certificate Monitoring” en Zabbix.

2. Cambie a la pestaña *Macros* y seleccione *Inherited and host macros*.

3. Haga clic en *Change* junto al valor de macro `{$CERT.EXPIRY.WARN}` configurado anteriormente y establezca un número de días muy alto (más de 365 días debería ser suficiente) para recibir una advertencia antes de que caduque el certificado.

4. Haga clic en *Update* para actualizar la configuración del host.

5. En unos momentos, Zabbix detectará el problema "SSL certificate expires soon", con el número de días hasta la expiración. El problema aparecerá en *Monitoring > Problems*.



Si las alertas están configuradas, también recibirá la notificación del problema.

6. Vuelva a cambiar el valor de la macro a su valor anterior para resolver el problema y continuar supervisando los valores del certificado.

Ver también

- Zabbix agent 2 - enumera las claves de item.
- Template *Website certificate by Zabbix agent 2* - información adicional sobre el template *Website certificate by Zabbix agent 2*.
- Template *Website certificate by Zabbix agent 2 active* - información adicional sobre el template *Website certificate by Zabbix agent 2 active*.
- Monitor websites with Browser items - cómo iniciar la monitorización básica de sitios web con Browser items.

Monitorizar un switch o router de red con Zabbix

Introducción

Esta guía le muestra los pasos necesarios para comenzar con la monitorización básica de su switch o router de red usando Zabbix. Se utiliza un router Cisco como ejemplo, pero el procedimiento se aplica a cualquier dispositivo de red habilitado para SNMP.

Para quién está dirigida esta guía

Esta guía está diseñada para nuevos usuarios de Zabbix y administradores de red que desean habilitar rápidamente la monitorización básica de dispositivos de red. Si necesita una personalización profunda o opciones avanzadas de configuración, consulte la página SNMP agent o la sección Configuration del manual de Zabbix.

Requisitos previos

Antes de continuar con esta guía, asegúrese de tener:

- Zabbix server y Zabbix frontend instalados: instálelos según las instrucciones para su sistema operativo (consulte Installation from packages y Web interface installation).
- Zabbix agent instalado si va a monitorizar métricas de la red local.
- Dispositivo habilitado para SNMP: un switch o router de red (por ejemplo, un router Cisco) con SNMP habilitado.
- Archivos MIB instalados: instalar archivos MIB permite a Zabbix traducir OID numéricos en nombres y descripciones legibles por humanos. Sin un soporte adecuado de MIB, es posible que solo vea valores numéricos, lo que dificulta la configuración de items y la resolución de problemas.

Para instalar archivos MIB en Ubuntu:

1. Instale el paquete descargador de MIB:

```
sudo apt-get update
sudo apt-get install snmp-mibs-downloader
```

Si necesita añadir MIB específicos del fabricante (por ejemplo, de Cisco o Juniper), colóquelos en el directorio MIB correspondiente:

- En sistemas basados en Linux, las ubicaciones comunes incluyen `/usr/share/snmp/mibs/` o `/usr/local/share/snmp/mibs/`.
- En instalaciones de Zabbix, los archivos MIB pueden almacenarse en `/var/lib/zabbix/mibs/`.

Asegúrese de que la variable de entorno `MIBDIRS` o el archivo `snmp.conf` incluya la ruta correcta.

Para verificar que su sistema reconoce los nuevos MIB, use:

```
snmptranslate -IR -On <MIB-NAME>::<object>
```

Para obtener instrucciones detalladas, consulte la documentación de su biblioteca SNMP:

- Cisco MIBs
- Juniper MIBs

2. Edite `/etc/snmp/snmp.conf` y comente la línea que comienza con `mibs :` para permitir que el sistema cargue todos los MIB disponibles.

3. Verifique ejecutando un `snmpwalk` (por ejemplo, `snmpwalk -v 2c -c <your_community_string> <device_IP>`) y compruebe que los OID se muestran con nombres descriptivos.

Dependiendo de su entorno, algunos pasos de esta guía pueden diferir ligeramente. Esta guía se basa en un entorno que ejecuta Ubuntu y un dispositivo de red Cisco Catalyst 3750V2-24FS que está siendo monitorizado.

Se asume que su dispositivo de red ya está instalado físicamente y conectado.

Configure el dispositivo de red (ejemplo de router Cisco)

Para la supervisión mediante SNMP, debe configurar su dispositivo de red para permitir consultas SNMP. El ejemplo siguiente corresponde a SNMPv2 y no tiene en cuenta la configuración existente. Precaución: aplicar estos comandos puede sobrescribir las configuraciones SNMP actuales.

Para un router Cisco, la configuración normalmente implica los pasos que se indican a continuación.

Ejemplo de SNMPv2

1. Habilite SNMP y establezca la cadena de comunidad.

Inicie sesión en la consola de su router Cisco e ingrese al modo de configuración:

```
configure terminal
```

Luego, habilite SNMP especificando una cadena de comunidad de solo lectura. Por ejemplo:

```
snmp-server community <your_community_string> RO
```

Reemplace `<your_community_string>` con su cadena de comunidad segura. Nota: la opción RO (solo lectura) permite que SNMP recupere datos del dispositivo, pero impide cualquier cambio de configuración.

Se recomienda restringir el acceso SNMP solo a los dispositivos necesarios por motivos de seguridad. Para obtener más orientación sobre la configuración de listas de control de acceso (ACL), consulte la documentación oficial de Cisco.

2. Guarde la configuración.

Guarde los cambios para asegurarse de que la configuración de SNMP persista después de un reinicio:

```
write memory
```

Ejemplo de SNMPv3

SNMPv3 proporciona seguridad mejorada con autenticación y cifrado. Su configuración es más segura que la de SNMPv2 y debe verificarse con la documentación específica de su dispositivo.

1. Cree un grupo SNMP.

Configure un grupo SNMPv3 con privacidad (cifrado) habilitada:

```
configure terminal
snmp-server group <your_group> v3 priv
```

2. Cree un usuario SNMP.

Agregue un usuario SNMPv3 con autenticación y privacidad. Reemplace los marcadores de posición con los valores que desee:

```
snmp-server user <your_user> <your_group> v3 auth md5 <auth_password> priv aes 128 <priv_password>
```

3. Guarde la configuración:

```
write memory
```

Para obtener más detalles o instrucciones específicas del modelo, puede consultar tutoriales externos de configuración de SNMP de Cisco. Sin embargo, esta guía proporciona los pasos básicos para habilitar la supervisión de SNMP.

Configurar el frontend de Zabbix

Crear un host en Zabbix frontend

1. Inicie sesión en Zabbix frontend.

2. Añada un nuevo host.

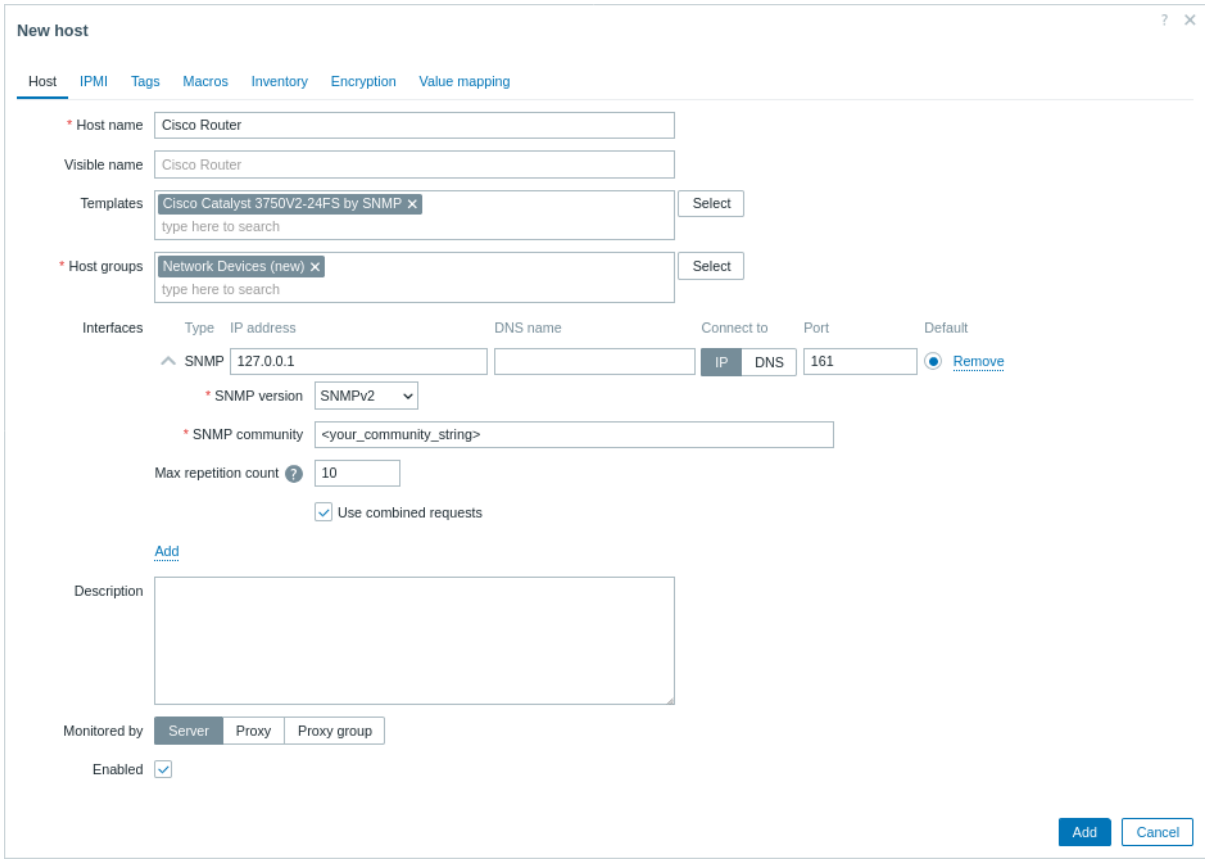
Vaya a *Data collection* > *Hosts* y haga clic en *Create host*.

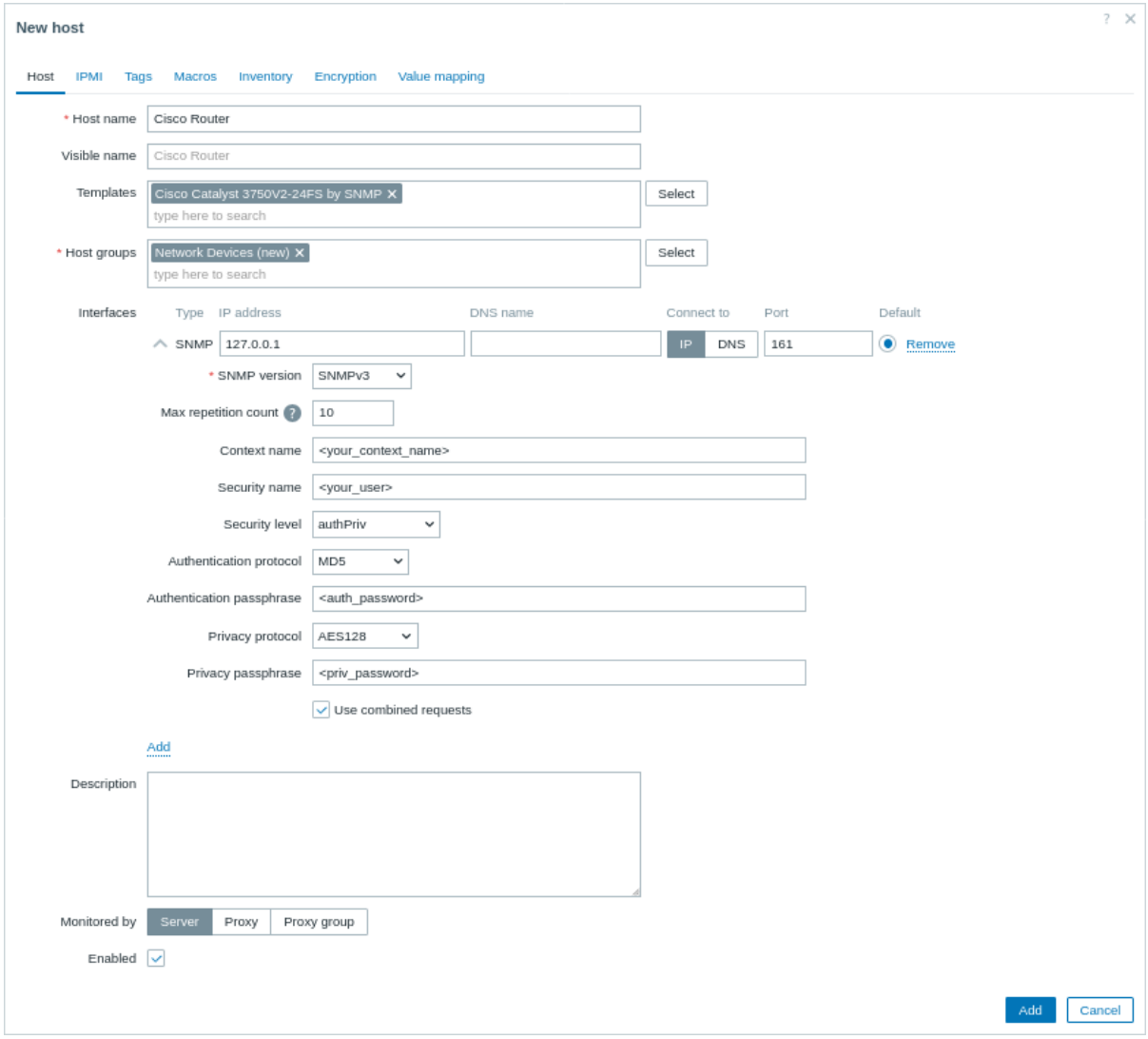
- *Host name*: introduzca un nombre para su dispositivo (por ejemplo, ”Cisco Router”).
- Host groups: seleccione un grupo existente o cree un grupo nuevo, como ”Network Devices”.
- Interfaces:
  - Haga clic en *Add* en Interfaces.
  - Elija *SNMP* como tipo de interfaz.
  - Introduzca la dirección IP o el nombre DNS de su router Cisco.
  - Establezca el puerto SNMP predeterminado (normalmente 161).
  - Use el menú desplegable para seleccionar la versión SNMP adecuada (por ejemplo, SNMPv2).
  - Para SNMPv1/v2, introduzca la cadena de comunidad en el campo *SNMP community*. Para SNMPv3, se solicitarán credenciales adicionales (*Context name*, *Security name* y *Security level*, etc.).

3. Vincular templates

En el campo *Templates*, seleccione el template SNMP que mejor se adapte a su dispositivo. Zabbix proporciona una amplia gama de SNMP templates predefinidos para muchas familias de dispositivos. Por ejemplo, si está supervisando un dispositivo Cisco, elija el template que corresponda al sistema operativo o modelo de su dispositivo (como Cisco IOS SNMP o Cisco Catalyst 3750<device model> SNMP).

4. Haga clic en *Add* para guardar el host.



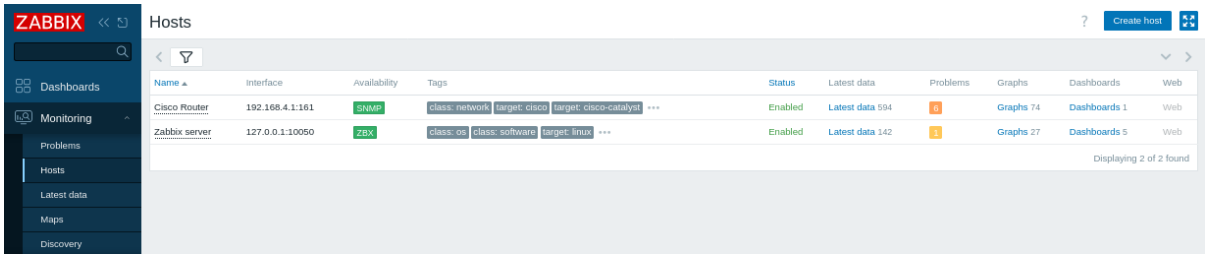


Ver métricas recopiladas

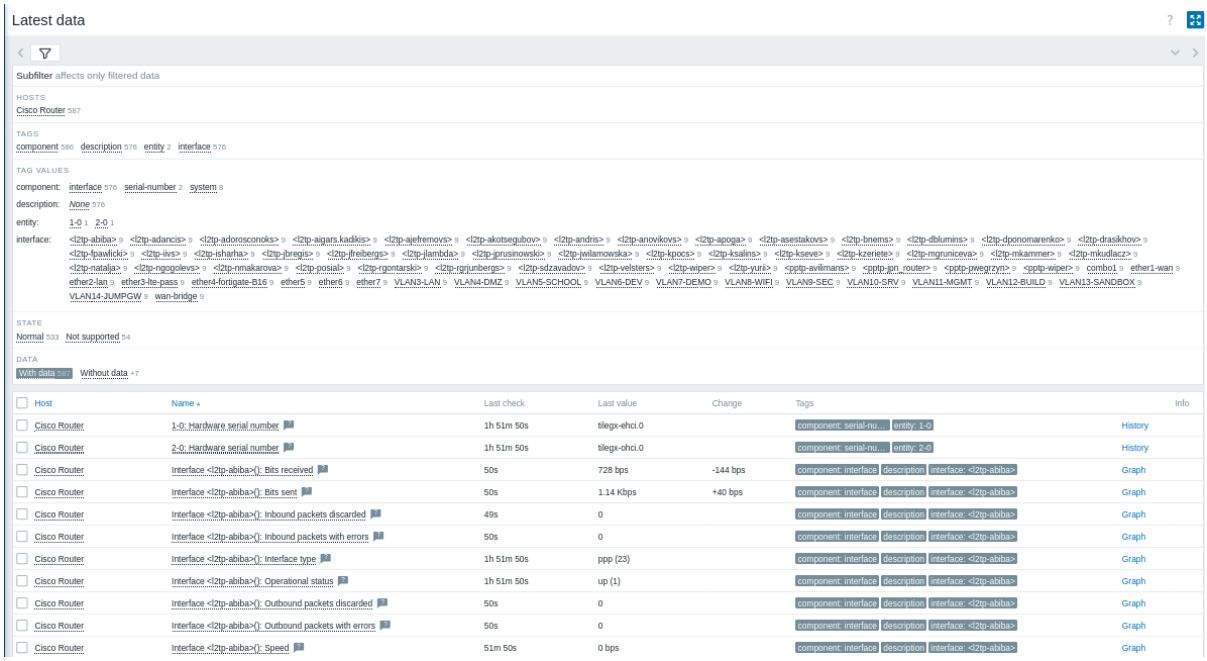
¡Enhorabuena! Zabbix ya está configurado para supervisar su dispositivo de red.

Datos más recientes:

- Vaya a Monitoring > Latest data en el frontend de Zabbix.



- Seleccione su host ”Cisco Router” (o los hosts descubiertos) para ver métricas como el tiempo de actividad del hardware y de la red, la pérdida ICMP, el ping y el tiempo de respuesta, etc.



- Gráficas y pantallas:

Para visualizar los datos de rendimiento, haga clic en *Graphs* junto a los items SNMP para ver métricas detalladas.

Como siguiente paso, puede:

- Añadir items SNMP personalizados para supervisar métricas adicionales.
- Configurar alertas de problemas para recibir notificaciones sobre posibles incidencias.

Crear items SNMP

Una vez que el host esté configurado, puede crear items para supervisar métricas específicas. Nota: este paso es opcional si está usando un template, ya que los templates ya contienen conjuntos predeterminados de items.

1. Identifique el OID SNMP:

Use el comando `snmpwalk` para listar los OID disponibles en su dispositivo. Por ejemplo:

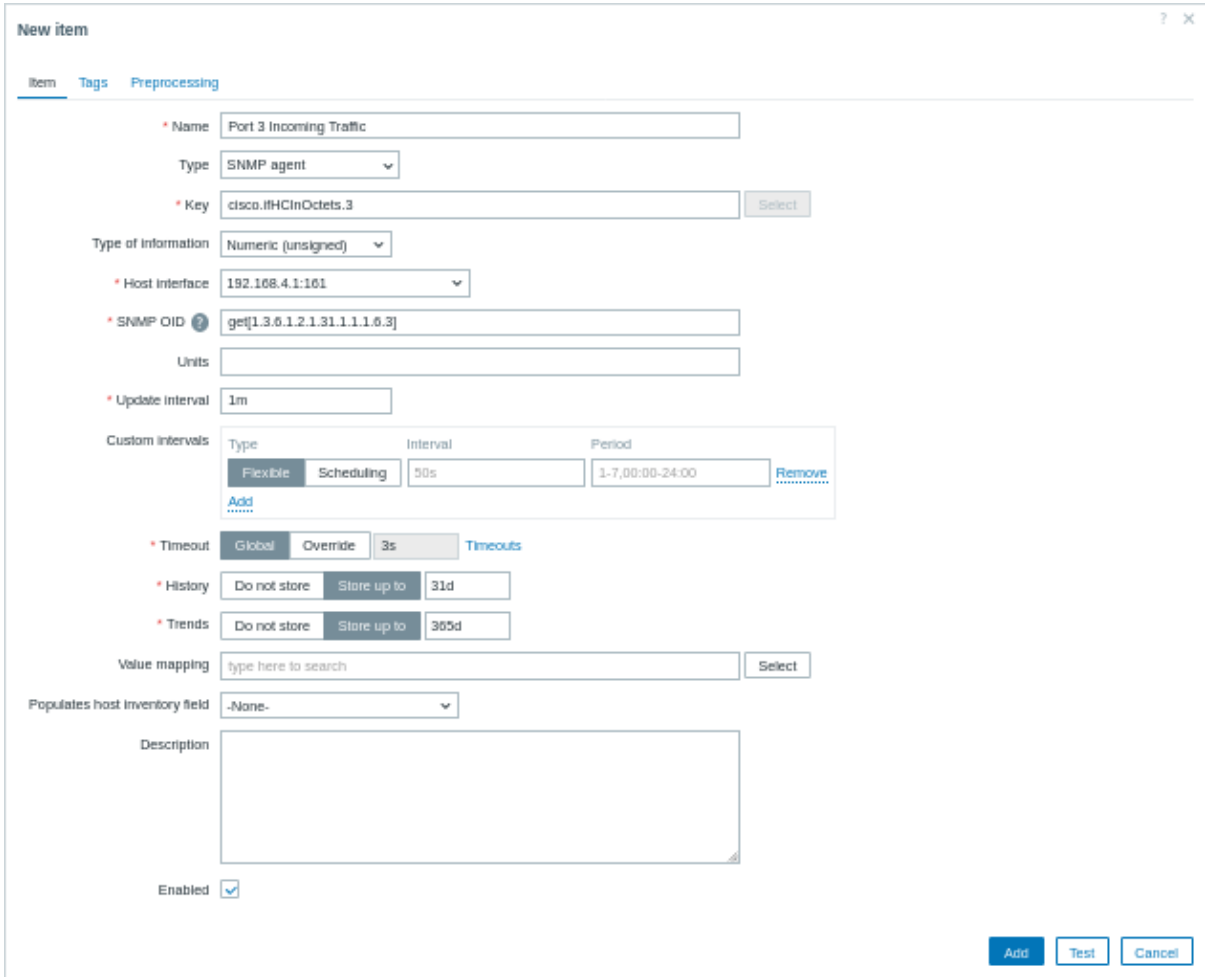
```
snmpwalk -v 2c -c <your_community_string> <device_IP> .
```

Encuentre el OID de la métrica que desea supervisar (por ejemplo, IF-MIB::ifHCInOctets.3 para el tráfico entrante en el puerto 3). Para obtener el OID numérico, puede usar:

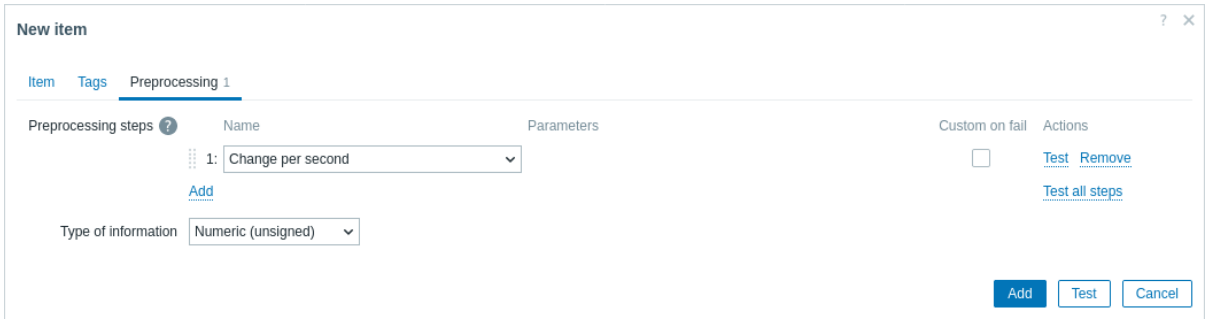
```
snmpget -v 2c -c <your_community_string> -On <device_IP> IF-MIB::ifHCInOctets.3
```

2. Cree un item SNMP:

- Vaya a *Data collection* > *Hosts* y haga clic en la pestaña *Items* de su host SNMP, luego haga clic en *Create item*.
- *Name*: introduzca un nombre descriptivo (por ejemplo, "Tráfico entrante del puerto 3").
- *Type*: seleccione *SNMP agent*.
- *Key*: proporcione una clave significativa (por ejemplo, `cisco.ifHCInOctets.3`).
- *Host interface*: asegúrese de que esté seleccionada la interfaz SNMP.
- *SNMP OID*: introduzca el OID usando uno de los formatos admitidos, por ejemplo:
  - `get[1.3.6.1.2.1.31.1.1.1.6.3]` para un valor único;
  - `walk[1.3.6.1.2.1.31.1.1.1.6.3]` para recuperar de forma asíncrona un subárbol de valores.



- *Preprocessing* (si es necesario): si el item devuelve un contador acumulativo (como el tráfico de una interfaz), vaya a la pestaña *Preprocessing* y añada un paso de preprocesamiento como "Change per second" para calcular la tasa.



Para recuperar varios valores en una sola transacción SNMP, puede especificar varios OID usando la sintaxis `walk[OID1,OID2,...]`.

Traducción de OIDs entre nombres numéricos y nombres MIB

Al trabajar con SNMP, es posible que necesite convertir entre OIDs numéricos y sus nombres MIB correspondientes. Esta traducción ayuda a identificar y solucionar problemas de métricas con mayor facilidad.

- Para traducir un nombre MIB a un OID numérico: use el comando `snmptranslate` con la opción `-On`. Por ejemplo, para traducir el nombre MIB `IF-MIB::ifHCInOctets.3` a su OID numérico, ejecute:

```
snmptranslate -On IF-MIB::ifHCInOctets.3
```

Este comando podría devolver:

```
.1.3.6.1.2.1.31.1.1.1.6.3
```

- Para traducir un OID numérico a su nombre MIB: use el comando `snmptranslate` con la opción `-IR` (o `-m ALL`) para invertir la traducción. Por ejemplo, para traducir el OID numérico `.1.3.6.1.2.1.31.1.1.1.6.3` de vuelta a su nombre MIB, ejecute:

```
snmptranslate -IR -On .1.3.6.1.2.1.31.1.1.1.6.3
```

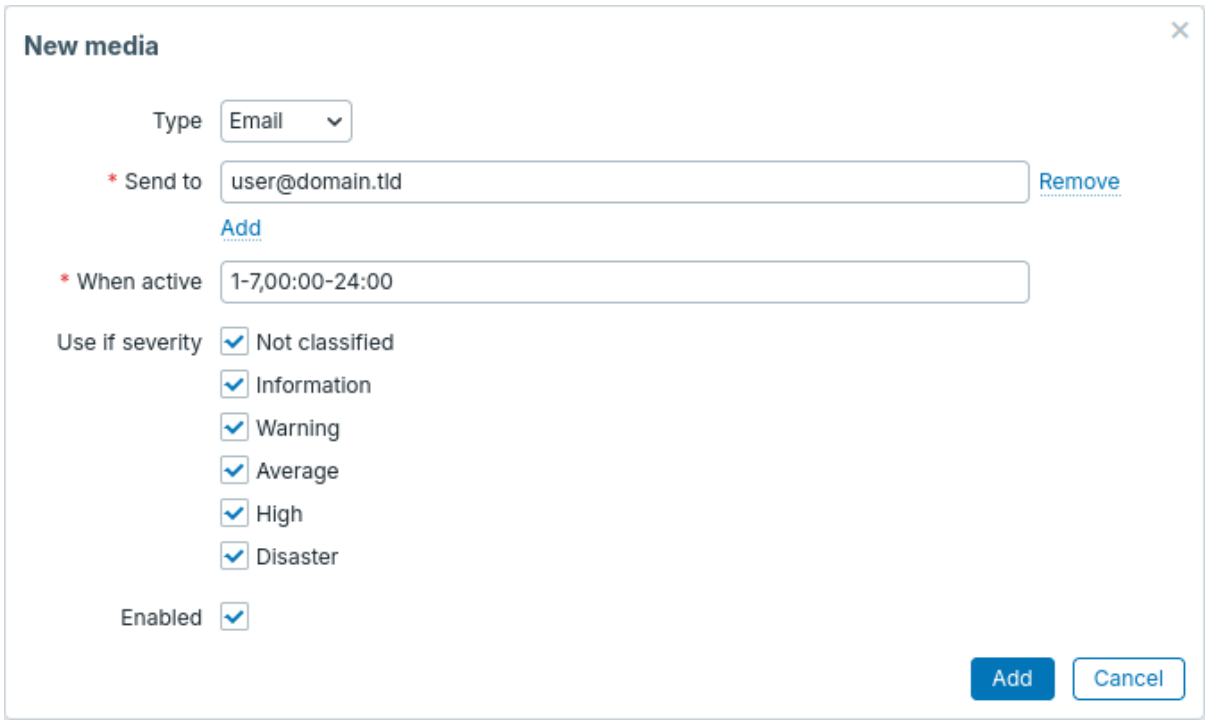
Este comando podría devolver:

```
IF-MIB::ifHCInOctets.3
```

Configurar alertas de problemas

Esta guía proporciona pasos básicos de configuración para enviar alertas por correo electrónico.

1. Vaya a *User settings > Profile*, cambie a la pestaña *Media* y añada su correo electrónico.



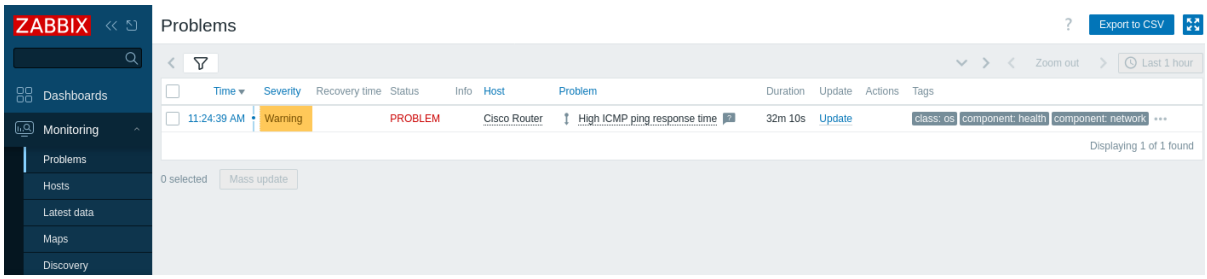
2. Siga la guía para Recibir una notificación de problema.

La próxima vez que Zabbix detecte un problema, debería recibir una alerta por correo electrónico.

Pruebe su configuración

Para asegurarse de que Zabbix detecta correctamente los problemas de rendimiento de red, simule un problema real aumentando el umbral del tiempo de respuesta del ping ICMP.

1. Abra la configuración del host "Cisco Router" en Zabbix.
2. Vaya a la pestaña *Macros* y seleccione *Inherited and host macros*.
3. Localice la macro `{$ICMP_RESPONSE_TIME_WARN}` (o una macro similar de umbral de tiempo de respuesta).
4. Establezca un valor muy bajo (por ejemplo, 0.001) para activar una alerta cuando la respuesta del ping supere este valor.
5. Haga clic en *Update* para aplicar los cambios.
6. Espere unos momentos a que Zabbix detecte el problema simulado.
7. Vaya a *Monitoring > Problems* para verificar que aparezca una alerta (por ejemplo, "High ICMP ping response time").



Si las alertas están configuradas, también debería recibir una alerta de problema.

8. Revierta el valor de la macro a su configuración original y haga clic en *Update* para guardar los cambios.
9. Confirme que el problema se ha resuelto y desaparece de la sección *Problems*.

Solución de problemas de monitoreo SNMP

Si observa que el icono de SNMP en el frontend de Zabbix aparece en ROJO o no se recopilan datos, pruebe los siguientes pasos:

1. Compruebe la conectividad SNMP.

Para SNMPv2, ejecute el siguiente comando desde su server de Zabbix:

```
snmpwalk -v 2c -c <community_string> <device_IP> .
```

Este comando verifica que el dispositivo responde a consultas SNMP.

Para SNMPv3, incluya las credenciales SNMPv3 correspondientes:

```
snmpwalk -v3 -u <your_user> -l authPriv -a MD5 -A <auth_password> -x AES -X <priv_password> <device_IP> .
```

Esto verifica que las credenciales SNMPv3 sean correctas y que el dispositivo responda de forma segura.

2. Asegúrese de que los archivos MIB estén instalados y habilitados como se describe en los prerrequisitos. Para asegurarlo, el siguiente comando no debe devolver errores cuando consulte un dispositivo de red:

```
snmpwalk -v 2c -c <your_community_string> <device_IP> ifInOctets
```

Esto debería devolver OID traducidos sin errores.

3. Confirme que la versión SNMP y las credenciales configuradas en Zabbix coincidan con las configuradas en su dispositivo. Por ejemplo, revise la configuración SNMP en la configuración del host de Zabbix y compárela con la configuración de su dispositivo. En un dispositivo Cisco, puede comprobar la configuración SNMP ejecutando:

```
show running-config | include snmp
```

Esto garantiza que la cadena de comunidad (para SNMPv2) o los detalles del usuario SNMPv3 sean correctos.

4. Verifique que SNMP esté habilitado correctamente en su dispositivo de red. En un router Cisco, inicie sesión en la consola y ejecute:

```
show running-config | include snmp
```

Este comando muestra la configuración SNMP activa y ayuda a confirmar que SNMP está configurado correctamente.

5. Asegúrese de que ningún firewall o problema de red esté bloqueando el tráfico SNMP (normalmente en el puerto 161) entre el server de Zabbix y el dispositivo. Puede probar la conectividad usando:

nc -zv <device_IP> 161

`nc -zv` comprueba si el puerto 161 está abierto y escuchando en el dispositivo.

Además, si está usando UFW en Ubuntu, compruebe el estado del firewall:

```
sudo ufw status
```

O, para iptables:

```
sudo iptables -L -n
```

6. Revise los archivos de registro del server de Zabbix para detectar cualquier error relacionado con SNMP que ayude a identificar el problema:

```
tail -f /tmp/zabbix_server.log
```

`tail -f` le permite supervisar las actualizaciones del registro en tiempo real.

Ver también:

- Creación de un item - aprenda cómo añadir métricas adicionales.
- agent SNMP - información adicional sobre la monitorización SNMP con Zabbix.
- templates estandarizadas para dispositivos de red - información sobre las templates SNMP disponibles.
- Descubrimiento de OID SNMP - información adicional sobre el descubrimiento SNMP en un switch.
- Configuración de una regla de descubrimiento de red - información adicional sobre cómo configurar una regla de descubrimiento de red utilizada por Zabbix para descubrir hosts y servicios.

Supervisar el registro de eventos de Windows mediante comprobaciones activas

Introducción

Esta guía explica cómo supervisar los registros de eventos de Windows con Zabbix mediante comprobaciones activas. Con las claves de item específicas de Windows de Zabbix, puede recopilar y analizar eventos críticos (como intentos de inicio de sesión fallidos, errores del sistema, etc.) en tiempo real.

A quién va dirigida esta guía

Esta guía está diseñada para nuevos usuarios de Zabbix y administradores de red que desean supervisar los registros de eventos de Windows. Para opciones de configuración avanzadas, consulte la documentación de claves de item específicas de Windows.

Requisitos previos

Antes de continuar con esta guía, debe descargar e instalar Zabbix server y Zabbix frontend según las instrucciones para su sistema operativo. También necesita tener Zabbix agent descargado e instalado en el equipo Windows que desea supervisar.

Configurar el agent de Zabbix para la supervisión del registro de eventos de Windows

1. Abra el archivo de configuración del agent de Zabbix en su host de Windows y asegúrese de que el parámetro `ServerActive` esté establecido en la dirección IP de su server de Zabbix y que el parámetro `Hostname` coincida con el nombre del host que se definirá en el frontend de Zabbix. Esto permite que el agent solicite comprobaciones activas para su host desde el server de Zabbix especificado. Por ejemplo:

```
ServerActive=192.0.2.1
Hostname=MyWindowsHost
```

2. Reinicie el servicio del agent de Zabbix para aplicar los cambios:

```
net stop "Zabbix Agent" && net start "Zabbix Agent"
```

3. Compruebe que el host de Windows esté en ejecución:

- Asegúrese de que el servicio del agent de Zabbix esté en ejecución en el host de Windows.
- Compruebe que el host de Windows pueda conectarse al server de Zabbix en el puerto 10051. Para probar la conectividad desde el host de Windows, abra PowerShell y ejecute el siguiente comando:

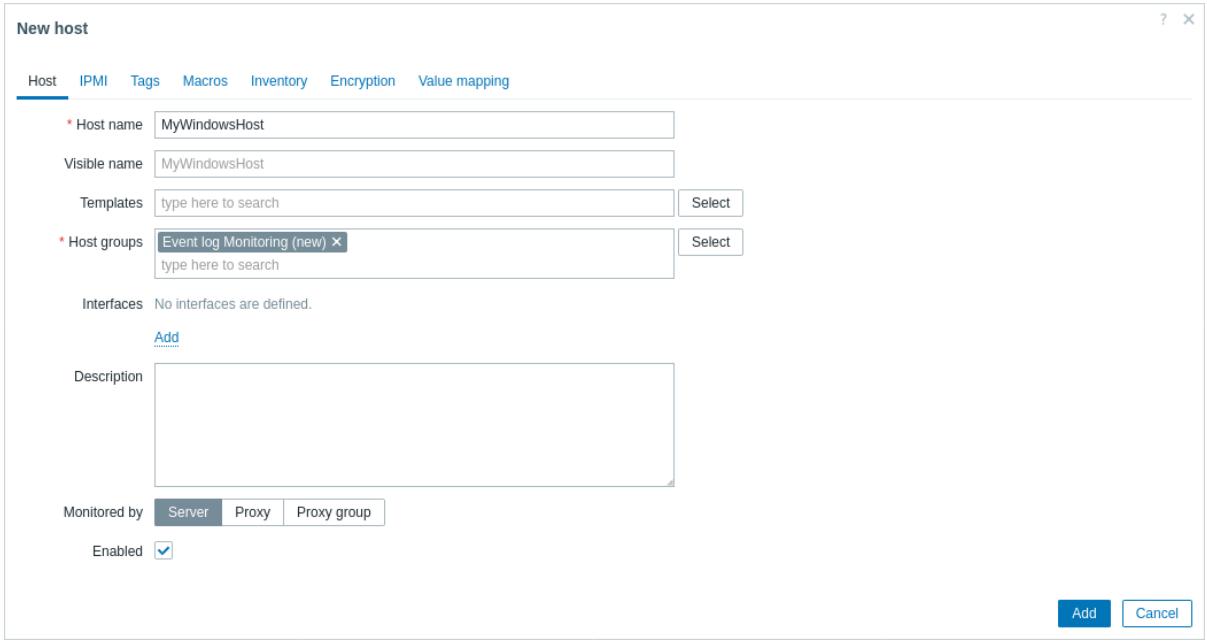
```
Test-NetConnection -ComputerName <Zabbix-server-IP> -Port 10051
```

Configurar el frontend de Zabbix

1. Vaya a *Data collection* > *Hosts* y cree un host:

- En el campo *Host name*, introduzca un nombre de host (por ejemplo, ”MyWindowsHost”).
- En el campo *Host groups*, escriba o seleccione un grupo de hosts (por ejemplo, ”Event log Monitoring”).
- Haga clic en *Add* para guardar el host configurado.

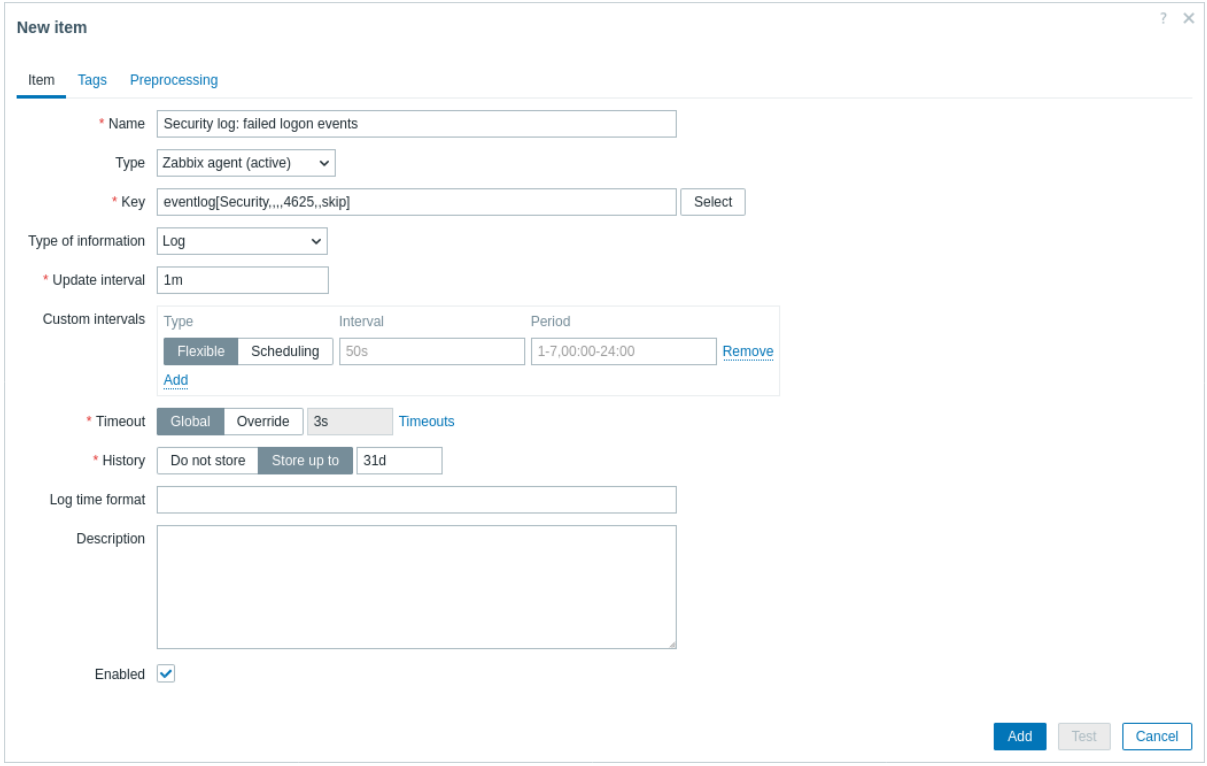
Note:
En el campo *Templates* puede añadir la template ”Windows by Zabbix agent active” para ayudarle a solucionar problemas observando si otros items activos en el mismo host se están actualizando.



2. Cree un nuevo item con los siguientes parámetros:

- En el campo *Name*, introduzca un nombre descriptivo para el item (por ejemplo, ”Security log: failed logon events”).
- En la lista desplegable *Type*, seleccione ”Zabbix agent (active)” (requerido para la supervisión de Event log).

- En el campo *Key*, use la clave de item eventlog. Por ejemplo, para supervisar intentos fallidos de inicio de sesión (Event ID: 4625) en el registro Security y omitir las entradas anteriores a la última comprobación del item (usando el parámetro `skip`), introduzca la siguiente clave de item: `eventlog[Security,,,,4625,,skip]`
- En la lista desplegable *Type of information*, seleccione "Log".



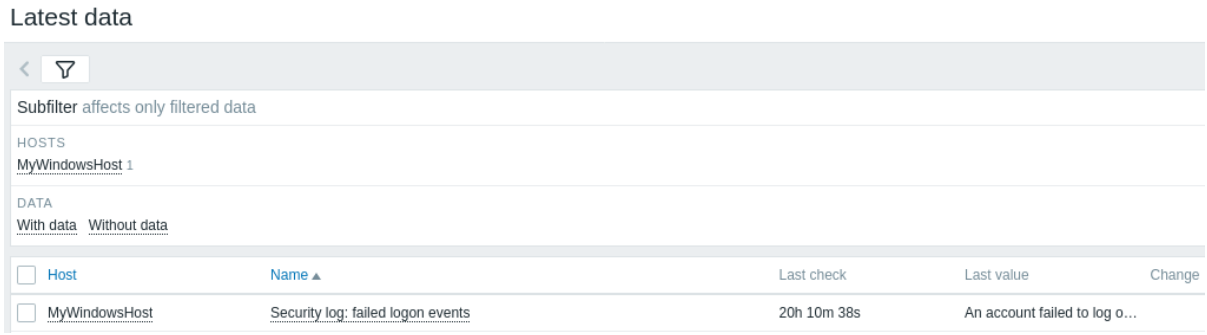
3. Haga clic en *Add* para guardar el item.

Probar y ver las métricas recopiladas

¡Enhorabuena! Zabbix ya está configurado para recopilar los registros de eventos de Windows. Para verificar que los registros de eventos se estén recopilando, puede probar el item "Security log: failed logon events" cerrando sesión en su cuenta de Windows e intentando iniciar sesión con credenciales incorrectas.

Luego, vea los registros recopilados en el frontend de Zabbix:

1. Vaya a *Monitoring* > *Latest data* en el frontend de Zabbix.



2. Filtre por su host "MyWindowsHost" en el campo *Name*.

3. Haga clic en *History* para ver los valores de registro almacenados.

MyWindowsHost: Security log: failed logon events

Timestamp	Local time	Source	Severity	Event ID	Value
2025-04-07 11:13:11 PM	2025-04-07 11:12:19 PM	Microsoft-Windows-Security-Auditing	Failure Audit	4625	An account failed to log on. Subject: Security ID: NT AUTHORITY\SYSTEM Account Name: DESKTOP-NNEM26U$ Account Domain: WORKGROUP Logon ID: 0x3E7 Logon Type: 2 Account For Which Logon Failed: Security ID: NULL SID Account Name: Admin Account Domain: DESKTOP-NNEM26U Failure Information: Failure Reason: Unknown user name or bad password. Status: 0xC000006D Sub Status: 0xC000006A

4. Si no hay valores de registro, continúe con la sección Troubleshooting de la guía.

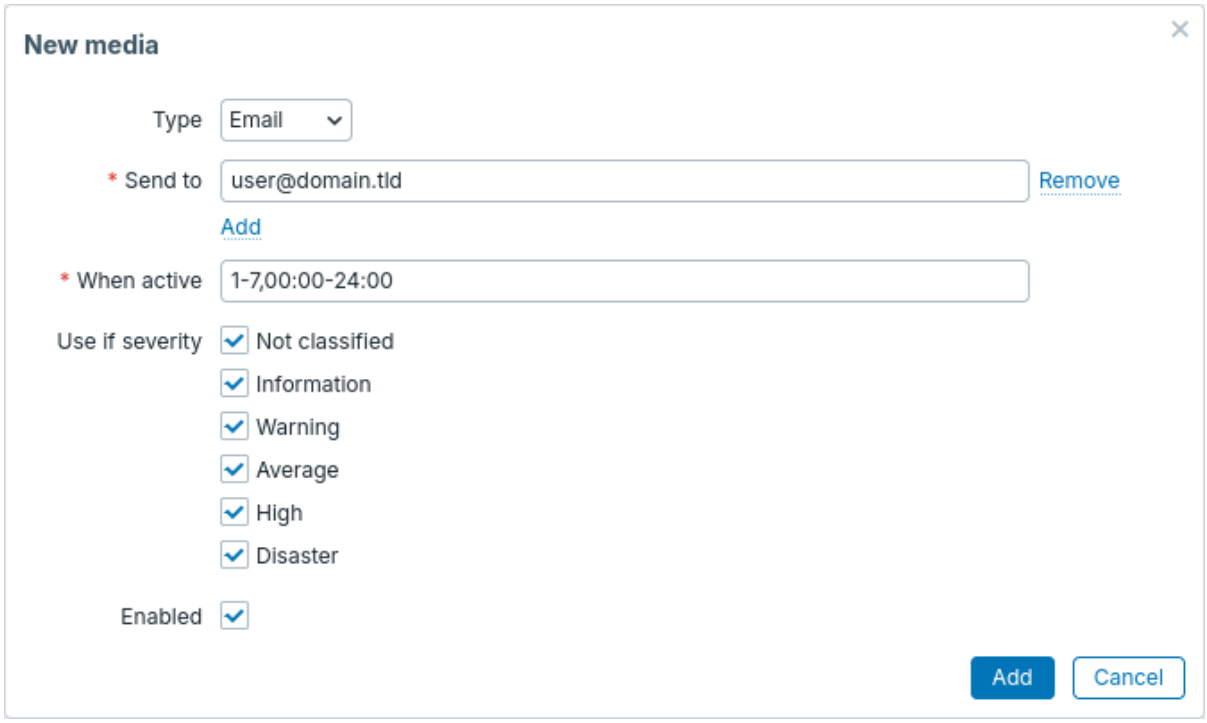
Configurar alertas de problemas

Esta guía proporciona pasos básicos de configuración para enviar alertas por correo electrónico.

1. Vaya a Recopilación de datos > Hosts para definir un trigger que se active cuando su item de registro de eventos registre el patrón que le interesa. Por ejemplo, para detectar intentos de inicio de sesión fallidos en el registro de Seguridad, use la función find():

find(/MyWindowsHost/eventlog[Security,,,4625,skip],10m,"like","Logon failed")

2. Vaya a *Ajustes de usuario > Perfil*, cambie a la pestaña *Media* y añada su correo electrónico.



3. Siga la guía para Recibir una notificación de problema.

La próxima vez que Zabbix detecte un problema, debería recibir una alerta por correo electrónico.

Solución de problemas

Si encuentra problemas al recopilar o visualizar registros de eventos de Windows, utilice los siguientes consejos para identificar y resolver problemas comunes:

1. En el server de Zabbix (Linux), enumere sus reglas de iptables con el siguiente comando:

```
sudo iptables -L -n
```

y verifique que exista una regla ACCEPT para el puerto TCP 10051.

2. Asegúrese de que su clave `eventlog[...]` use exactamente el nombre del registro (sensible a mayúsculas y minúsculas), el ID del evento, el modo (por ejemplo, skip) y otros parámetros exactamente como se muestran en las Windows-specific item keys.

Véase también:

- Crear un item - aprenda cómo añadir métricas adicionales.
- Zabbix agent en Microsoft Windows - instrucciones detalladas de instalación.
- Monitorizar Windows con Zabbix agent - una guía completa para configurar la monitorización básica de equipos Windows usando Zabbix agent.
- Claves de item específicas de Windows - información detallada sobre las claves de item específicas de Windows compatibles con Zabbix agents, incluidas las de monitorización de registros de eventos.
- Monitorización de archivos de registro - instrucciones para configurar Zabbix para la monitorización y el análisis centralizados de archivos de registro, aplicable a los registros de eventos de Windows.

Instalación: guías adicionales

Si busca la forma más sencilla de instalar Zabbix, consulte Instalación y primeros pasos.

Las guías siguientes describen formas **adicionales** o específicas del sistema para instalar Zabbix o sus componentes:

- Instalación desde contenedores
- Instalación desde las fuentes
- Notas sobre la instalación desde paquetes
- Instalación del agent de Zabbix desde Windows MSI
- Compilación de Zabbix agent 2 en Windows
- Compilación del agent de Zabbix en Windows
- Instalación del agent para macOS desde PKG
- Zabbix agent en IBM AIX
- Compilación del agent de Zabbix en macOS
- Zabbix appliance
- Guía de implementación de Zabbix server en AWS
- Guía de implementación de Zabbix proxy en AWS
- Guía de implementación de Zabbix server en Azure
- Guía de implementación de Zabbix proxy en Azure
- Guía de implementación en Google Cloud

Instalación desde contenedores

Descripción general

Esta página describe cómo desplegar Zabbix usando Docker o Docker Compose.

Docker Compose es la forma más rápida de desplegar Zabbix. Lee un archivo de configuración e inicia automáticamente todos los contenedores que componen una instalación completa de Zabbix, en el orden correcto.

Docker (despliegue manual) logra el mismo resultado desplegando cada componente paso a paso.

Requisitos previos

Antes de comenzar, asegúrese de que Docker (1.12.0 o posterior) esté instalado en su sistema. Si no lo está, siga la guía de instalación de Docker. Para la implementación con Docker Compose, también se requiere Docker Compose (2.24.0 o posterior).

Algunos componentes de Zabbix requieren que determinados puertos estén abiertos en el host donde se ejecuta Docker (por ejemplo, 10051/TCP para el server de Zabbix y 162/UDP para las trampas SNMP). Consulte Requisitos para obtener la lista completa

de puertos utilizados por los componentes de Zabbix. Para el server y el agent de Zabbix, el puerto predeterminado se puede cambiar estableciendo la variable de entorno `ZBX_LISTENPORT` en el contenedor correspondiente.

Imágenes de Docker disponibles

Zabbix proporciona una imagen de Docker para cada componente de Zabbix, todas publicadas en Docker Hub. Cada imagen se utiliza para crear un contenedor que ejecuta ese componente.

Componente de Zabbix	Imagen de Docker
Agent	zabbix/zabbix-agent
Agent 2	zabbix/zabbix-agent2
Server (MySQL)	zabbix/zabbix-server-mysql
Server (PostgreSQL)	zabbix/zabbix-server-pgsql
Interfaz web (Apache + MySQL)	zabbix/zabbix-web-apache-mysql
Interfaz web (Apache + PostgreSQL)	zabbix/zabbix-web-apache-pgsql
Interfaz web (Nginx + MySQL)	zabbix/zabbix-web-nginx-mysql
Interfaz web (Nginx + PostgreSQL)	zabbix/zabbix-web-nginx-pgsql
Proxy (SQLite3)	zabbix/zabbix-proxy-sqlite3
Proxy (MySQL)	zabbix/zabbix-proxy-mysql
Java gateway	zabbix/zabbix-java-gateway
Servicio web	zabbix/zabbix-web-service
Traps SNMP	zabbix/zabbix-snmptraps
Servidor MCP	zabbix/zabbix-mcp-server

Note:
Para utilizar traps SNMP, el contenedor de traps SNMP debe compartir un volumen con el contenedor del server o proxy de Zabbix (consulte cómo utilizar esta imagen en Docker Hub y el ejemplo que aparece a continuación).

Todas las imágenes de los componentes de Zabbix en Docker Hub se basan en las últimas versiones principales de los sistemas operativos compatibles. Estas imágenes se reconstruyen automáticamente cuando se actualizan las imágenes del sistema operativo subyacente.

Etiquetas de imagen

Cada imagen admite etiquetas para seleccionar el sistema operativo base y la versión de Zabbix:

```
zabbix/<image>:<os>-<version>
```

Valores `<os>` admitidos:

- `alpine` - Alpine Linux
- `ubuntu` - Ubuntu
- `centos` - CentOS Stream
- `ol` - Oracle Linux
- `ltsc2022` - Windows 11 LTSC 2022 (solo Zabbix agent)

Valores `<version>` admitidos:

- `latest` - La última versión estable de Zabbix en Alpine Linux
- `<os>-latest` - La última versión estable de Zabbix en el SO seleccionado
- `<os>-trunk` - La compilación de desarrollo (nightly) más reciente en el SO seleccionado
- `<os>-X.X-latest` - La última versión menor de Zabbix de una versión mayor específica de Zabbix en el SO seleccionado
- `<os>-X.X.*` - Una versión menor específica de Zabbix en el SO seleccionado

Ejemplos:

```
#### Último Zabbix proxy estable (SQLite3) en Alpine Linux:
docker pull zabbix/zabbix-proxy-sqlite3:latest

#### Último Zabbix proxy estable (SQLite3) en Ubuntu:
docker pull zabbix/zabbix-proxy-sqlite3:ubuntu-latest

#### Compilación de desarrollo (nightly) más reciente de Zabbix server (MySQL) en Ubuntu:
docker pull zabbix/zabbix-server-mysql:ubuntu-trunk

#### Última versión menor 8.0 de Zabbix server (MySQL) en Alpine Linux:
```

```
docker pull zabbix/zabbix-server-mysql:alpine-8.0-latest

#### Versión 8.0.1 de Zabbix server (MySQL) en Alpine Linux:
docker pull zabbix/zabbix-server-mysql:alpine-8.0.1
```

Docker Compose

Docker Compose es la forma más rápida de implementar Zabbix. Lee un archivo de configuración (archivo Compose) y gestiona toda la configuración: descarga las imágenes de Docker, crea una red interna entre los contenedores, configura el almacenamiento, inicializa la base de datos e inicia todo en el orden correcto.

El repositorio de Dockerfiles oficiales de Zabbix proporciona archivos Docker Compose listos para usar y un sistema de configuración basado en `.env` compatible con distintos sistemas operativos, backends de bases de datos y configuraciones de los componentes de Zabbix.

1. Clone el repositorio, acceda a él y cambie a la versión 8.0:

```
git clone https://github.com/zabbix/zabbix-docker.git
cd zabbix-docker
git checkout 8.0
```

2. (Opcional) Personalice la implementación con variables de entorno. Este paso se puede omitir para utilizar la configuración predeterminada.

3. Implemente la configuración predeterminada: el server de Zabbix, la interfaz web (ejecutada en NGINX) y un backend de base de datos (MySQL o PostgreSQL), cada uno ejecutándose en su propio contenedor en Alpine Linux.

```
#### Con MySQL como base de datos:
docker compose -f ./compose.yaml up -d

#### Con PostgreSQL como base de datos:
docker compose -f ./compose_pgsql.yaml up -d
```

Una vez que los contenedores estén activos y en ejecución (normalmente en un plazo de 1 a 3 minutos), el server de Zabbix se inicia y la interfaz web estará disponible para iniciar sesión.

> **Note:**
> Use `docker compose ps` para comprobar el estado de los contenedores. Todos los contenedores (excepto el contenedor `zabbix-docker-server-db-init-1`) deben tener el estado `Up`. Si alguno muestra `Exited`, compruebe sus registros para detectar errores con `docker logs -f <container-name>`.

Variables de entorno

El comportamiento de Docker Compose y la configuración de los componentes de Zabbix se pueden personalizar mediante variables de entorno.

Las **variables de nivel de Compose** (definidas en el archivo `.env`) controlan qué imágenes de Docker, puertos y rangos de IP de red se utilizan. Puede usar estas variables en línea antes del comando `docker compose` o editar el archivo `.env`.

Por ejemplo, el siguiente comando implementa una configuración completa de Zabbix con varios contenedores utilizando una imagen basada en Ubuntu (en lugar de Alpine Linux, que es la opción predeterminada) y expone la interfaz web mediante NGINX en puertos HTTP (8282) y HTTPS (8443) personalizados:

```
OS=ubuntu \
ZABBIX_WEB_NGINX_HTTP_PORT=8282 \
ZABBIX_WEB_NGINX_HTTPS_PORT=8443 \
docker compose -f ./compose.yaml up -d
```

Las **variables de nivel de componente** (definidas en los archivos `env_vars/.env_<component>`) controlan la configuración de los componentes de Zabbix. Edite el archivo `.env_<component>` correspondiente antes de ejecutar el comando `docker compose`.

Por ejemplo, puede ajustar el número de pollers pasivos y la configuración de la caché del server de Zabbix, así como configurar la zona horaria de la interfaz web, editando las siguientes variables:

```
#### env_vars/.env_srv
ZBX_STARTPOLLERS=20
ZBX_CACHESIZE=64M
ZBX_HISTORYCACHESIZE=32M
```

```
#### env_vars/.env_web
PHP_TZ=Europe/Paris
```

Note:
Las variables de entorno de nivel de componente corresponden a los parámetros de configuración de los componentes de Zabbix, pero utilizan un estilo de nomenclatura diferente (por ejemplo, ZBX_STARTPOLLERS corresponde a StartPollers). Algunas variables son específicas de Docker y algunos parámetros de configuración no se pueden cambiar (por ejemplo, PIDFile y LogType). Al utilizar variables de entorno, consulte la sección *Variables de entorno* de la descripción de la imagen de Docker de cada componente en Docker Hub.

Volúmenes

Docker Compose almacena los datos persistentes en el directorio zbx_env/, creado junto al archivo Compose. Este directorio conserva los datos de los componentes entre reinicios y actualizaciones de los contenedores.

El contenido de zbx_env/ está predefinido para la imagen de cada componente. Por ejemplo:

- zbx_env/etc/ - utilizado por el contenedor de la interfaz web de Zabbix
- zbx_env/usr/ y zbx_env/var/ - utilizados por el contenedor del server de Zabbix

Para obtener más información sobre cada volumen, consulte la sección *Allowed volumes* en la descripción de la imagen de Docker de cada componente en Docker Hub.

Makefile

El repositorio de Dockerfiles de Zabbix también proporciona un Makefile como acceso directo para las tareas comunes de Docker Compose. En lugar de utilizar el comando completo docker compose, puede usar comandos make más cortos (ejecute make help para ver las opciones disponibles):

```
#### Implementación predeterminada (server de Zabbix, interfaz web, MySQL, todo en Alpine Linux)
make up

#### Implementación personalizada (server de Zabbix, interfaz web con puertos NGINX personalizados, Postgr
make up \
  OS=ubuntu \
  DB=pgsql \
  ZABBIX_WEB_NGINX_HTTP_PORT=8282 \
  ZABBIX_WEB_NGINX_HTTPS_PORT=8443
```

Attention:
Al detener o eliminar contenedores, especifique siempre el mismo tipo de base de datos que se implementó (por ejemplo, make down DB=pgsql).

De forma predeterminada, make up inicia solo un conjunto mínimo de servicios (server de Zabbix, interfaz web y base de datos). Esto mantiene la configuración ligera y evita iniciar componentes que podrían no ser necesarios. Para incluir componentes adicionales, puede utilizar los siguientes perfiles de Compose:

```
#### Implementación predeterminada + Zabbix agent, Java gateway, servicio web y traps SNMP:
make up COMPOSE_PROFILES=full

#### COMPOSE_PROFILES=full + proxy de Zabbix (MySQL y SQLite3):
make up COMPOSE_PROFILES=all
```

Docker (implementación manual)

Utilice la implementación manual cuando desee desplegar Zabbix paso a paso, ejecutar componentes individuales, integrarlos con un entorno existente o usar un runtime de contenedores alternativo como Podman.

Por ejemplo, para desplegar un proxy de Zabbix, ejecute el siguiente comando:

```
docker run --name zabbix-proxy-sqlite3 \
  -e ZBX_SERVER_HOST=192.0.2.1 \
  -e ZBX_PROXYMODE=0 \
  -e ZBX_HOSTNAME=zabbix-proxy-sqlite3 \
  -v zabbix-proxy-data:/var/lib/zabbix/db_data \
  --init \
```

```
  -d \
  zabbix/zabbix-proxy-sqlite3:alpine-8.0-latest
```

Este comando:

- Descarga la imagen de Docker `zabbix/zabbix-proxy-sqlite3:alpine-8.0-latest`.
- Crea e inicia un contenedor `zabbix-proxy-sqlite3` basado en la imagen de Docker (con las opciones `--init` y `-d`).
- Configura el parámetro `Server` del proxy de Zabbix mediante la variable de entorno `ZBX_SERVER_HOST`. Este parámetro especifica la dirección IP del server de Zabbix desde el cual el proxy recuperará los datos de configuración y al cual enviará los datos recopilados. Tenga en cuenta que otras variables necesarias para el funcionamiento del proxy activo usan valores predeterminados y pueden omitirse.

Note:
Las variables de entorno corresponden a los parámetros de configuración de los componentes de Zabbix mediante un estilo de nomenclatura diferente (por ejemplo, `ZBX_SERVER_HOST` corresponde a `Server`). Algunas variables son específicas de Docker, y algunos parámetros de configuración no se pueden cambiar (por ejemplo, `PIDFile` y `LogType`). Al usar variables de entorno, consulte la sección *Environment variables* en la descripción de la imagen de Docker de cada componente en Docker Hub.

- Conecta una ubicación de almacenamiento administrada por Docker (por ejemplo, `/var/lib/docker/volumes/zabbix-proxy-dat` con el directorio `/var/lib/zabbix` del contenedor, de modo que los datos del proxy de Zabbix se almacenen de forma permanente, incluso si el contenedor se elimina.

Note:
Para obtener más información sobre cada volumen, consulte la sección *Allowed volumes* en la descripción de la imagen de Docker de cada componente en Docker Hub.

Después de desplegar el contenedor del proxy de Zabbix, puede continuar con añadir el proxy en su interfaz web de Zabbix y configurar hosts para que sean monitorizados por este proxy.

Los ejemplos siguientes cubren tres escenarios adicionales de implementación:

- server de Zabbix (MySQL) con Java gateway
- server de Zabbix (PostgreSQL) con trampas SNMP
- server de Zabbix (MySQL) con Java gateway en RHEL 8-10

Para más ejemplos, consulte la descripción de la imagen de Docker de cada componente en Docker Hub.

Implementar Zabbix server (MySQL) con Java gateway

Este ejemplo muestra cómo implementar un Zabbix server con un backend MySQL, un Java gateway para la supervisión de JMX y una interfaz web basada en NGINX.

1. Cree una red Docker dedicada para que todos los contenedores de los componentes de Zabbix puedan comunicarse entre sí mediante el nombre del contenedor:

```
docker network create --subnet 172.20.0.0/16 --ip-range 172.20.240.0/20 zabbix-net
```

2. Ejecute un contenedor vacío de MySQL server (sustituya `zabbix_pwd` y `root_pwd` por contraseñas seguras y utilice esos valores en los pasos siguientes):

```
docker run --name mysql-server -t \
  -e MYSQL_DATABASE="zabbix" \
  -e MYSQL_USER="zabbix" \
  -e MYSQL_PASSWORD="zabbix_pwd" \
  -e MYSQL_ROOT_PASSWORD="root_pwd" \
  --network=zabbix-net \
  --restart unless-stopped \
  -d mysql:8.4-oracle \
  --character-set-server=utf8mb4 --collation-server=utf8mb4_bin
```

3. Siga los registros de MySQL y espere hasta que MySQL esté listo para aceptar conexiones (utilice `Ctrl+C` para salir de los registros):

```
docker logs -f mysql-server

#### ... [Server] /usr/sbin/mysqld: ready for connections. ...
```

4. Ejecute un contenedor de Zabbix Java gateway:

```
docker run --name zabbix-java-gateway -t \
  --network=zabbix-net \
  --restart unless-stopped \
  -d zabbix/zabbix-java-gateway:alpine-8.0-latest
```

5. Habilite `log_bin_trust_function_creators` en el contenedor de MySQL server. Esto relaja temporalmente las restricciones de seguridad sobre las funciones almacenadas; la configuración se deshabilitará en pasos posteriores. Permite que el usuario de la base de datos de Zabbix cree funciones almacenadas sin el privilegio `SUPER`, que MySQL requiere cuando el registro binario está habilitado (de forma predeterminada desde MySQL 8.0). Ejecute el siguiente comando:

```
docker exec -it mysql-server mysql -u root -p'root_pwd' \
  -e "SET GLOBAL log_bin_trust_function_creators = 1;"
```

6. Ejecute un contenedor de Zabbix server y vincúlelo a los contenedores de Java Gateway y MySQL server:

```
docker run --name zabbix-server-mysql -t \
  -e DB_SERVER_HOST="mysql-server" \
  -e MYSQL_DATABASE="zabbix" \
  -e MYSQL_USER="zabbix" \
  -e MYSQL_PASSWORD="zabbix_pwd" \
  -e ZBX_JAVAGATEWAY="zabbix-java-gateway" \
  --network=zabbix-net \
  -p 10051:10051 \
  --restart unless-stopped \
  --init \
  -d zabbix/zabbix-server-mysql:alpine-8.0-latest
```

7. Siga los registros de Zabbix server y espere hasta que Zabbix server haya terminado de inicializar el esquema de la base de datos (utilice `Ctrl+C` para salir de los registros):

```
docker logs -f zabbix-server-mysql

#### ... [info]: ** Creating 'zabbix' schema in MySQL
#### ... [info]: ** Database schema successfully created!
```

8. Deshabilite `log_bin_trust_function_creators` en el contenedor de MySQL server. Esto vuelve a aplicar la restricción de seguridad que impide a los usuarios que no tienen `SUPER` crear funciones almacenadas. Ejecute el siguiente comando:

```
docker exec -it mysql-server mysql -u root -p'root_pwd' \
  -e "SET GLOBAL log_bin_trust_function_creators = 0;"
```

9. Ejecute un contenedor de web frontend de Zabbix y vincúlelo a los contenedores de Zabbix server y MySQL server:

```
docker run --name zabbix-web-nginx-mysql -t \
  -e ZBX_SERVER_HOST="zabbix-server-mysql" \
  -e DB_SERVER_HOST="mysql-server" \
  -e MYSQL_DATABASE="zabbix" \
  -e MYSQL_USER="zabbix" \
  -e MYSQL_PASSWORD="zabbix_pwd" \
  -e PHP_TZ="Europe/Riga" \
  --network=zabbix-net \
  -p 80:8080 \
  --restart unless-stopped \
  -d zabbix/zabbix-web-nginx-mysql:alpine-8.0-latest
```

Una vez que los contenedores estén activos y en ejecución, Zabbix server se iniciará y el web frontend estará disponible para iniciar sesión.

> **Note:**
> Utilice `docker ps` para comprobar el estado de los contenedores. Todos los contenedores deben mostrar el estado `Up`. Si alguno muestra `Exited`, compruebe sus registros para detectar errores con `docker logs -f <container-name>`.

Implementar Zabbix server (PostgreSQL) con traps SNMP

Este ejemplo muestra cómo implementar un Zabbix server con un backend PostgreSQL, traps SNMP y una interfaz web basada en NGINX.

1. Cree una red Docker dedicada para que todos los contenedores de los componentes de Zabbix puedan comunicarse entre sí mediante el nombre del contenedor:

```
docker network create --subnet 172.20.0.0/16 --ip-range 172.20.240.0/20 zabbix-net
```

2. Ejecute un contenedor vacío de PostgreSQL server (sustituya `zabbix_pwd` por una contraseña segura y utilice ese valor en los pasos siguientes):

```
docker run --name postgres-server -t \
  -e POSTGRES_USER="zabbix" \
  -e POSTGRES_PASSWORD="zabbix_pwd" \
  -e POSTGRES_DB="zabbix" \
  --network=zabbix-net \
  --restart unless-stopped \
  -d postgres:latest
```

3. Ejecute un contenedor de traps SNMP de Zabbix:

```
docker run --name zabbix-snmptraps -t \
  -v /zbx_instance/snmptraps:/var/lib/zabbix/snmptraps:rw \
  -v /var/lib/zabbix/mibs:/usr/share/snmp/mibs:ro \
  --network=zabbix-net \
  -p 162:1162/udp \
  --restart unless-stopped \
  -d zabbix/zabbix-snmptraps:alpine-8.0-latest
```

4. Ejecute un contenedor de Zabbix server y vincúlelo a los contenedores de PostgreSQL server y traps SNMP:

```
docker run --name zabbix-server-pgsql -t \
  -e DB_SERVER_HOST="postgres-server" \
  -e POSTGRES_USER="zabbix" \
  -e POSTGRES_PASSWORD="zabbix_pwd" \
  -e POSTGRES_DB="zabbix" \
  -e ZBX_ENABLE_SNMP_TRAPS="true" \
  --network=zabbix-net \
  -p 10051:10051 \
  --volumes-from zabbix-snmptraps \
  --restart unless-stopped \
  --init \
  -d zabbix/zabbix-server-pgsql:alpine-8.0-latest
```

5. Siga los registros de Zabbix server y espere hasta que Zabbix server haya terminado de inicializar el esquema de la base de datos (utilice `Ctrl+C` para salir de los registros):

```
docker logs -f zabbix-server-pgsql

#### ... [info]: ** Creating 'zabbix' schema in PostgreSQL
#### ... [info]: ** Database schema successfully created!
```

6. Ejecute un contenedor de web frontend de Zabbix y vincúlelo a los contenedores de Zabbix server y PostgreSQL server:

```
docker run --name zabbix-web-nginx-pgsql -t \
  -e ZBX_SERVER_HOST="zabbix-server-pgsql" \
  -e DB_SERVER_HOST="postgres-server" \
  -e POSTGRES_DB="zabbix" \
  -e POSTGRES_USER="zabbix" \
  -e POSTGRES_PASSWORD="zabbix_pwd" \
  -e PHP_TZ="Europe/Riga" \
  --network=zabbix-net \
  -p 443:8443 \
  -p 80:8080 \
  -v /etc/ssl/nginx:/etc/ssl/nginx:ro \
  --restart unless-stopped \
  -d zabbix/zabbix-web-nginx-pgsql:alpine-8.0-latest
```

Una vez que los contenedores estén activos y en ejecución, Zabbix server se iniciará y el web frontend estará disponible para iniciar sesión.

> **Note:**
> Utilice `docker ps` para comprobar el estado de los contenedores. Todos los contenedores deben mostrar el estado Up. Si alguno muestra Exited, compruebe sus registros para detectar errores con `docker logs -f <container-name>`.

Implementar Zabbix server (MySQL) con Java gateway en RHEL 8-10

Este ejemplo muestra cómo implementar un Zabbix server con un backend MySQL, un Java gateway para la supervisión de JMX y una interfaz web basada en NGINX, todo ello ejecutándose en Red Hat Enterprise Linux 8, 9 o 10.

En Red Hat Enterprise Linux, el runtime de contenedores recomendado es Podman en lugar de Docker. Podman funciona de forma similar a Docker, pero no requiere un servicio en segundo plano ejecutándose como root, por lo que se adapta mejor a los entornos Red Hat.

1. Cree un pod nuevo con el nombre `zabbix` y exponga los puertos para el web frontend de Zabbix y el trapper de Zabbix server:

```
podman pod create --name zabbix -p 80:8080 -p 10051:10051
```

2. (Opcional) Ejecute un contenedor de Zabbix agent en el pod `zabbix`:

```
podman run --name zabbix-agent \
  -e ZBX_SERVER_HOST="127.0.0.1,localhost" \
  --restart=always \
  --pod=zabbix \
  -d registry.connect.redhat.com/zabbix/zabbix-agent-80:latest
```

3. Cree un directorio `./mysql/` en el host RHEL:

```
mkdir -p ./mysql
```

4. Ejecute un contenedor vacío de MySQL server (sustituya `zabbix_pwd` y `root_pwd` por contraseñas seguras y utilice esos valores en los pasos siguientes):

```
podman run --name mysql-server -t \
  -e MYSQL_DATABASE="zabbix" \
  -e MYSQL_USER="zabbix" \
  -e MYSQL_PASSWORD="zabbix_pwd" \
  -e MYSQL_ROOT_PASSWORD="root_pwd" \
  -v ./mysql/:/var/lib/mysql/:Z \
  --restart=always \
  --pod=zabbix \
  -d mysql:8.4 \
  --character-set-server=utf8mb4 --collation-server=utf8mb4_bin
```

5. Siga los registros de MySQL y espere hasta que MySQL esté listo para aceptar conexiones (utilice `Ctrl+C` para salir de los registros):

```
podman logs -f mysql-server

#### ... [Server] /usr/sbin/mysqld: ready for connections. ...
```

6. Ejecute un contenedor de Zabbix Java gateway:

```
podman run --name zabbix-java-gateway -t \
  --restart=always \
  --pod=zabbix \
  -d registry.connect.redhat.com/zabbix/zabbix-java-gateway-80
```

7. Habilite `log_bin_trust_function_creators` en el contenedor de MySQL server. Esto relaja temporalmente las restricciones de seguridad sobre las funciones almacenadas; la configuración se deshabilitará en pasos posteriores. Permite que el usuario de la base de datos de Zabbix cree funciones almacenadas sin el privilegio `SUPER`, que MySQL requiere cuando el registro binario está habilitado (de forma predeterminada desde MySQL 8.0). Ejecute el siguiente comando:

```
podman exec -it mysql-server mysql -u root -p'root_pwd' \
  -e "SET GLOBAL log_bin_trust_function_creators = 1;"
```

8. Ejecute un contenedor de Zabbix server y vincúlelo a los contenedores de Java Gateway y MySQL server:

```
podman run --name zabbix-server-mysql -t \
  -e DB_SERVER_HOST="127.0.0.1" \
```

```
  -e MYSQL_DATABASE="zabbix" \
  -e MYSQL_USER="zabbix" \
  -e MYSQL_PASSWORD="zabbix_pwd" \
  -e ZBX_JAVAGATEWAY="127.0.0.1" \
  --restart=always \
  --init \
  --pod=zabbix \
  -d registry.connect.redhat.com/zabbix/zabbix-server-mysql-80
```

9. Siga los registros de Zabbix server y espere hasta que Zabbix server haya terminado de inicializar el esquema de la base de datos (utilice Ctrl+C para salir de los registros):

```
podman logs -f zabbix-server-mysql

#### ... [info]: ** Creating 'zabbix' schema in MySQL
#### ... [info]: ** Database schema successfully created!
```

10. Deshabilite log_bin_trust_function_creators en el contenedor de MySQL server. Esto vuelve a aplicar la restricción de seguridad que impide a los usuarios que no tienen SUPER crear funciones almacenadas. Ejecute el siguiente comando:

```
podman exec -it mysql-server mysql -u root -p'root_pwd' \
  -e "SET GLOBAL log_bin_trust_function_creators = 0;"
```

11. Ejecute un contenedor de web frontend de Zabbix y vincúlelo a los contenedores de Zabbix server y MySQL server:

```
podman run --name zabbix-web-mysql -t \
  -e ZBX_SERVER_HOST="127.0.0.1" \
  -e DB_SERVER_HOST="127.0.0.1" \
  -e MYSQL_DATABASE="zabbix" \
  -e MYSQL_USER="zabbix" \
  -e MYSQL_PASSWORD="zabbix_pwd" \
  -e PHP_TZ="Europe/Riga" \
  --restart=always \
  --pod=zabbix \
  -d registry.connect.redhat.com/zabbix/zabbix-web-mysql-80
```

Note:
El pod zabbix expone el puerto 80/TCP (HTTP) a la máquina host desde el puerto 8080/TCP del contenedor zabbix-web-mysql.

Una vez que los contenedores estén activos y en ejecución, Zabbix server se iniciará y el web frontend estará disponible para iniciar sesión.

Note:
Utilice `docker ps` para comprobar el estado de los contenedores. Todos los contenedores deben mostrar el estado Up. Si alguno muestra Exited, compruebe sus registros para detectar errores con `docker logs -f <container-name>`.

Instalación desde fuentes

Puede obtener la última versión de Zabbix compilándola desde las fuentes. Consulte también Obtener el código fuente de Zabbix.

Aquí se proporciona un tutorial paso a paso para instalar Zabbix desde las fuentes.

Instalación de los demonios de Zabbix

1 Descargar el archivo fuente

Vaya a la página de descargas de Zabbix y descargue el archivo fuente. Una vez descargado, extraiga las fuentes ejecutando:

```
tar -zxvf zabbix-8.0.0.tar.gz
```

Note:
Introduzca la versión correcta de Zabbix en el comando. Debe coincidir con el nombre del archivo descargado.

2 Crear cuenta de usuario

Todos los procesos demonio de Zabbix se ejecutan bajo usuarios del sistema sin privilegios. Si un demonio de Zabbix se inicia desde una cuenta de usuario sin privilegios, continuará ejecutándose como ese usuario.

En la configuración predeterminada, si un demonio se inicia como `root`, cambiará a la cuenta de usuario `zabbix`, que debe estar presente. Para crear un usuario y grupo de sistema `zabbix`, ejecute los comandos que se indican a continuación.

Sistema basado en RedHat:

```
groupadd --system zabbix
useradd --system -g zabbix -d /usr/lib/zabbix -s /sbin/nologin -c "Zabbix Monitoring System" zabbix
```

Sistema basado en Debian:

```
addgroup --system --quiet zabbix
adduser --quiet --system --disabled-login --ingroup zabbix --home /var/lib/zabbix --no-create-home zabbix
```

No es necesario crear una cuenta de usuario separada para el frontend de Zabbix.

Recomendación de seguridad

Si el server y el agent de Zabbix se ejecutan en la misma máquina, se recomienda ejecutarlos bajo **cuentas de usuario separadas**. Ejecutar ambos como el mismo usuario permite que el agent acceda al archivo de configuración del server, lo que podría exponer información confidencial, como la contraseña de la base de datos, a cualquier usuario con nivel de Admin en Zabbix.

> **Attention:**
> Ejecutar Zabbix como `root`, `bin` o cualquier otra cuenta con derechos especiales es un riesgo de seguridad.

Directorio home (opcional)

Los procesos de Zabbix no requieren un directorio home, por lo que generalmente no se recomienda crearlo. Sin embargo, si necesita una funcionalidad que requiera un directorio home (por ejemplo, almacenar credenciales de MySQL en $HOME/.my.cnf), puede crearlo utilizando los comandos que se indican a continuación.

En sistemas basados en RedHat, ejecute:

```
mkdir -m u=rwx,g=rwx,o= -p /usr/lib/zabbix
chown zabbix:zabbix /usr/lib/zabbix
```

En sistemas basados en Debian, ejecute:

```
mkdir -m u=rwx,g=rwx,o= -p /var/lib/zabbix
chown zabbix:zabbix /var/lib/zabbix
```

3 Crear la base de datos de Zabbix

Para los demonios de Zabbix server y proxy, así como para el frontend de Zabbix, se requiere una base de datos. No es necesario para ejecutar el agent de Zabbix.

Se proporcionan scripts SQL para crear el esquema de la base de datos e insertar el conjunto de datos. La base de datos del proxy de Zabbix solo necesita el esquema, mientras que la base de datos del server de Zabbix requiere también el conjunto de datos además del esquema.

Una vez creada la base de datos de Zabbix, continúe con los siguientes pasos para compilar Zabbix.

4 Configurar las fuentes

Se requiere C99 con extensiones GNU para compilar Zabbix server, Zabbix proxy o Zabbix agent. Esta versión se puede especificar explícitamente estableciendo CFLAGS=”-std=gnu99”:

```
export CFLAGS="-std=gnu99"
```

> **Note:**
> Si se instala desde el repositorio Git de Zabbix, es necesario ejecutar primero:
> `./bootstrap.sh`

Al configurar las fuentes para un Zabbix server o proxy, debe especificar el tipo de base de datos que se utilizará. Solo se puede compilar un tipo de base de datos con un proceso de server o proxy a la vez.

Para ver todas las opciones de configuración compatibles, dentro del directorio de fuentes de Zabbix extraído, ejecute:

```
./configure --help
```

Para configurar las fuentes para un Zabbix server y agent, puede ejecutar algo como:

```
./configure --enable-server --enable-agent --with-mysql --enable-ipv6 --with-net-snmp --with-libcurl --wit
```

Para configurar las fuentes para un Zabbix server (con PostgreSQL, etc.), puede ejecutar:

```
./configure --enable-server --with-postgresql --with-net-snmp
```

Para configurar las fuentes para un Zabbix proxy (con SQLite, etc.), puede ejecutar:

```
./configure --prefix=/usr --enable-proxy --with-net-snmp --with-sqlite3 --with-ssh2
```

Para configurar las fuentes para un Zabbix agent, puede ejecutar:

```
./configure --enable-agent
```

o, para Zabbix agent 2:

```
./configure --enable-agent2
```

Note:
Debe instalarse una versión compatible de Go para compilar Zabbix agent 2.

Notas sobre las opciones de compilación:

- `--enable-agent`: compila Zabbix agent, así como las utilidades de línea de comandos Zabbix get y Zabbix sender.
- `--with-libcurl`: necesario para la monitorización de máquinas virtuales, la autenticación SMTP y los items de Zabbix agent `web.page.*`, así como para los items de consulta de telemetría. Consulte también: Requisitos (libcurl).
- `--with-libxml2`: necesario para la monitorización de máquinas virtuales.
- `--with-libpcre2[=DIR]`: Zabbix siempre se compila con la biblioteca PCRE2; esta opción solo permite especificar una ruta de instalación personalizada de PCRE2.
- `--with-mysql=/path/to/mysql_config`: especifica la ruta a la configuración de una biblioteca cliente de MySQL concreta. Es útil cuando hay instaladas varias versiones de MySQL o MariaDB.
- `--enable-static`: enlaza estáticamente las bibliotecas (no compatible con Solaris). Utilice esta opción si planea distribuir binarios compilados a sistemas que no dispongan de las bibliotecas necesarias. No se recomienda al compilar Zabbix server. Para compilar el server estáticamente, se requiere una versión estática de cada biblioteca externa. El script configure no comprueba esto automáticamente.
- `--with-stacksize=<value>`: establece el tamaño de la pila por subproceso en kilobytes (por ejemplo, `--with-stacksize=512`). Puede aumentar este valor si Zabbix se bloquea o se congela debido a desbordamientos de pila (por ejemplo, durante el preprocesamiento en sistemas con límites de pila predeterminados bajos para los subprocesos).
- `--with-apm` (usar solo con `--enable-proxy`): necesario para la monitorización del rendimiento de aplicaciones (APM) en Zabbix proxy. Consulte también: Requisitos (g++, gRPC, protobuf, Abseil).

Attention:
Si `./configure` falla debido a bibliotecas faltantes u otros problemas, compruebe el archivo `config.log` para obtener información detallada sobre el error.
Por ejemplo, si falta `libssl`, el mensaje de error inmediato puede resultar engañoso:

```
checking for main in -lmysqlclient... no
configure: error: Not found mysqlclient library
```

En este caso, `config.log` muestra la causa real:

```
/usr/bin/ld: cannot find -lssl
/usr/bin/ld: cannot find -lcrypto
```

Consulte también:

- Compilar Zabbix con compatibilidad con el cifrado
- Problemas de compilación conocidos.

5 Compilar e instalar todo

Note:
Si se instala desde Zabbix Git repository, es necesario ejecutar primero:

```
$ make dbschema
```

```
make install
```

Este paso debe ejecutarse como un usuario con permisos suficientes (normalmente 'root', o usando `sudo`).

Al ejecutar `make install`, de forma predeterminada se instalarán los binarios del daemon (zabbix_server, zabbix_agentd, zabbix_proxy) en /usr/local/sbin y los binarios del cliente (zabbix_get, zabbix_sender) en /usr/local/bin.

Note:
Para especificar una ubicación distinta de `/usr/local`, use una clave `--prefix` en el paso anterior de configuración de las fuentes, por ejemplo, `--prefix=/home/zabbix`. En este caso, los binarios del daemon se instalarán en `<prefix>/sbin`, mientras que las utilidades se instalarán en `<prefix>/bin`. Las páginas de manual se instalarán en `<prefix>/share`.

6 Revisar y editar los archivos de configuración

- edite el archivo de configuración del agent de Zabbix (`zabbix_agentd.conf`).

Debe configurar este archivo para cada host donde esté instalado `zabbix_agentd`.

Debe especificar la **dirección IP** del server de Zabbix en el archivo. Se rechazarán las conexiones de otros hosts.

- edite el archivo de configuración del server de Zabbix (`zabbix_server.conf`).

Debe especificar el nombre de la base de datos, el usuario y la contraseña (si se utiliza).

El resto de los parámetros se ajustará a sus necesidades con sus valores predeterminados si tiene una instalación pequeña (hasta diez hosts monitorizados). Sin embargo, debe cambiar los parámetros predeterminados si desea maximizar el rendimiento del server (o proxy) de Zabbix.

- si ha instalado un proxy de Zabbix, edite el archivo de configuración del proxy (`zabbix_proxy.conf`).

Debe especificar la dirección IP del server y el nombre de host del proxy (que el server debe conocer), así como el nombre de la base de datos, el usuario y la contraseña (si se utiliza).

Note:
Con SQLite, debe especificarse la ruta completa al archivo de la base de datos; no se requieren el usuario ni la contraseña de la base de datos.

7 Iniciar los demonios

Ejecute zabbix_server en el lado del servidor.

```
zabbix_server
```

Note:
Asegúrese de que su sistema permita la asignación de 36MB (o un poco más) de memoria compartida, de lo contrario el servidor puede no iniciarse y verá "Cannot allocate shared memory for <type of cache>." en el archivo de registro del servidor. Esto puede ocurrir en FreeBSD, Solaris 8.

Ejecute zabbix_agentd en todas las máquinas monitorizadas.

```
zabbix_agentd
```

Note:
Asegúrese de que su sistema permita la asignación de 2MB de memoria compartida, de lo contrario el agent puede no iniciarse y verá "Cannot allocate shared memory for collector." en el archivo de registro del agent. Esto puede ocurrir en Solaris 8.

Si ha instalado Zabbix proxy, ejecute zabbix_proxy.

```
zabbix_proxy
```

Instalación de la interfaz web de Zabbix

Copia de archivos PHP

El frontend de Zabbix está escrito en PHP, por lo que para ejecutarlo se necesita un servidor web compatible con PHP. La instalación se realiza simplemente copiando los archivos PHP del directorio ui al directorio de documentos HTML del servidor web.

Las ubicaciones comunes de los directorios de documentos HTML para servidores web Apache incluyen:

- /usr/local/apache2/htdocs (directorio predeterminado al instalar Apache desde el código fuente)
- /srv/www/htdocs (OpenSUSE, SLES)
- /var/www/html (Debian, Ubuntu, Fedora, RHEL)

Se recomienda usar un subdirectorio en lugar de la raíz HTML. Para crear un subdirectorio y copiar en él los archivos del frontend de Zabbix, ejecute los siguientes comandos, reemplazando <htdocs> por el directorio real:

```
mkdir <htdocs>/zabbix
cd ui
cp -a . <htdocs>/zabbix
```

Si planea usar cualquier otro idioma que no sea inglés, consulte Instalación de idiomas adicionales del frontend para obtener instrucciones.

Instalación del frontend

Consulte la página Instalación de la interfaz web para obtener información sobre el asistente de instalación del frontend de Zabbix.

Instalación de plugins cargables de Zabbix agent 2

La instalación de plugins cargables de Zabbix agent 2 solo es necesaria si desea supervisar objetivos que no están cubiertos por los plugins integrados (por ejemplo, servidores o clústeres de MongoDB, PostgreSQL y sus bifurcaciones, etc.). Consulte la lista completa de plugins cargables y plugins integrados.

Attention:
Antes de instalar un plugin, revise su archivo README. Puede contener requisitos específicos e instrucciones de instalación.

Para instalar desde el código fuente, primero descargue y extraiga el archivo fuente del plugin cargable.

Para compilar el plugin, vaya al directorio del plugin extraído y ejecute `make`:

```
make
```

Note:
Debe estar instalada una versión compatible de Go para compilar plugins cargables de Zabbix agent 2.

El ejecutable del plugin puede colocarse en cualquier ubicación, siempre que Zabbix agent 2 pueda cargarlo. Especifique la ruta al binario del plugin en el archivo de configuración del plugin, por ejemplo, en `postgresql.conf` para el plugin PostgreSQL:

```
Plugins.PostgreSQL.System.Path=/path/to/executable/zabbix-agent2-plugin-postgresql
```

La ruta al archivo de configuración del plugin debe especificarse en el parámetro `Include` del archivo de configuración de Zabbix agent 2:

```
Include=/path/to/plugin/configuration/file/postgresql.conf
```

Continúe con configuración para obtener más detalles sobre la configuración de plugins.

Los plugins cargables proporcionados por Zabbix usan makefiles simples con los siguientes objetivos de compilación:

- `make` - compilar el plugin
- `make clean` - eliminar todos los archivos creados al compilar el plugin
- `make check` - ejecutar autopruebas (requiere un objetivo de supervisión real, por ejemplo, una base de datos PostgreSQL)
- `make style` - comprobar el estilo del código Go con `golangci-lint`
- `make format` - formatear el código Go con `go fmt`
- `make dist` - crear un archivo fuente que incluya todas las dependencias

Instalación de Java gateway

Solo es necesario instalar Java gateway si desea supervisar aplicaciones JMX. Java gateway es ligero y no requiere una base de datos.

Para instalar desde el código fuente, primero descargue y extraiga el archivo fuente.

Para compilar Java gateway, ejecute el script `./configure` con la opción `--enable-java`. Se recomienda especificar la opción `--prefix` para indicar una ruta de instalación distinta de la predeterminada /usr/local, porque la instalación de Java gateway creará todo un árbol de directorios, no solo un único ejecutable.

```
./configure --enable-java --prefix=$PREFIX
```

Para compilar y empaquetar Java gateway en un archivo JAR, ejecute `make`. Tenga en cuenta que para este paso necesitará los ejecutables `javac` y `jar` en su ruta.

```
make
```

Ahora tiene un archivo zabbix-java-gateway-$VERSION.jar en `src/zabbix_java/bin`. Si le resulta cómodo ejecutar Java gateway desde `src/zabbix_java` en el directorio de distribución, puede continuar con las instrucciones para configurar y ejecutar Java gateway. De lo contrario, asegúrese de tener suficientes privilegios y ejecute `make install`.

```
make install
```

Continúe con la configuración para obtener más detalles sobre cómo configurar y ejecutar Java gateway.

Instalación del servicio web de Zabbix

La instalación del servicio web de Zabbix solo es necesaria si desea usar informes programados.

Para instalar desde el código fuente, primero descargue y extraiga el archivo fuente.

Para compilar el servicio web de Zabbix, ejecute el script `./configure` con la opción `--enable-webservice`.

> **Note:**
> Debe estar instalada una versión compatible de Go para compilar el servicio web de Zabbix.

Ejecute zabbix_web_service en la máquina donde está instalado el servicio web:

```
zabbix_web_service
```

Continúe con la configuración para obtener más detalles sobre la configuración de la generación de informes programados.

Instalación del servidor MCP de Zabbix

Para instalar desde las fuentes, primero descargue y extraiga el archivo de fuentes.

Para compilar el servidor MCP de Zabbix, ejecute el siguiente comando en el directorio de fuentes del servidor MCP:

```
make build
```

> **Note:**
> Debe estar instalada una versión de Go compatible para compilar el servidor MCP de Zabbix.

Para indicar una versión específica en la salida de `zabbix_mcp_server -V` y en el `User-Agent` de las solicitudes a la API de Zabbix, establezca la variable `VERSION`, por ejemplo, `make build VERSION=8.0.0`.

Para instalar el binario en `/usr/local/sbin` y el archivo de configuración de ejemplo en `/etc/zabbix`, ejecute:

```
sudo make install
```

Edite el archivo de configuración `/etc/zabbix/zabbix_mcp_server.conf` y establezca al menos los siguientes parámetros:

- `FrontendURL`: la URL de la API de Zabbix, que termina en `api_jsonrpc.php`;
- `AllowedIP`: las direcciones de los clientes MCP a los que se permite conectarse.

Para cifrar las conexiones de los clientes MCP, establezca `TLSAccept=cert` y especifique `TLSCertFile` y `TLSKeyFile`, o ejecute el servidor MCP detrás de un proxy inverso de confianza que gestione TLS.

Si el usuario `zabbix` no existe (por ejemplo, si solo el servidor MCP está instalado en el equipo), créelo como se describe en Crear una cuenta de usuario.

Valide el archivo de configuración e inicie zabbix_mcp_server:

```
zabbix_mcp_server -c /etc/zabbix/zabbix_mcp_server.conf -T
zabbix_mcp_server -c /etc/zabbix/zabbix_mcp_server.conf
```

Consulte Servidor MCP para obtener más información sobre la configuración del servidor y la conexión de clientes MCP.

Obtener el código fuente de Zabbix

Existen varias formas de obtener el código fuente de Zabbix:

- Puede descargar las versiones estables publicadas desde el sitio web oficial de Zabbix
- Puede descargar compilaciones nocturnas desde la página de desarrolladores del sitio web oficial de Zabbix
- Puede obtener la última versión de desarrollo desde el sistema de repositorio de código fuente Git:
  - La ubicación principal del repositorio completo es https://git.zabbix.com/scm/zbx/zabbix.git
  - Las versiones master y soportadas también se reflejan en Github en https://github.com/zabbix/zabbix

Debe instalarse un cliente Git para clonar el repositorio. El paquete oficial del cliente Git de línea de comandos se llama comúnmente **git** en las distribuciones. Para instalarlo, por ejemplo, en Debian/Ubuntu, ejecute:

```
sudo apt-get update
sudo apt-get install git
```

Para obtener todo el código fuente de Zabbix, cambie al directorio donde desea colocar el código y ejecute:

```
git clone https://git.zabbix.com/scm/zbx/zabbix.git
```

Problemas de compilación

Estos son los problemas conocidos relacionados con la compilación de Zabbix desde fuentes. Para todos los demás casos, consulte la página de Problemas conocidos.

Biblioteca en una ubicación no estándar

Zabbix permite especificar una biblioteca ubicada en una ubicación no estándar. En el siguiente ejemplo, Zabbix ejecutará `curl-config` desde la ubicación no estándar especificada y usará su salida para determinar la libcurl correcta a utilizar.

```
$ ./configure --enable-server --with-mysql --with-libcurl=/usr/local/bin/curl-config
```

Esto funcionará si es la única libcurl instalada en el sistema, pero puede que no funcione si hay otra libcurl instalada en una ubicación estándar (por el gestor de paquetes, por ejemplo). Este es el caso cuando necesita una versión más reciente de la biblioteca para Zabbix y la anterior para otras aplicaciones.

Por lo tanto, especificar un componente en una ubicación no estándar no siempre funcionará cuando el mismo componente también exista en una ubicación estándar.

Por ejemplo, si utiliza una libcurl más reciente instalada en `/usr/local` con el paquete libcurl aún instalado, Zabbix podría elegir la incorrecta y la compilación fallará:

```
usr/bin/ld: ../../src/libs/zbxhttp/libzbxhttp.a(http.o): in function 'zbx_http_convert_to_utf8':
/tmp/zabbix-master/src/libs/zbxhttp/http.c:957: undefined reference to 'curl_easy_header'
collect2: error: ld returned 1 exit status
```

Aquí, la función `curl_easy_header()` no está disponible en la antigua `/usr/lib/x86_64-linux-gnu/libcurl.so`, pero sí está disponible en la nueva `/usr/local/lib/libcurl.so`.

El problema radica en el orden de las banderas del enlazador, y una solución es especificar la ruta completa a la biblioteca en una variable LDFLAGS:

```
$ LDFLAGS="-Wl,--no-as-needed /usr/local/lib/libcurl.so" ./configure --enable-server --with-mysql --with-l
```

Tenga en cuenta la opción `-Wl,--no-as-needed` que podría ser necesaria en algunos sistemas (ver también: opciones de enlace por defecto en sistemas basados en Debian).

Tamaño de pila demasiado pequeño en algunos sistemas

Si Zabbix se bloquea o se congela debido a desbordamientos de pila, puede aumentar el tamaño de la pila por hilo usando la opción `--with-stacksize` al configurar las fuentes. Este problema puede ocurrir en sistemas con límites predeterminados bajos para la pila de hilos, especialmente durante el preprocesamiento, donde se crean varios hilos.

El siguiente ejemplo establece el tamaño de pila en 512 KB por hilo:

```
./configure --enable-server --with-mysql --with-stacksize=512
```

Puede comprobar los límites de pila de hilos del sistema en tiempo de ejecución usando el comando `ulimit -s` en sistemas basados en Linux.

Notas sobre la instalación desde paquetes

Descripción general

Los paquetes oficiales de **prelanzamiento** de Zabbix 8.0 están disponibles en el sitio web de Zabbix, donde puede seleccionar su sistema operativo y el componente de Zabbix para generar instrucciones y comandos de instalación.

Además de las notas sobre la instalación de paquetes que se indican a continuación, consulte también la página Instalación y configuración.

Los paquetes están disponibles para las siguientes distribuciones de Linux:

- Red Hat Enterprise Linux y sus derivados: AlmaLinux, Amazon Linux 2023, CentOS Stream, CentOS 7, Oracle Linux, Rocky Linux
- Debian, Ubuntu, Raspberry Pi OS, Raspbian
- SUSE Linux Enterprise Server, openSUSE Leap

Attention:
Algunas distribuciones de SO, en particular las basadas en Debian, proporcionan sus propios paquetes de Zabbix. Estos paquetes **no cuentan con soporte de Zabbix** y pueden estar desactualizados o no incluir las funciones y correcciones de errores más recientes. Se recomienda utilizar únicamente los paquetes oficiales del repositorio oficial de Zabbix. Si instaló Zabbix anteriormente desde el repositorio de su sistema operativo, consulte los pasos para actualizar los paquetes de Zabbix desde los repositorios del SO.

Los paquetes son compatibles con bases de datos MySQL/PostgreSQL y servidores web Apache/Nginx. Tenga en cuenta que el server y el proxy de Zabbix no pueden compartir la misma base de datos; utilice nombres de base de datos diferentes si ambos están instalados en el mismo host.

Si es necesario, hay paquetes independientes para Zabbix agent/agent 2, Zabbix get y Zabbix sender disponibles en el repositorio oficial de Zabbix.

En los sistemas basados en Debian o Ubuntu, la instalación de un paquete adicional de Zabbix con `apt` también puede seleccionar actualizaciones para los paquetes de Zabbix ya instalados desde el mismo repositorio. Para instalar un paquete sin actualizar un paquete de server o proxy de Zabbix instalado, bloquee temporalmente dicho paquete antes de la instalación y desbloquéelo después:

```
apt-mark hold zabbix-server-mysql
apt install zabbix-sender
apt-mark unhold zabbix-server-mysql
```

Zabbix también proporciona binarios precompilados de Zabbix agent para sistemas operativos que no son Linux; consulte:

- Instalación del agent de Windows desde MSI
- Instalación del agent de macOS desde PKG
- Zabbix agent en IBM AIX
- Binarios heredados (para sistemas antiguos o menos comunes, como HP-UX, NetBSD, Tru64 y versiones anteriores de SLES)

Notas sobre la instalación de paquetes

Las siguientes notas se aplican a todos los sistemas:

- Si utiliza PostgreSQL, `DBHost=localhost` (o una dirección IP) en la configuración del server/proxy de Zabbix hace que PostgreSQL utilice un socket de red en lugar de un socket UNIX local; consulte Configuración de SELinux para obtener instrucciones de configuración relacionadas.
- Si utiliza TimescaleDB, consulte la configuración adicional de TimescaleDB.
- Si instala el Java gateway de Zabbix (para supervisar aplicaciones JMX), consulte las instrucciones de configuración adicionales para sistemas basados en RHEL y sistemas basados en Debian.
- Si instala el MCP server, instale el paquete `zabbix-mcp-server` y actualice el archivo de configuración (al menos `FrontendURL` y `AllowedIP`) antes de iniciar el servicio.
- Para ejecutar el Zabbix agent como root, consulte Ejecución del agent como root.
- La supervisión del rendimiento de aplicaciones (APM) en el Zabbix proxy es compatible con estos paquetes:
  - RHEL 10 y sus derivados: instale el paquete APM del Zabbix proxy (por ejemplo, `zabbix-proxy-mysql-apm` en lugar de `zabbix-proxy-mysql`); también es necesario instalar el repositorio EPEL y habilitar los repositorios de paquetes de dependencias CRB.
  - Debian 12 o posterior, Raspberry Pi OS (Raspbian) 12 o posterior, Ubuntu 24.04 o posterior: APM está incluido en los paquetes del Zabbix proxy.
  - En otros sistemas, puede instalar el Zabbix proxy desde las fuentes con la opción `--with-apm`.

Las siguientes notas se aplican a RHEL y sus derivados:

- Si ha habilitado el repositorio EPEL (por ejemplo, para el paquete APM del Zabbix proxy), que también proporciona paquetes de Zabbix, debe excluirlo de la resolución de paquetes antes de instalar los paquetes oficiales de Zabbix; consulte Instalación accidental de paquetes de Zabbix de EPEL.
- Para instalar paquetes de Zabbix en entornos Red Hat UBI, consulte Paquetes de Zabbix para RHEL en entornos Red Hat UBI.
- Para utilizar items de tipo ping ICMP, los paquetes de `fping` también están disponibles en el repositorio oficial de Zabbix.

Configuración de SELinux

Zabbix usa comunicación entre procesos basada en sockets. En sistemas donde Security-Enhanced Linux (SELinux) está habilitado, es posible que deba agregar reglas de SELinux para permitir que Zabbix cree/use sockets de dominio UNIX en el directorio SocketDir. Los archivos de socket son utilizados por Zabbix server (alerter, preprocessing, IPMI) y Zabbix proxy (IPMI), y están presentes mientras el proceso se está ejecutando.

Con SELinux habilitado en modo enforcing, ejecute los siguientes comandos para habilitar la comunicación entre Zabbix frontend y server:

Para RHEL 7 (y posteriores), AlmaLinux, CentOS Stream, Oracle Linux, Rocky Linux 8 (y posteriores):

```
setsebool -P httpd_can_connect_zabbix on
```

Si se accede a la base de datos a través de la red (incluido `localhost` para PostgreSQL), permita también que Zabbix frontend se conecte a la base de datos:

```
setsebool -P httpd_can_network_connect_db on
```

Para RHEL anteriores a la versión 7:

```
setsebool -P httpd_can_network_connect on
setsebool -P zabbix_can_network on
```

Después de aplicar la configuración de SELinux, reinicie Apache:

```
systemctl restart httpd
```

Opcionalmente, puede instalar un paquete `zabbix-selinux-policy` predefinido desde el Repositorio oficial de Zabbix. Este paquete se proporciona para todas las versiones de SO compatibles para simplificar la implementación de Zabbix y evitar que los usuarios desactiven SELinux debido a la complejidad de la configuración.
 Para la versión 10 de AlmaLinux, CentOS Stream, RHEL, Rocky Linux y Oracle Linux, la política de SELinux se proporciona mediante el paquete `zabbix-selinux-policy-main`; el paquete `zabbix-selinux-policy` tiene una dependencia de `zabbix-selinux-policy-main` o `selinux-policy-targeted-extra`. El paquete `zabbix-selinux-policy-main` entra en conflicto con `selinux-policy-targeted-extra`; por lo tanto, ambos paquetes no pueden instalarse al mismo tiempo.

> **Attention:**
> Para máxima seguridad, se recomienda establecer configuraciones personalizadas de SELinux.

El paquete `zabbix-selinux-policy` (`zabbix-selinux-policy-main` para la versión 10 de AlmaLinux, CentOS Stream, RHEL, Rocky Linux y Oracle Linux) contiene una política básica de SELinux que permite a Zabbix crear y usar sockets y habilita la conexión de HTTPd a PostgreSQL (utilizado por frontend).

El archivo fuente `zabbix_policy.te` contiene las siguientes reglas:

```
module zabbix_policy 1.2;

require {
    type zabbix_t;
    type zabbix_port_t;
    type zabbix_var_run_t;
    type postgresql_port_t;
    type httpd_t;
    class tcp_socket name_connect;
    class sock_file { create unlink };
    class unix_stream_socket connectto;
}

#####============= zabbix_t ==============
allow zabbix_t self:unix_stream_socket connectto;
allow zabbix_t zabbix_port_t:tcp_socket name_connect;
allow zabbix_t zabbix_var_run_t:sock_file create;
allow zabbix_t zabbix_var_run_t:sock_file unlink;
allow httpd_t zabbix_port_t:tcp_socket name_connect;

#####============= httpd_t ==============
allow httpd_t postgresql_port_t:tcp_socket name_connect;
```

Paquetes Debuginfo

Los paquetes `Debuginfo` contienen símbolos de depuración para los binarios de Zabbix. No son necesarios para una instalación u operación normales, pero son útiles para la resolución avanzada de problemas.

Tenga en cuenta que los paquetes `Debuginfo` están disponibles para las siguientes versiones: Red Hat Enterprise Linux 7 y anteriores (incluidas las derivadas), y SUSE Linux Enterprise Server 15 y anteriores (incluidas las derivadas).

Para habilitar el repositorio `zabbix-debuginfo`:

- En RHEL 7, edite `/etc/yum.repos.d/zabbix.repo` y establezca `enabled=1` para la sección `zabbix-debuginfo`:

```
[zabbix-debuginfo]
name=Zabbix Official Repository debuginfo - $basearch
baseurl=http://repo.zabbix.com/zabbix/8.0/stable/rhel/7/$basearch/debuginfo/
enabled=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-ZABBIX-A14FE591
gpgcheck=1
```

- En SUSE, edite `/etc/zypp/repos.d/zabbix.repo` y establezca `enabled=1` para la sección `zabbix-debuginfo`:

```
[zabbix-debuginfo]
name=Zabbix Official Repository debuginfo
type=rpm-md
baseurl=https://repo.zabbix.com/zabbix/8.0/unstable/sles/15/$basearch/debuginfo/
gpgcheck=1
gpgkey=https://repo.zabbix.com/zabbix/8.0/unstable/sles/15/$basearch/debuginfo/repodata/repomd.xml.key
enabled=0
update=1
```

Una vez habilitado, instale los paquetes:

- En RHEL, instale un único paquete con información de depuración para todos los componentes de Zabbix:

```
dnf install zabbix-debuginfo
```

- En SUSE, instale paquetes debuginfo específicos por componente:

```
zypper install zabbix-<component>-debuginfo
```

Habilitar los repositorios de versiones inestables

Las instrucciones siguientes son para habilitar los repositorios de versiones inestables de Zabbix (deshabilitados de forma predeterminada), que se usan para las versiones candidatas de las versiones menores de Zabbix.

Primero, instale o actualice al paquete zabbix-release más reciente. Para habilitar los paquetes rc en su sistema, haga lo siguiente:

Red Hat Enterprise Linux

Abra el archivo `/etc/yum.repos.d/zabbix.repo` y establezca `enabled=1` para el repositorio `zabbix-unstable`.

```
[zabbix-unstable]
name=Zabbix Official Repository (unstable) - $basearch
baseurl=https://repo.zabbix.com/zabbix/8.0/unstable/rhel/8/$basearch/
enabled=1
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-ZABBIX-A14FE591
```

Debian/Ubuntu

Abra el archivo `/etc/apt/sources.list.d/zabbix.list` y descomente "Zabbix unstable repository".

```
#### Zabbix unstable repository
deb https://repo.zabbix.com/zabbix/8.0/unstable/debian bullseye main
deb-src https://repo.zabbix.com/zabbix/8.0/unstable/debian bullseye main
```

SUSE

Abra el archivo `/etc/zypp/repos.d/zabbix.repo` y establezca `enable=1` para el repositorio `zabbix-unstable`.

```
[zabbix-unstable]
name=Zabbix Official Repository
type=rpm-md
baseurl=https://repo.zabbix.com/zabbix/8.0/unstable/sles/15/x86_64/
gpgcheck=1
gpgkey=https://repo.zabbix.com/zabbix/8.0/unstable/sles/15/x86_64/repodata/repomd.xml.key
enabled=1
update=1
```

Instalación del agent de Windows desde MSI

Descripción general

El agent de Zabbix se puede instalar en Windows utilizando paquetes de instalación MSI de 32 o 64 bits, disponibles para descarga.

Los requisitos mínimos del sistema operativo para la instalación de MSI son:

- **Para el agent de Zabbix:** Windows XP (64 bits) o Windows Server 2003
- **Para el agent 2 de Zabbix:** Windows 10 (32 bits) o Windows Server 2016

Los paquetes de 32 bits no se pueden instalar en sistemas de 64 bits.

Los paquetes incluyen:

- Soporte TLS (la configuración de TLS es opcional)
- Utilidades Zabbix get y Zabbix sender (pueden instalarse junto con el agent/agent 2 de Zabbix o por separado)

Attention:
Los paquetes del agent 2 de Zabbix no incluyen plugins cargables (MongoDB, PostgreSQL, MSSQL), que deben descargarse e instalarse por separado.

La instalación puede realizarse utilizando el Asistente de configuración o la línea de comandos.

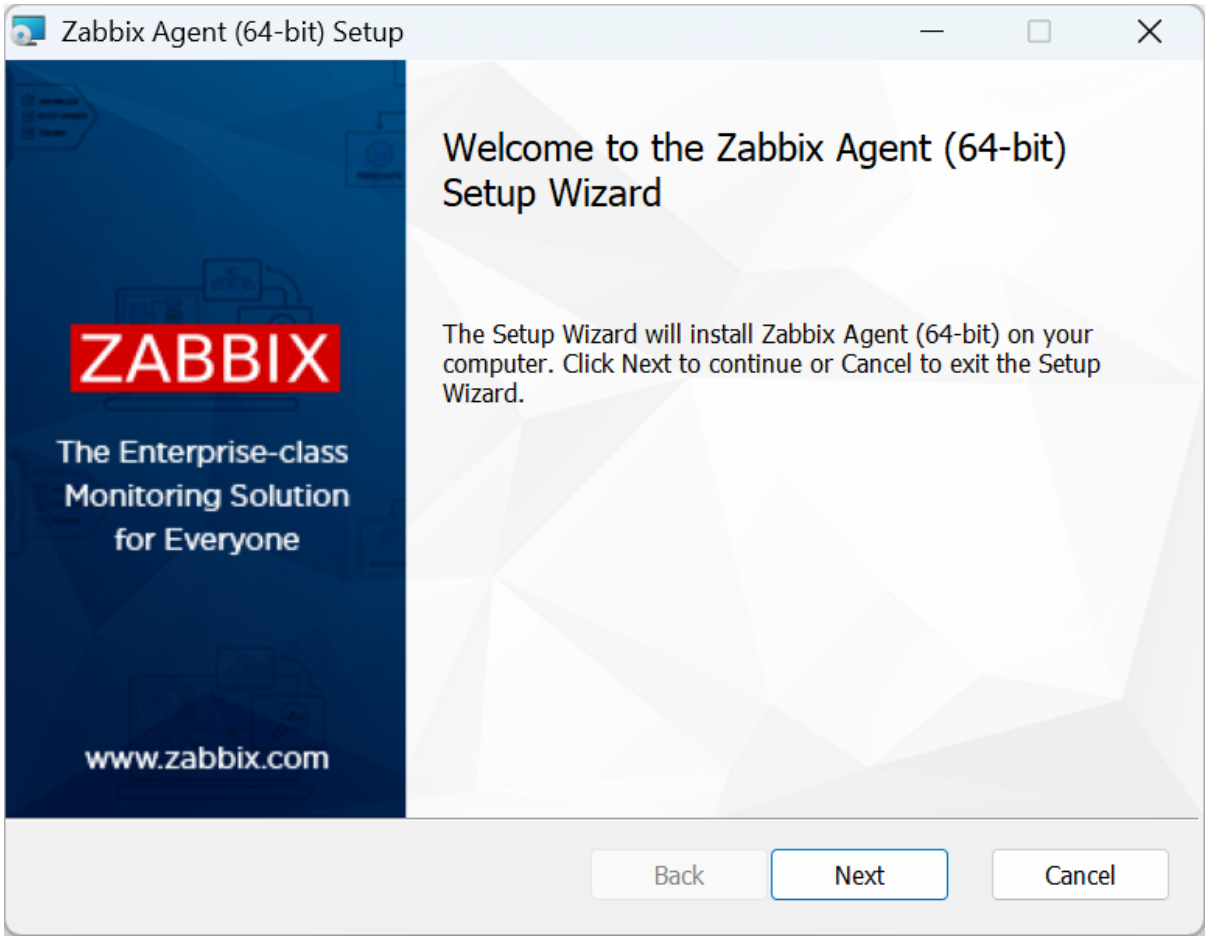
Aunque la instalación mediante paquetes MSI es totalmente compatible, se recomienda instalar al menos Microsoft .NET Framework 2 para un manejo adecuado de los errores.

Attention:
Se recomienda utilizar la ubicación de instalación predeterminada proporcionada por el instalador. El uso de una ubicación personalizada sin los permisos necesarios puede comprometer la seguridad de la instalación.

Instalación mediante el asistente de configuración

Los siguientes pasos de instalación se aplican tanto a Zabbix agent como a Zabbix agent 2.

1. Haga doble clic en el archivo MSI descargado para iniciar la instalación:

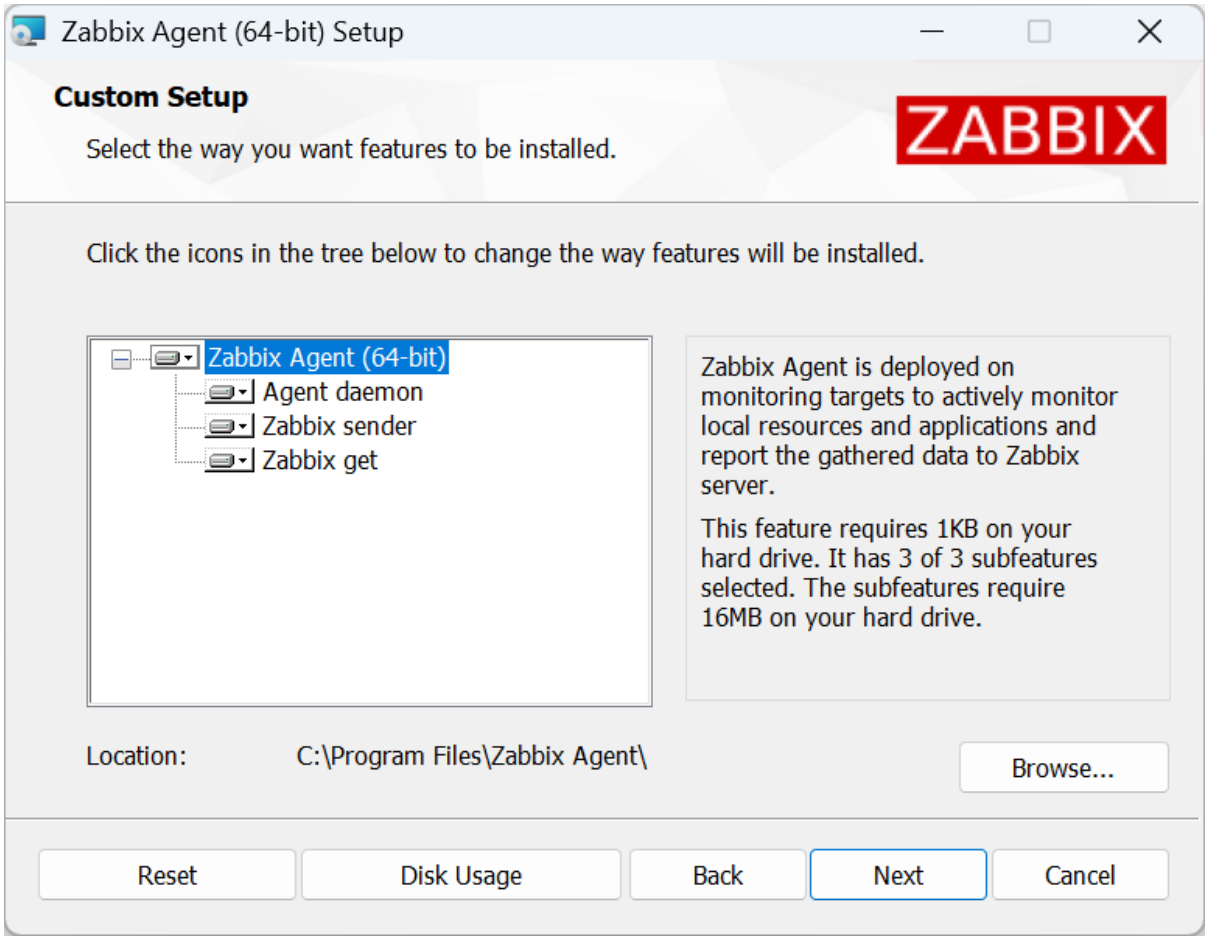


2. Acepte el Acuerdo de licencia de usuario final:



3. Seleccione los componentes de Zabbix (demonio del agent, Zabbix sender, Zabbix get) que desea instalar:

Attention:
Se recomienda utilizar la ubicación de instalación predeterminada proporcionada por el instalador. El uso de una ubicación personalizada sin los permisos necesarios puede poner en riesgo la seguridad de la instalación.

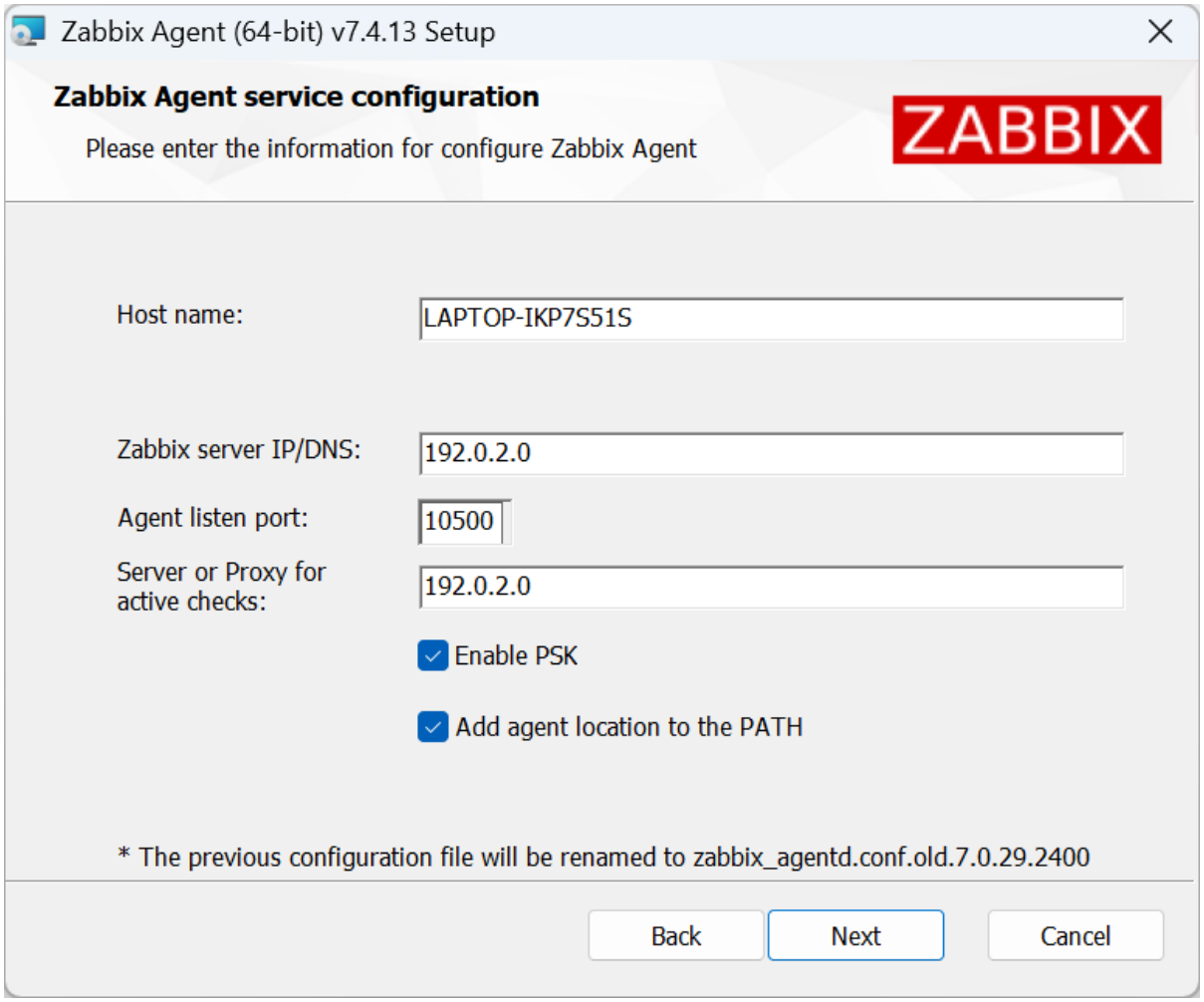


Note:
Si se utiliza una ubicación personalizada fuera del directorio `Program Files` y la carpeta de destino ya existe, el instalador no modificará sus permisos. Durante una instalación nueva, se le solicitará que establezca manualmente los permisos adecuados o que continúe sin modificarlos.

4. Configure los siguientes parámetros. Sus valores se establecerán en el archivo de configuración del agent de Zabbix:

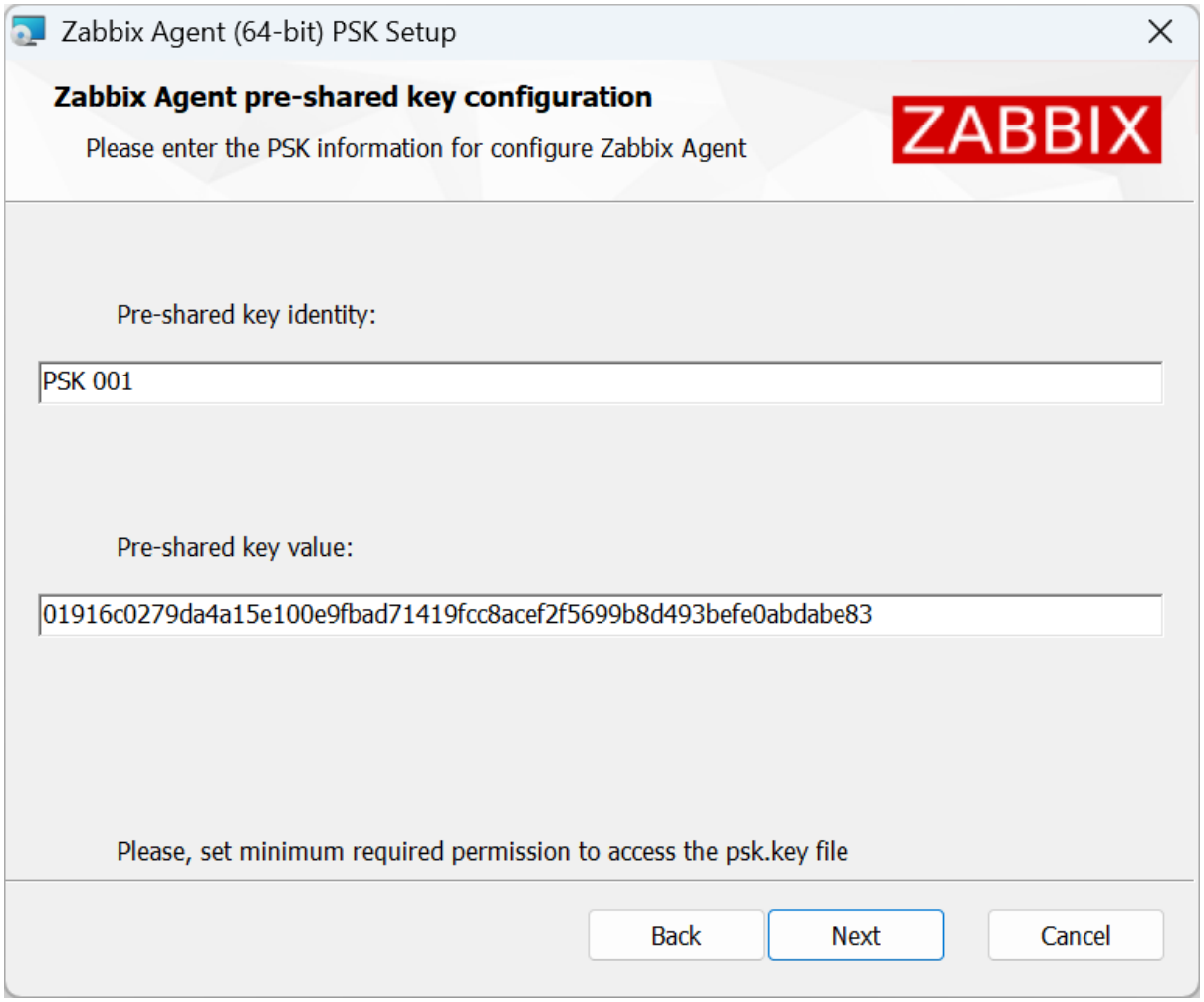
Parámetro	Descripción
Nombre del host	El nombre del host de la máquina donde se está instalando Zabbix agent. Establece el parámetro Hostname.
IP/DNS del server de Zabbix	Una lista de direcciones IP separadas por comas, opcionalmente en notación CIDR, o nombres DNS de servers o proxies de Zabbix. Este parámetro es **obligatorio**. Establece el parámetro Server.
Puerto de escucha del agent	El agent escuchará en este puerto las conexiones del server. Establece el parámetro ListenPort.
Server o proxy para las comprobaciones activas	La dirección del server/proxy de Zabbix o la configuración del clúster desde la que se obtendrán las comprobaciones activas. La dirección del server/proxy es una dirección IP o un nombre DNS y, opcionalmente, un puerto separado por dos puntos. Establece el parámetro ServerActive.
Habilitar PSK	Marque la casilla para habilitar la compatibilidad con TLS mediante claves precompartidas. Establece los parámetros TLSConnect y TLSAccept en `psk`.
Añadir la ubicación del agent al PATH	Marque la casilla para añadir la ubicación de Zabbix agent a la variable PATH del sistema.

Note:
Si se detecta un Zabbix agent existente, se mostrarán los valores de los parámetros de su archivo de configuración. Además, el archivo de configuración existente cambiará de nombre durante la instalación y se creará un nuevo archivo de configuración.



5. Configure los parámetros de PSK si marcó la casilla *Habilitar PSK* en el paso anterior. Estos parámetros también se establecerán en el archivo de configuración de Zabbix agent:

Parámetro	Descripción
Identidad de la clave precompartida	La cadena de identidad de la clave precompartida. Establece el parámetro TLSPSKIdentity.
Valor de la clave precompartida	El valor de la cadena de la clave precompartida. Crea el archivo `psk.key` que contiene la clave y establece el parámetro TLSPSKFile en la ubicación de la clave (predeterminada: `C:\Program Files\Zabbix Agent\psk.key`). Se **recomienda** restringir el acceso al archivo de la clave precompartida ajustando su configuración de seguridad para que solo Zabbix agent (o el usuario que ejecuta el agent) pueda leerlo.

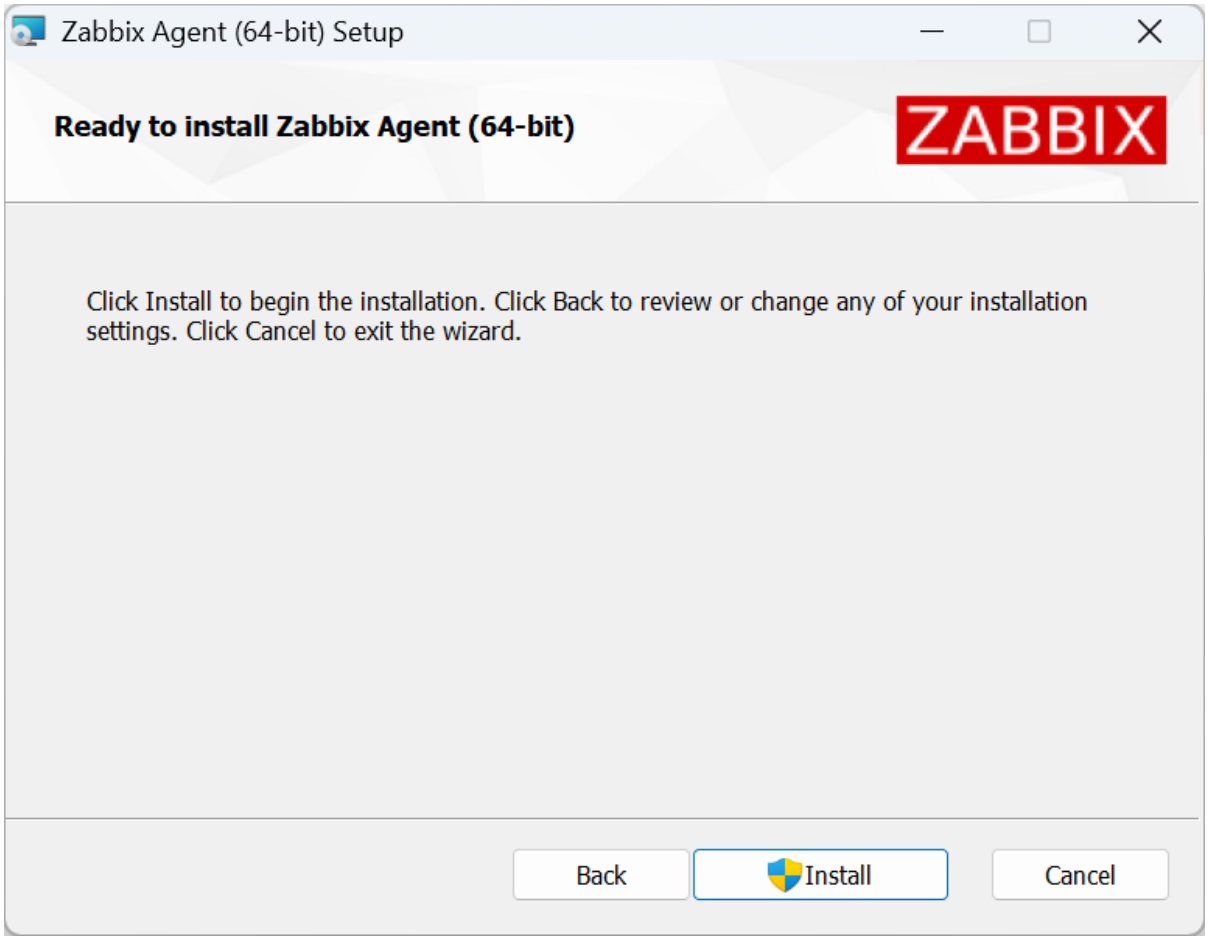


6. Haga clic en *Instalar* para comenzar la instalación.

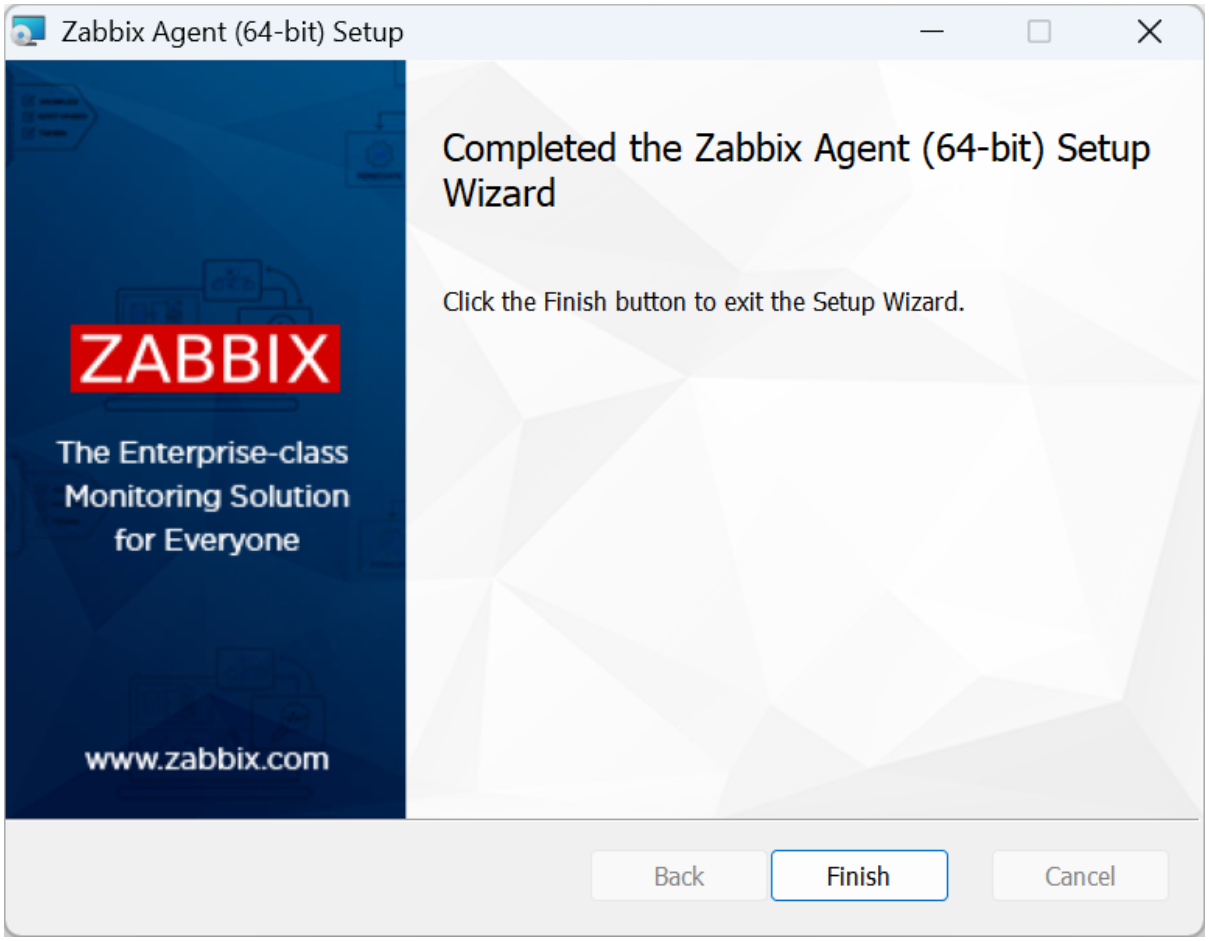
Todos los componentes de Zabbix seleccionados y el archivo de configuración de Zabbix agent se instalarán en la ubicación especificada (predeterminada: `C:\Program Files\Zabbix Agent`). Lo mismo se aplica a Zabbix agent 2, excepto que los archivos de configuración adicionales de sus plugins integrados se instalarán en la subcarpeta `zabbix_agent2.d\plugins.d`.

Además, `zabbix_agentd.exe` (o `zabbix_agent2.exe`) se configurará como un servicio de Windows con inicio automático retrasado (o inicio automático en versiones de Windows anteriores a Windows Vista/Server 2008).

Si durante la instalación se está ejecutando una versión diferente de Zabbix agent, se le solicitará que elija entre cerrar la aplicación e intentar reiniciarla, o dejarla abierta, en cuyo caso será necesario reiniciar el sistema.



7. Haga clic en el botón *Finalizar* para salir del asistente de configuración.



Instalación desde la línea de comandos

Zabbix agent se puede instalar desde la línea de comandos ejecutando el instalador MSI con `msiexec`. Por ejemplo:

```
msiexec.exe /l*v "C:\package.log" /i "C:\zabbix_agent-8.0.0-windows-amd64-openssl.msi" /qn+ SERVER=192.0.2
```

Este método permite instalaciones desatendidas y configuraciones personalizadas mediante parámetros.

Parámetros compatibles

Los paquetes instaladores MSI de Zabbix agent admiten los siguientes parámetros tanto para Zabbix agent como para Zabbix agent 2.

Note:
Los parámetros de Zabbix agent/agent2 se establecen en el archivo de configuración durante la instalación. Haga clic en el nombre de un parámetro para ver su descripción detallada y ejemplos de configuración en la página Zabbix agent (Windows). Para Zabbix agent 2, consulte la página Zabbix agent 2 (Windows).

Parameter	Description
ADDDEFAULT	Una lista de componentes separados por comas para instalar con su configuración predeterminada. Para obtener más información, consulte ADDDEFAULT property. Valores posibles: `AgentProgram`, `GetProgram`, `SenderProgram`, `ALL` Ejemplo: `ADDDEFAULT=AgentProgram,GetProgram`
ADDLOCAL	Una lista de componentes separados por comas para instalar localmente. Para obtener más información, consulte ADDLOCAL property. Valores posibles: `AgentProgram`, `GetProgram`, `SenderProgram`, `ALL` Ejemplo: `ADDLOCAL=AgentProgram,SenderProgram`
ALLOWDENYKEY	Una lista de parámetros AllowKey o DenyKey separados por punto y coma para restringir las comprobaciones de Zabbix agent. Si es necesario, use una barra invertida para escapar el delimitador (`\;`). Establece los parámetros AllowKey y DenyKey en el archivo de configuración del agent. Ejemplo: `ALLOWDENYKEY="AllowKey=system.run[more C:\Windows\System32\drivers\etc\hosts\\; echo 'File read complete'];DenyKey=system.run[*]"`
CONF	La ruta completa a un archivo de configuración de template para Zabbix agent. Durante la instalación, este archivo se convertirá en el archivo de configuración del agent. El archivo debe contener al menos los parámetros Server y LogFile. Ejemplo: `CONF="C:\full\path\to\example.conf"`
DONOTSTART	Use `DONOTSTART=1` para impedir que el instalador MSI inicie el servicio de Zabbix agent.
ENABLEPATH	Use `ENABLEPATH=1` para agregar la ubicación de Zabbix agent a la variable PATH del sistema.
ENABLEPERSISTENTBUFFER	Solo Zabbix agent 2. Habilita el uso de almacenamiento persistente local para los items activos.
HOSTINTERFACE	Parámetro opcional que define la interfaz del host.
HOSTMETADATA	Parámetro opcional que define los metadatos del host.
HOSTMETADATAITEM	Parámetro opcional que define un item usado para obtener los metadatos del host.
HOSTNAME	Parámetro opcional que define el nombre del host.
HOSTNAMEITEM	Parámetro opcional que define un item usado para obtener el nombre del host.
INCLUDE	Una lista de archivos individuales o de todos los archivos de un directorio, separados por punto y coma, para incluir en el archivo de configuración de Zabbix agent.
IGNOREALREADYEXISTSFOLDER	Para instalaciones nuevas, use `IGNOREALREADYEXISTSFOLDER=1` para permitir la instalación en una carpeta de destino existente fuera del directorio `Program Files`. Si no se especifica este parámetro, la instalación falla cuando ya existe una carpeta de destino de este tipo.
INSTALLFOLDER	La ruta completa a una carpeta donde se instalarán los componentes de Zabbix y el archivo de configuración de Zabbix agent. Para Zabbix agent 2, los archivos de configuración adicionales para los plugins integrados se instalarán en la subcarpeta `zabbix_agent2.d\plugins.d`. Ejemplo: `INSTALLFOLDER="C:\Program Files\Zabbix Agent"`
LISTENIP	Una lista de direcciones IP separadas por comas en las que el agent debe escuchar.
LISTENPORT	El agent escuchará en este puerto las conexiones desde el server.
LOGFILE	El nombre del archivo de registro de Zabbix agent.
LOGTYPE	El tipo de salida del registro.
NONMSICONFNAME	La ruta completa a un archivo de configuración personalizado para Zabbix agent. Durante la instalación, cualquier parámetro válido de configuración del agent presente en este archivo (limitado a los enumerados en esta tabla) se escribirá en el nuevo archivo de configuración del agent creado. El archivo debe contener al menos el parámetro Server. Ejemplo: `NONMSICONFNAME="C:\full\path\to\example.conf"`
PERSISTENTBUFFERFILE	Solo Zabbix agent 2. El archivo donde Zabbix agent 2 debe conservar la base de datos SQLite.

Parameter	Description
PERSISTENTBUFFERPERIOD	Solo Zabbix agent 2. El período de tiempo durante el cual deben almacenarse los datos cuando no hay conexión con el server o proxy.
SERVER	Una lista de direcciones IP separadas por comas, opcionalmente en notación CIDR, o nombres DNS de Zabbix servers o Zabbix proxies. Este parámetro es **obligatorio**, excepto cuando STARTAGENTS está establecido en 0.
SERVERACTIVE	La dirección del Zabbix server/proxy o la configuración de clúster desde la que obtener comprobaciones activas.
SKIP	Use `SKIP=fw` para impedir que el instalador MSI agregue una regla de excepción en el Firewall de Windows para Zabbix agent.
SOURCEIP	La dirección IP de origen para las conexiones salientes a Zabbix server o Zabbix proxy, o para realizar conexiones al ejecutar algunos items (web.page.get, net.tcp.port, etc.).
STARTAGENTS	El número de instancias pre-fork de zabbix_agentd que procesan comprobaciones pasivas. Si se establece en 0, las comprobaciones pasivas se deshabilitan y el agent no escuchará en ningún puerto TCP.
STARTUPTYPE	Tipo de inicio del servicio de Zabbix agent. Valores posibles: **automatic** - inicia el servicio automáticamente al arrancar Windows; **delayed** - *(predeterminado)* retrasa el inicio del servicio hasta que los servicios iniciados automáticamente hayan completado su arranque (disponible en Windows Vista/Server 2008 y versiones posteriores); **manual** - inicia el servicio manualmente (por un usuario o una aplicación); **disabled** - deshabilita el servicio para que no pueda ser iniciado por un usuario o una aplicación. Ejemplo: `STARTUPTYPE=disabled`
STATUSPORT	Solo Zabbix agent 2. Si se establece, el agent escuchará en este puerto las solicitudes HTTP de estado (http://localhost:<port>/status).
TIMEOUT	Especifica cuánto tiempo esperar (en segundos) para establecer la conexión e intercambiar datos con Zabbix proxy o server.
TLSACCEPT	Las conexiones entrantes que se aceptarán (se usa para comprobaciones pasivas). Si se establece en `psk`, entonces TLSCONNECT también se establecerá en `psk` (a menos que se especifique lo contrario).
TLSCAFILE	La ruta completa de un archivo que contiene los certificados de la CA de nivel superior para la verificación del certificado del par.
TLSCERTFILE	La ruta completa de un archivo que contiene el certificado del agent o la cadena de certificados.
TLSCONNECT	Cómo debe conectarse el agent a Zabbix server o proxy (se usa para comprobaciones activas). Si se establece en `psk`, entonces TLSACCEPT también se establecerá en `psk` (a menos que se especifique lo contrario).
TLSCRLFILE	La ruta completa de un archivo que contiene certificados revocados.
TLSKEYFILE	La ruta completa de un archivo que contiene la clave privada de Zabbix agent.
TLSPSKFILE	La ruta completa de un archivo que contiene la clave precompartida de Zabbix agent. Si se establecen tanto TLSPSKFILE como TLSPSKVALUE, el valor de TLSPSKVALUE se escribirá en el archivo especificado en TLSPSKFILE. Se **recomienda** restringir el acceso al archivo de clave precompartida ajustando la configuración de seguridad del archivo para que solo Zabbix agent (o el usuario que ejecuta el agent) pueda leerlo.
TLSPSKIDENTITY	La cadena de identidad de la clave precompartida.
TLSPSKVALUE	El valor de la cadena de la clave precompartida. Si se establecen tanto TLSPSKFILE como TLSPSKVALUE, el valor de TLSPSKVALUE se escribirá en el archivo especificado en TLSPSKFILE. Ejemplo: `TLSPSKVALUE=1f87b595725ac58dd977beef14b97461a7c1045b9a1c963065002c5473194952`
TLSSERVERCERTISSUER	El emisor del certificado del server (proxy) permitido.
TLSSERVERCERTSUBJECT	El sujeto del certificado del server (proxy) permitido.
UNSAFEUSERPARAMETERS	Permite que todos los caracteres se pasen en los argumentos de los parámetros definidos por el usuario.

Ejemplos

El siguiente ejemplo instala Zabbix agent con una configuración personalizada. También habilita la compatibilidad con TLS mediante claves precompartidas.

```
mkdir "C:\Program Files\Zabbix Agent" 2>nul
msiexec.exe /l*v "C:\package.log" /i "C:\zabbix_agent-8.0.0-windows-amd64-openssl.msi" /qn+^
 SERVER=192.0.2.1^
 INSTALLFOLDER="C:\Program Files\Zabbix Agent"^
 HOSTNAME=LAPTOP-IKP7S51S^
```

```
 TLSACCEPT=psk^
 TLSCONNECT=psk^
 TLSPSKIDENTITY="PSK 001"^
 TLSPSKFILE="C:\Program Files\Zabbix Agent\psk.key"^
 TLSPSKVALUE=1f87b595725ac58dd977beef14b97461a7c1045b9a1c963065002c5473194952^
 ENABLEPATH=1^
 ALLOWDENYKEY="AllowKey=system.run[type C:\Windows\System32\drivers\etc\hosts];DenyKey=system.run[*]"
```

El siguiente ejemplo instala una versión más reciente de Zabbix agent y usa un **archivo de configuración de template** (`CONF="C:\agent-template.conf"`). Durante la instalación, este archivo se convertirá en el archivo de configuración del agent. Para heredar parámetros del archivo de configuración anterior, use marcadores de posición de parámetros (por ejemplo, `[AllowDenyKey]`).

```
msiexec.exe /l*v "C:\package.log" /i "C:\zabbix_agent-8.0.1-windows-amd64-openssl.msi" /qn+ NONMSICONFNAME

#### ejemplo de agent-template.conf:
LogFile=[LogFile]
[AllowDenyKey]
Server=192.0.2.8
Hostname=DESKTOP-X9F4A2B
[Include]
[TLSConnect]
[TLSAccept]
[TLSPSKIdentity]
[TLSPSKFile]
```

Como alternativa, puede usar un **archivo de configuración personalizado** (`NONMSICONFNAME="C:\agent-custom.conf"`). Durante la instalación, cualquier parámetro de configuración válido de agent presente en este archivo (limitado a los enumerados en la tabla anterior) se escribirá en el nuevo archivo de configuración del agent creado. Para conservar la configuración existente del agent, defina los parámetros que deben preservarse.

```
msiexec.exe /l*v "C:\package.log" /i "C:\zabbix_agent-8.0.1-windows-amd64-openssl.msi" /qn+ NONMSICONFNAME

#### ejemplo de agent-custom.conf:
Server=192.0.2.8
Hostname=DESKTOP-X9F4A2B
```

Plugins cargables del agent 2 de Zabbix

Los plugins cargables del agent 2 de Zabbix pueden instalarse en Windows utilizando paquetes instaladores MSI de 64 bits, disponibles para descarga.

Los requisitos mínimos del sistema operativo para la instalación MSI son Windows 10 (64 bits) o Windows Server 2016.

> **Attention:**
> Antes de instalar un plugin, por favor revise su archivo README. Puede contener requisitos específicos e instrucciones de instalación.

De manera similar al agent/agent2 de Zabbix, los plugins cargables pueden instalarse utilizando el Asistente de Configuración o la línea de comandos.

Instalación desde el Asistente de Configuración

1. Haga doble clic en el archivo MSI descargado para iniciar la instalación.
2. Acepte el Acuerdo de Licencia de Usuario Final.
3. Seleccione los plugins cargables de Zabbix agent 2 que se instalarán.

> **Attention:**
> Se recomienda utilizar la ubicación de instalación predeterminada proporcionada por el instalador. El uso de una ubicación personalizada sin los permisos necesarios puede comprometer la seguridad de la instalación.

4. Haga clic en *Instalar* para comenzar la instalación.

Todos los plugins cargables de Zabbix agent 2 seleccionados se instalarán en la ubicación especificada (por defecto: `C:\Program Files\Zabbix Agent 2`), con sus archivos de configuración instalados en la subcarpeta `zabbix_agent2.d`.

5. Haga clic en el botón *Finalizar* para salir del Asistente de Configuración.

Instalación desde la línea de comandos

Los plugins cargables de Zabbix agent 2 se pueden instalar desde la línea de comandos ejecutando el instalador MSI con `msiexec`. Por ejemplo:

```
msiexec.exe /l*v "C:\package.log" /i "C:\zabbix_agent2_plugins-8.0.0-windows-amd64.msi" /qn+
```

Los paquetes del instalador MSI de plugins cargables de Zabbix agent 2 admiten los siguientes parámetros.

Parameter	Description
ADDDEFAULT	Una lista de componentes separados por comas para instalar con su configuración predeterminada. Para más información, consulte ADDDEFAULT property. Valores posibles: `ALL`, `CephPlugin`, `EmberplusPlugin`, `MongodbPlugin`, `MssqlPlugin`, `NvidiagpuPlugin`, `PostgresqlPlugin` Ejemplo: `ADDDEFAULT=MongodbPlugin,PostgresqlPlugin`
ADDLOCAL	Una lista de componentes separados por comas para instalar localmente. Para más información, consulte ADDLOCAL property. Valores posibles: `ALL`, `CephPlugin`, `EmberplusPlugin`, `MongodbPlugin`, `MssqlPlugin`, `NvidiagpuPlugin`, `PostgresqlPlugin` Ejemplo: `ADDLOCAL=MongodbPlugin,MssqlPlugin`
INSTALLFOLDER	La ruta completa a una carpeta donde se instalarán los componentes de Zabbix, con sus archivos de configuración instalados en la subcarpeta `zabbix_agent2.d`. Ejemplo: `INSTALLFOLDER="C:\Program Files\Zabbix Agent 2"`

Compilar Zabbix agent 2 en Windows

Descripción general

Esta página muestra cómo compilar el Zabbix agent 2 a partir del código fuente en Windows 10 (64 bits o 32 bits).

Las versiones de 32 bits y 64 bits pueden compilarse en una plataforma de 64 bits, pero en una plataforma de 32 bits solo puede compilarse la versión de 32 bits.

Para compilar el Zabbix agent 2 se requiere:

- Herramientas de compilación MinGW
- Lenguaje de programación Go
- OpenSSL (para las funciones de cifrado de Zabbix)
- PCRE2 (expresiones regulares compatibles con Perl; para las funciones de coincidencia de patrones mediante expresiones regulares de Zabbix)

Puede compilar el Zabbix agent 2 mediante uno de los siguientes métodos:

- Uso de vcpkg: un enfoque automatizado que simplifica la gestión de dependencias mediante un gestor de paquetes de C++.
- Compilación manual: un enfoque manual que requiere instalar todas las dependencias antes de compilar el agent.

Attention:
Antes de iniciar el proceso de compilación, tenga en cuenta lo siguiente:

- Para ejecutar comandos, utilice el símbolo del sistema, iniciado por un usuario con permisos suficientes para escribir en carpetas protegidas. Sin embargo, al instalar OpenSSL y PCRE2, utilice la terminal de MSYS2.
- Se recomienda crear un directorio de trabajo en `C:\Zabbix` para todos los archivos de código fuente y las carpetas de compilación. Sin embargo, los componentes compilados deben instalarse en `C:\Zabbix\x64` (o en `C:\Zabbix\x86` para compilaciones de 32 bits).

Compilación de Zabbix agent 2 con vcpkg

Esta sección contiene instrucciones para compilar Zabbix agent con vcpkg, un administrador de paquetes que simplifica la gestión de dependencias y la integración con proyectos de C++.

1. Descargue e instale Build Tools for Visual Studio 2022. Durante la instalación, asegúrese de seleccionar la carga de trabajo *Desktop development with C++*, que incluye el administrador de paquetes vcpkg.

2. Descargue e instale Go (disponible como instalador MSI). Durante la instalación, asegúrese de especificar `C:\Zabbix\Go` como carpeta de instalación.

3. Descargue la distribución de MinGW que utilice la biblioteca de tiempo de ejecución de Microsoft Visual C; por ejemplo:

- Para compilaciones de 64 bits: `x86_64-15.1.0-release-win32-seh-msvcrt-rt_v12-rev0.7z`
- Para compilaciones de 32 bits: `i686-15.1.0-release-win32-dwarf-msvcrt-rt_v12-rev0.7z`

> **Attention:**
> La versión de MinGW debe cumplir los requisitos mínimos.

A continuación, extráigala en `C:\Zabbix\mingw64` (o `C:\Zabbix\mingw32` para compilaciones de 32 bits).

4. Inicialice vcpkg e instale las dependencias necesarias para compilar Zabbix agent 2 (tenga en cuenta que esto puede tardar algún tiempo):

```
cd C:\Zabbix

set PATH=%PATH%;"C:\Program Files (x86)\Microsoft Visual Studio\2022\BuildTools\VC\vcpkg"
vcpkg new --application
vcpkg add port pcre2
vcpkg add port libiconv
vcpkg add port openssl

#### Para compilaciones de 64 bits:
set PATH=C:\Zabbix\mingw64\bin;%PATH%
vcpkg install --triplet x64-mingw-static --x-install-root=x64

#### Para compilaciones de 32 bits:
set PATH=C:\Zabbix\mingw32\bin;%PATH%
vcpkg install --triplet x86-mingw-static --x-install-root=x86
```

5. Descargue el archivo de código fuente de Zabbix y extráigalo en `C:\Zabbix\zabbix-8.0.0`.

6. Acceda al directorio de compilación de Zabbix (`C:\Zabbix\zabbix-8.0.0\build\mingw`) y cree el siguiente script `build.bat`:

- Para compilaciones de 64 bits:

```
:: Agregar MinGW y Go a la variable del sistema `PATH` para la sesión actual:
set PATH=C:\Zabbix\mingw64\bin;%PATH%
set PATH=C:\Zabbix\Go\bin;%PATH%

:: Establecer la ruta de instalación de vcpkg:
set vcpkg="C:\Zabbix\x64\x64-mingw-static"

:: Establecer las opciones del enlazador para la biblioteca Crypt32:
SET CGO_LDFLAGS="-lCrypt32"

:: Ejecutar el proceso de compilación:
mingw32-make GOFLAGS="-buildvcs=false" ARCH=AMD64 ^
    PCRE2="%vcpkg%" ^
    OPENSSL="%vcpkg%" ^
    all
```

- Para compilaciones de 32 bits:

```
:: Agregar MinGW y Go a la variable del sistema `PATH` para la sesión actual:
set PATH=C:\Zabbix\mingw32\bin;%PATH%
set PATH=C:\Zabbix\Go\bin;%PATH%

:: Establecer la ruta de instalación de vcpkg:
set vcpkg="C:\Zabbix\x86\x86-mingw-static"

:: Establecer las opciones del enlazador para la biblioteca Crypt32:
SET CGO_LDFLAGS="-lCrypt32"

:: Ejecutar el proceso de compilación:
mingw32-make GOFLAGS="-buildvcs=false" ARCH=x86 ^
    PCRE2="%vcpkg%" ^
```

```
    OPENSSL="%vcpkg%" ^
    all
```

7. Compile Zabbix agent 2 ejecutando el script:

```
build.bat
```

Después de la compilación, el binario de Zabbix agent 2 se encontrará en `C:\Zabbix\zabbix-8.0.0\bin\win64` (para compilaciones de 64 bits) o en `C:\Zabbix\zabbix-8.0.0\bin\win32` (para compilaciones de 32 bits). Los archivos de configuración de Zabbix agent 2 se encuentran en `C:\Zabbix\zabbix-8.0.0\src\go\conf`.

Para ejecutar el agent, copie `zabbix_agent2.exe` y sus archivos de configuración en una carpeta dedicada (por ejemplo, `C:\Zabbix\agent2`) y, a continuación, ejecute el agent:

```
mkdir C:\Zabbix\agent2

#### Para compilaciones de 64 bits:
copy C:\Zabbix\zabbix-8.0.0\bin\win64\zabbix_agent2.exe C:\Zabbix\agent2\

#### Para compilaciones de 32 bits:
copy C:\Zabbix\zabbix-8.0.0\bin\win32\zabbix_agent2.exe C:\Zabbix\agent2\

copy C:\Zabbix\zabbix-8.0.0\src\go\conf\zabbix_agent2.conf C:\Zabbix\agent2\
xcopy /E /I C:\Zabbix\zabbix-8.0.0\src\go\conf\zabbix_agent2.d C:\Zabbix\agent2\zabbix_agent2.d\

C:\Zabbix\agent2\zabbix_agent2.exe -c C:\Zabbix\agent2\zabbix_agent2.conf
```

Si es necesario, continúe con la compilación de plugins cargables de Zabbix agent 2.

Compilación de plugins cargables de Zabbix agent 2

1. Descargue el código fuente del plugin de Zabbix que coincida con la versión de su Zabbix agent 2 (por ejemplo, `zabbix-agent2-plugin-postgresql-8.0.0.tar.gz`) y extráigalo en `C:\Zabbix`.

> **Attention:**
> Antes de instalar un plugin, consulte su archivo README. Puede contener requisitos específicos e instrucciones de instalación.

2. Vaya al directorio del plugin extraído y compile el plugin:

```
cd C:\Zabbix\zabbix-agent2-plugin-ember-plus-8.0.0

#### Para compilaciones de 64 bits:
mingw32-make ARCH=AMD64

#### Para compilaciones de 32 bits:
mingw32-make ARCH=x86
```

Después de la compilación, el binario del plugin de Zabbix agent 2 y su archivo de configuración se encontrarán en el mismo directorio del plugin.

El ejecutable del plugin puede colocarse en cualquier ubicación, siempre que pueda ser cargado por Zabbix agent 2. Especifique la ruta al binario del plugin en el archivo de configuración del plugin, por ejemplo, en postgresql.conf para el plugin PostgreSQL:

```
Plugins.PostgreSQL.System.Path=/path/to/executable/zabbix-agent2-plugin-postgresql
```

La ruta al archivo de configuración del plugin debe especificarse en el parámetro Include del archivo de configuración de Zabbix agent 2:

```
Include=/path/to/plugin/configuration/file/postgresql.conf
```

Continúe en configuración para obtener más detalles sobre la configuración de plugins.

Compilar Zabbix agent 2 manualmente

> **Attention:**
> Este método de compilación de Zabbix agent 2 es adecuado para usuarios que requieren un control total sobre el entorno de compilación o se encuentran en un entorno restringido donde usar vcpkg no es posible.

Esta sección contiene instrucciones para compilar Zabbix agent 2 manualmente, lo que incluye instalar las herramientas de compilación y dependencias necesarias, y luego compilar el agent.

Configuración de las herramientas de compilación

1. Descargue e instale MSYS2 (disponible como instalador MSI). Durante la instalación, asegúrese de especificar `C:\Zabbix\msys64` como carpeta de instalación.

2. Descargue e instale Go (disponible como instalador MSI; consulte las versiones de Go actualmente compatibles). Durante la instalación, asegúrese de especificar `C:\Zabbix\Go` como carpeta de instalación.

3. Descargue la distribución de MinGW que utiliza la biblioteca en tiempo de ejecución Microsoft Visual C; por ejemplo:

- Para compilaciones de 64 bits: `x86_64-15.1.0-release-win32-seh-msvcrt-rt_v12-rev0.7z`
- Para compilaciones de 32 bits: `i686-15.1.0-release-win32-dwarf-msvcrt-rt_v12-rev0.7z`

Attention:
La versión de MinGW debe cumplir los requisitos mínimos.

Luego, extráigala en `C:\Zabbix\mingw64` (o `C:\Zabbix\mingw32` para compilaciones de 32 bits).

Instalación de OpenSSL

Note:
Para compilar Zabbix agent sin compatibilidad con TLS, continúe con la sección Instalación de PCRE2.

1. Abra el terminal MSYS2 MSYS con privilegios de administrador y ejecute los siguientes comandos:

```
pacman -S perl-Locale-Maketext-Simple
pacman -S nasm
pacman -S make
pacman -S cmake
```

2. Descargue el archivo de código fuente de OpenSSL y extráigalo en `C:\Zabbix\openssl-3.5.0`.

3. Vaya al directorio de OpenSSL extraído y cree el siguiente script `build.sh`:

- Para compilaciones de 64 bits:

```
####!/usr/bin/env bash

export PATH="/c/Zabbix/mingw64/bin:$PATH"
export d="/c/Zabbix/x64/OpenSSL-Win64-350-static"

perl Configure mingw64 no-shared no-capieng no-winstore no-srp no-gost no-dgram no-dtls1-method no-dtls1_2

make
make install
```

- Para compilaciones de 32 bits:

```
####!/usr/bin/env bash

export PATH="/c/Zabbix/mingw32/bin:$PATH"
export d="/c/Zabbix/x86/OpenSSL-Win64-350-static"

perl Configure mingw no-shared no-capieng no-winstore no-srp no-gost no-dgram no-dtls1-method no-dtls1_2-m

make
make install
```

Attention:
Asegúrese de revocar el acceso de escritura a los usuarios que no sean administradores en el directorio `C:\Zabbix\x64\OpenSSL-Win64-350-static`. De lo contrario, el agent cargará la configuración de SSL desde una ruta que puede ser modificada por usuarios sin privilegios, lo que podría provocar una vulnerabilidad de seguridad.

- La opción `no-shared` hace que libcrypto.lib y libssl.lib sean bibliotecas estáticas autocontenidas de OpenSSL, por lo que los binarios de Zabbix incluyen OpenSSL sin necesidad de DLL externas. Esto significa que los binarios de Zabbix se pueden

copiar a otros equipos Windows sin las bibliotecas de OpenSSL; sin embargo, cuando se publique una nueva versión de corrección de errores de OpenSSL, será necesario volver a compilar Zabbix agent.
- Sin la opción `no-shared`, Zabbix depende de las DLL de OpenSSL durante el tiempo de ejecución. Esto significa que las actualizaciones de OpenSSL pueden no requerir volver a compilar Zabbix agent; sin embargo, al copiarlo a otros equipos, también se deben copiar las DLL de OpenSSL.

Para obtener más información sobre otras opciones de configuración de OpenSSL, consulte la documentación de OpenSSL.

4. Configure e instale OpenSSL ejecutando el script (tenga en cuenta que esto puede tardar algún tiempo):

```
cd /c/Zabbix/openssl-3.5.0
./build.sh
```

Instalación de PCRE2

1. Descargue el archivo fuente de PCRE2 y extráigalo en `C:\Zabbix\pcre2-10.45`.

2. Abra la terminal MSYS2 MSYS con privilegios de administrador. Luego, cree un directorio `build` en el directorio extraído de PCRE2 y navegue hasta él:

```
mkdir /c/Zabbix/pcre2-10.45/build
cd /c/Zabbix/pcre2-10.45/build
```

3. Configure PCRE2:

```
#### Para compilaciones de 64 bits:
export PATH="/c/Zabbix/mingw64/bin:$PATH"
cmake -DCMAKE_C_COMPILER=gcc -DCMAKE_C_FLAGS="-O2 -g" -DCMAKE_INSTALL_PREFIX="/c/Zabbix/x64/PCRE2" ..

#### Para compilaciones de 32 bits:
export PATH="/c/Zabbix/mingw32/bin:$PATH"
cmake -DCMAKE_C_COMPILER=gcc -DCMAKE_C_FLAGS="-m32 -O2 -g" -DCMAKE_EXE_LINKER_FLAGS="-Wl,-mi386pe" -DCMAKE
```

Note:
Si se produce algún error, se recomienda eliminar la caché de CMake antes de intentar repetir el proceso de compilación de CMake. La caché (`CMakeCachecache.txt`) puede encontrarse en el directorio de compilación del directorio extraído de PCRE2.

4. Instale PCRE2:

```
make install
```

Compilación de Zabbix agent 2

1. Descargue el archivo comprimido del código fuente de Zabbix y extráigalo en `C:\Zabbix\zabbix-8.0.0`.

Si necesita generar un archivo comprimido del código fuente a partir del repositorio de código fuente sin procesar (por ejemplo, para aplicar parches personalizados o compilar a partir del código fuente más reciente), ejecute los siguientes comandos en una máquina **Linux** con Go instalado (necesario para configurar Zabbix agent 2):

```
git clone https://git.zabbix.com/scm/zbx/zabbix.git
cd zabbix
./bootstrap.sh
./configure
make dist
```

Esto creará un archivo comprimido del código fuente que, posteriormente, podrá copiarse a una máquina Windows.

2. Abra el símbolo del sistema con privilegios de administrador. A continuación, vaya al directorio de compilación de Zabbix y compile Zabbix agent; asegúrese de especificar correctamente los directorios donde están instalados OpenSSL y PCRE2:

- Para compilaciones de 64 bits:

```
cd C:\Zabbix\zabbix-8.0.0\build\mingw
set PATH=C:\Zabbix\mingw64\bin;%PATH%
mklink /D C:\Zabbix\x64\OpenSSL-Win64-350-static\lib C:\Zabbix\x64\OpenSSL-Win64-350-static\lib64

#### Con compatibilidad con TLS:
mingw32-make ARCH=AMD64 PCRE2="C:\Zabbix\x64\PCRE2" OPENSSL="C:\Zabbix\x64\OpenSSL-Win64-350-static"
```

```
#### Sin compatibilidad con TLS:
mingw32-make ARCH=AMD64 PCRE2="C:\Zabbix\x64\PCRE2"
```

- Para compilaciones de 32 bits:

```
cd C:\Zabbix\zabbix-8.0.0\build\mingw
set PATH=C:\Zabbix\mingw32\bin;%PATH%

#### Con compatibilidad con TLS:
mingw32-make ARCH=x86 PCRE2="C:\Zabbix\x86\PCRE2" OPENSSL="C:\Zabbix\x86\OpenSSL-Win32-350-static"

#### Sin compatibilidad con TLS:
mingw32-make ARCH=x86 PCRE2="C:\Zabbix\x86\PCRE2"
```

Después de la compilación, el binario de Zabbix agent 2 se encontrará en `C:\Zabbix\zabbix-8.0.0\bin\win64` (o en `C:\Zabbix\zabbix-8.0.0\bin\win32` para compilaciones de 32 bits). Los archivos de configuración de Zabbix agent 2 se encuentran en `C:\Zabbix\zabbix-8.0.0\src\go\conf`.

Para ejecutar el agent, copie el binario `zabbix_agent2.exe` y sus archivos de configuración a una carpeta dedicada (por ejemplo, `C:\Zabbix\agent2`) y, a continuación, ejecute el agent:

```
mkdir C:\Zabbix\agent2
copy C:\Zabbix\zabbix-8.0.0\bin\win64\zabbix_agent2.exe C:\Zabbix\agent2\
copy C:\Zabbix\zabbix-8.0.0\src\go\conf\zabbix_agent2.conf C:\Zabbix\agent2\
xcopy /E /I C:\Zabbix\zabbix-8.0.0\src\go\conf\zabbix_agent2.d C:\Zabbix\agent2\zabbix_agent2.d\

C:\Zabbix\agent2\zabbix_agent2.exe -c C:\Zabbix\agent2\zabbix_agent2.conf
```

Si es necesario, continúe con la compilación de plugins cargables de Zabbix agent 2.

Compilar el agent de Zabbix en Windows

Descripción general

Esta página muestra cómo compilar el agent de Zabbix a partir del código fuente en Windows 10 (64 bits).

Estas instrucciones se aplican a las versiones de Windows compatibles con Visual Studio 2022.

Para compilar el agent de Zabbix se requiere:

- Compilador de C (incluido en Build Tools for Visual Studio 2022)
- OpenSSL (para las funciones de cifrado de Zabbix)
- PCRE2 (expresiones regulares compatibles con Perl; para las funciones de coincidencia de patrones mediante expresiones regulares de Zabbix)

Puede compilar el agent de Zabbix mediante uno de los siguientes métodos:

- Uso de vcpkg: un enfoque automatizado que simplifica la gestión de dependencias mediante un gestor de paquetes de C++.
- Compilación manual: un enfoque manual que requiere instalar todas las dependencias antes de compilar el agent.

Según sus necesidades de monitorización, pueden ser necesarias bibliotecas adicionales. Para obtener más información, consulte Requisitos.

Attention:
Antes de iniciar el proceso de compilación, tenga en cuenta lo siguiente:

- Para ejecutar comandos, utilice "x64 Native Tools Command Prompt" (incluido en Build Tools for Visual Studio 2022), iniciado por un usuario con permisos suficientes para escribir en carpetas protegidas.
- Se recomienda crear un directorio de trabajo en `C:\Zabbix` para todos los archivos de código fuente y las carpetas de compilación. Sin embargo, los componentes compilados deben instalarse en `C:\Program Files\Zabbix\x64`.

Compilación del Zabbix agent con vcpkg

Esta sección contiene instrucciones para compilar el Zabbix agent con vcpkg, un administrador de paquetes que simplifica la gestión de dependencias y la integración con proyectos de C++.

1. Descargue e instale Build Tools for Visual Studio 2022. Durante la instalación, asegúrese de seleccionar la carga de trabajo *Desktop development with C++*, que incluye las herramientas necesarias para compilar el agent con vcpkg:

- Compilador de C (Microsoft Visual C++)
- Herramienta de línea de comandos NMake
- Administrador de paquetes vcpkg
- "x64 Native Tools Command Prompt"

2. Abra `cmd.exe` con privilegios de administrador, inicialice vcpkg e instale las dependencias necesarias para compilar Zabbix agent (tenga en cuenta que esto puede tardar un tiempo):

```
cd C:\Zabbix
vcpkg new --application
vcpkg add port pcre2
vcpkg add port openssl
vcpkg install --triplet x64-windows-static --x-install-root="C:\Program Files\Zabbix\x64"
```

3. Descargue el archivo de código fuente de Zabbix y extráigalo en `C:\Zabbix\zabbix-8.0.0`.

4. Acceda al directorio de compilación de Zabbix (`C:\Zabbix\zabbix-8.0.0\build\win32\project`) y cree el siguiente script `build.bat`; asegúrese de especificar correctamente los directorios donde están instalados OpenSSL y PCRE2:

```
:: Set vcpkg installation path:
set vcpkg=C:\Program Files\Zabbix\x64\x64-windows-static

:: Run the build process:
nmake -f Makefile CPU=AMD64 ^
    PCRE2INCDIR="%vcpkg%\include" ^
    PCRE2LIBDIR="%vcpkg%\lib" ^
    TLS=openssl ^
    TLSINCDIR="%vcpkg%\include" ^
    TLSLIBDIR="%vcpkg%\lib" ^
    LIBS="$(LIBS) Crypt32.lib" ^
    all
```

5. Abra "x64 Native Tools Command Prompt" y compile Zabbix agent ejecutando el script:

```
cd C:\Zabbix\zabbix-8.0.0\build\win32\project
build.bat
```

Si ha abierto otro símbolo del sistema, inicialice primero el entorno de compilación de Visual Studio:

```
"%ProgramFiles(x86)%\Microsoft Visual Studio\2022\BuildTools\VC\Auxiliary\Build\vcvarsall.bat" x64
cd C:\Zabbix\zabbix-8.0.0\build\win32\project
build.bat
```

Encierre entre comillas las rutas que contengan espacios.

Después de la compilación, los binarios de los componentes de Zabbix se encontrarán en `C:\Zabbix\zabbix-8.0.0\bin\win64`. El archivo de configuración de Zabbix agent se encuentra en `C:\Zabbix\zabbix-8.0.0\conf`.

Para ejecutar el agent, copie `zabbix_agentd.exe` y su archivo de configuración en una carpeta específica (por ejemplo, `C:\Zabbix\agent`) y, a continuación, ejecute el agent:

```
mkdir C:\Zabbix\agent
copy C:\Zabbix\zabbix-8.0.0\bin\win64\zabbix_agentd.exe C:\Zabbix\agent\
copy C:\Zabbix\zabbix-8.0.0\conf\zabbix_agentd.conf C:\Zabbix\agent\

C:\Zabbix\agent\zabbix_agentd.exe -c C:\Zabbix\agent\zabbix_agentd.conf
```

Compilación manual del agente Zabbix

> **Attention:**
> Este método de compilación del agente Zabbix es adecuado para usuarios que requieren un control total sobre el entorno de compilación o se encuentran en un entorno restringido donde usar vcpkg no es posible.

Esta sección contiene instrucciones para compilar manualmente el agente Zabbix, lo que incluye la instalación de las herramientas de compilación y dependencias necesarias (Perl, OpenSSL, PCRE2), y luego la compilación del agente.

Instalación de herramientas de compilación

1. Descargue e instale Build Tools for Visual Studio 2022. Durante la instalación, asegúrese de seleccionar la carga de trabajo *Desarrollo de escritorio con C++*, que incluye las herramientas necesarias para compilar manualmente el agent:

- Compilador de C (Microsoft Visual C++)
- Herramienta de línea de comandos NMake
- "x64 Native Tools Command Prompt"

Instalación de OpenSSL

> **Note:**
> Para compilar el agent de Zabbix sin compatibilidad con TLS, continúe con la sección Instalación de PCRE2.

1. Descargue e instale Strawberry Perl (disponible como instalador MSI). Durante la instalación, asegúrese de especificar `C:\Zabbix\Strawberry` como carpeta de instalación.

2. Instale el módulo de Perl `Text::Template`:

```
cpanm Text::Template
```

3. Verifique que Netwide Assembler (NASM; necesario para compilar OpenSSL) se haya compilado durante la instalación de Strawberry Perl:

```
nasm -v
#### NASM version 2.16.01 compiled on May  3 2024
```

Si NASM no está compilado, instálelo manualmente. Para obtener más información, consulte la documentación de NASM.

4. Descargue el archivo de código fuente de OpenSSL y extráigalo en `C:\Zabbix\openssl-3.5.0`.

5. Acceda al directorio extraído y configure OpenSSL, por ejemplo:

```
cd C:\Zabbix\openssl-3.5.0
perl Configure VC-WIN64A no-shared no-capieng no-winstore no-srp no-gost no-dgram no-dtls1-method no-dtls1
```

> **Attention:**
> Si elige un directorio personalizado para OpenSSL al compilar el agent de Zabbix en Windows (por ejemplo, `C:\zabbix` o `C:\openssl-64bit`), asegúrese de revocar el acceso de escritura a los usuarios que no sean administradores para este directorio. De lo contrario, el agent cargará la configuración de SSL desde una ruta que puede ser modificada por usuarios sin privilegios, lo que puede provocar una vulnerabilidad de seguridad.

- La opción `no-shared` hace que libcrypto.lib y libssl.lib sean bibliotecas estáticas de OpenSSL independientes, de modo que los binarios de Zabbix incluyan OpenSSL sin necesidad de DLL externas. Esto significa que los binarios de Zabbix se pueden copiar a otros equipos Windows sin las bibliotecas de OpenSSL; sin embargo, cuando se publique una nueva versión de corrección de errores de OpenSSL, será necesario volver a compilar el agent de Zabbix.
- Sin la opción `no-shared`, Zabbix depende de las DLL de OpenSSL durante el tiempo de ejecución. Esto significa que las actualizaciones de OpenSSL pueden no requerir volver a compilar el agent de Zabbix; sin embargo, al copiarlo a otros equipos, también se deben copiar las DLL de OpenSSL.

Para obtener más información sobre otras opciones de configuración de OpenSSL, consulte la documentación de OpenSSL.

6. Compile OpenSSL y ejecute las pruebas (tenga en cuenta que esto puede tardar algún tiempo):

> **Attention:**
> Ejecute las pruebas sin privilegios administrativos; de lo contrario, podrían producirse resultados inesperados o riesgos de seguridad. Si alguna prueba falla, consulte la documentación de OpenSSL para obtener ayuda con la resolución de problemas.

```
nmake
nmake test
...
All tests successful.
Files=325, Tests=3101, 822 wallclock secs ( 4.81 usr +  0.81 sys =  5.62 CPU)
Result: PASS
```

7. Instale OpenSSL:

```
nmake install
```

Para instalar únicamente los componentes de software (bibliotecas y archivos de encabezado, pero sin documentación), puede utilizar `nmake install_sw`.

Instalación de PCRE2

1. Descargue e instale CMake (disponible como instalador MSI). Durante la instalación, asegúrese de especificar `C:\Zabbix\CMake` como la carpeta de instalación y seleccione la opción *Add CMake to the PATH environment variable*.

2. Descargue el archivo fuente de PCRE2 y extráigalo en `C:\Zabbix\pcre2-10.45`.

3. Cree un directorio `build` en el directorio extraído de PCRE2 y navegue hasta él:

```
mkdir C:\Zabbix\pcre2-10.45\build
cd C:\Zabbix\pcre2-10.45\build
```

4. Configure PCRE2:

```
cmake -G "NMake Makefiles" -DPCRE_SUPPORT_UNICODE_PROPERTIES=ON -DCMAKE_BUILD_TYPE=Release -DCMAKE_INSTALL
```

Note:
Si ocurre algún error, se recomienda eliminar la caché de CMake antes de intentar repetir el proceso de compilación con CMake. La caché (`CMakeCachecache.txt`) puede encontrarse en el directorio de compilación del directorio extraído de PCRE2.

5. Compile PCRE2 usando NMake:

```
nmake
```

6. Instale PCRE2:

```
cmake --install .
```

Compilación del Zabbix agent

1. Descargue el archivo comprimido del código fuente de Zabbix y extráigalo en `C:\Zabbix\zabbix-8.0.0`.

Si necesita generar un archivo comprimido del código fuente a partir del repositorio de código fuente sin procesar (por ejemplo, para aplicar parches personalizados o compilar a partir del código fuente más reciente), ejecute los siguientes comandos en una máquina **Linux**:

```
git clone https://git.zabbix.com/scm/zbx/zabbix.git
cd zabbix
./bootstrap.sh
./configure
make dist
```

Esto creará un archivo comprimido del código fuente, que posteriormente podrá copiarse a una máquina Windows.

2. Vaya al directorio de compilación de Zabbix, abra el símbolo del sistema "x64 Native Tools Command Prompt" y compile el Zabbix agent (u otros componentes); asegúrese de especificar correctamente los directorios donde están instalados OpenSSL y PCRE2:

```
"%ProgramFiles(x86)%\Microsoft Visual Studio\2022\BuildTools\VC\Auxiliary\Build\vcvarsall.bat" x64

cd C:\Zabbix\zabbix-8.0.0\build\win32\project

#### Con compatibilidad con TLS:
nmake /K -f Makefile_agent PCRE2INCDIR="C:\Program Files\Zabbix\x64\PCRE2\include" PCRE2LIBDIR="C:\Program
nmake /K -f Makefile_get PCRE2INCDIR="C:\Program Files\Zabbix\x64\PCRE2\include" PCRE2LIBDIR="C:\Program F
nmake /K -f Makefile_sender PCRE2INCDIR="C:\Program Files\Zabbix\x64\PCRE2\include" PCRE2LIBDIR="C:\Progra

#### Sin compatibilidad con TLS:
nmake /K -f Makefile_agent PCRE2INCDIR="C:\Program Files\Zabbix\x64\PCRE2\include" PCRE2LIBDIR="C:\Program
nmake /K -f Makefile_get PCRE2INCDIR="C:\Program Files\Zabbix\x64\PCRE2\include" PCRE2LIBDIR="C:\Program F
nmake /K -f Makefile_sender PCRE2INCDIR="C:\Program Files\Zabbix\x64\PCRE2\include" PCRE2LIBDIR="C:\Progra
```

Encierre entre comillas las rutas que contengan espacios.

Después de la compilación, los binarios de los componentes de Zabbix se encontrarán en `C:\Zabbix\zabbix-8.0.0\bin\win64`.
El archivo de configuración del Zabbix agent se encuentra en `C:\Zabbix\zabbix-8.0.0\conf`.

Para ejecutar el agent, copie `zabbix_agentd.exe` y su archivo de configuración en una carpeta específica (por ejemplo, `C:\Zabbix\agent`) y, a continuación, ejecute el agent:

```
mkdir C:\Zabbix\agent
copy C:\Zabbix\zabbix-8.0.0\bin\win64\zabbix_agentd.exe C:\Zabbix\agent\
```

```
copy C:\Zabbix\zabbix-8.0.0\conf\zabbix_agentd.conf C:\Zabbix\agent\

C:\Zabbix\agent\zabbix_agentd.exe -c C:\Zabbix\agent\zabbix_agentd.conf -f
```

Instalación del agent en macOS desde PKG

Resumen

Zabbix agent se puede instalar en macOS mediante paquetes instaladores PKG, disponibles para descarga.

Los paquetes de Zabbix agent están disponibles con o sin cifrado.

Instalación del agent

El agent se puede instalar utilizando la interfaz gráfica de usuario o desde la línea de comandos, por ejemplo:

```
sudo installer -pkg zabbix_agent-8.0.0-macos-arm64-openssl.pkg -target /
```

Asegúrese de utilizar la versión correcta del paquete de Zabbix en el comando. Debe coincidir con el nombre del paquete descargado.

Ejecutar el agent

El agent se iniciará automáticamente después de la instalación o reinicio.

Puede editar el archivo de configuración en `/usr/local/etc/zabbix/zabbix_agentd.conf` si es necesario.

Para iniciar el agent manualmente, puede ejecutar:

```
sudo launchctl start com.zabbix.zabbix_agentd
```

Para detener el agent manualmente:

```
sudo launchctl stop com.zabbix.zabbix_agentd
```

Durante la actualización, el archivo de configuración existente no se sobrescribe. En su lugar, se crea un nuevo archivo `zabbix_agentd.conf.NEW` para que se utilice para revisar y actualizar el archivo de configuración existente, si es necesario. Recuerde reiniciar el agent después de realizar cambios manuales en el archivo de configuración.

Solución de problemas y eliminación del agent

Esta sección enumera algunos comandos útiles que se pueden utilizar para la solución de problemas y la eliminación de la instalación del agent de Zabbix.

Ver si el agent de Zabbix se está ejecutando:

```
ps aux | grep zabbix_agentd
```

Ver si el agent de Zabbix se ha instalado desde paquetes:

```
pkgutil --pkgs | grep zabbix
com.zabbix.pkg.ZabbixAgent
```

Ver los archivos que se instalaron desde el paquete instalador (tenga en cuenta que la / inicial no se muestra en esta vista):

```
pkgutil --only-files --files com.zabbix.pkg.ZabbixAgent
Library/LaunchDaemons/com.zabbix.zabbix_agentd.plist
usr/local/bin/zabbix_get
usr/local/bin/zabbix_sender
usr/local/etc/zabbix/zabbix_agentd/userparameter_examples.conf.NEW
usr/local/etc/zabbix/zabbix_agentd/userparameter_mysql.conf.NEW
usr/local/etc/zabbix/zabbix_agentd.conf.NEW
usr/local/sbin/zabbix_agentd
```

Detener el agent de Zabbix si se lanzó con `launchctl`:

```
sudo launchctl unload /Library/LaunchDaemons/com.zabbix.zabbix_agentd.plist
```

Eliminar archivos (incluyendo configuración y logs) que se instalaron con el paquete instalador:

```
sudo rm -f /Library/LaunchDaemons/com.zabbix.zabbix_agentd.plist
sudo rm -f /usr/local/sbin/zabbix_agentd
sudo rm -f /usr/local/bin/zabbix_get
```

```
sudo rm -f /usr/local/bin/zabbix_sender
sudo rm -rf /usr/local/etc/zabbix
sudo rm -rf /var/log/zabbix
```

Olvidar que el agent de Zabbix ha sido instalado:

```
sudo pkgutil --forget com.zabbix.pkg.ZabbixAgent
```

Compilar el agent de Zabbix en macOS

Descripción general

Esta sección muestra cómo compilar los binarios del agente de Zabbix para macOS desde el código fuente con o sin TLS.

Requisitos previos

Necesitará herramientas de desarrollo de línea de comandos (Xcode no es necesario), Automake, pkg-config y PCRE (v8.x) o PCRE2 (v10.x). Si desea compilar binarios de agent con TLS, también necesitará OpenSSL o GnuTLS.

Para instalar Automake y pkg-config, necesitará el gestor de paquetes Homebrew de https://brew.sh/. Para instalarlo, abra la terminal y ejecute el siguiente comando:

```
/bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
```

Luego instale Automake y pkg-config:

```
brew install automake
brew install pkg-config
```

La preparación de las bibliotecas PCRE, OpenSSL y GnuTLS depende de la forma en que se vayan a enlazar con el agent.

Si tiene previsto ejecutar binarios de agent en un equipo macOS que ya tenga estas bibliotecas, puede usar bibliotecas precompiladas proporcionadas por Homebrew. Normalmente, se trata de equipos macOS que usan Homebrew para compilar binarios de Zabbix agent o para otros fines.

Si los binarios de agent se van a usar en equipos macOS que no tienen la versión compartida de las bibliotecas, debe compilar bibliotecas estáticas a partir del código fuente y enlazar Zabbix agent con ellas.

Compilar binarios del agente con bibliotecas compartidas

Instale PCRE2 (reemplace *pcre2* por *pcre* en los comandos a continuación, si es necesario):

```
brew install pcre2
```

Al compilar con TLS, instale OpenSSL y/o GnuTLS:

```
brew install openssl
brew install gnutls
```

Descargue el código fuente de Zabbix:

```
git clone https://git.zabbix.com/scm/zbx/zabbix.git
```

Compile el agente sin TLS:

```
cd zabbix
./bootstrap.sh
./configure --sysconfdir=/usr/local/etc/zabbix --enable-agent --enable-ipv6
make
make install
```

Compile el agente con OpenSSL:

```
cd zabbix
./bootstrap.sh
./configure --sysconfdir=/usr/local/etc/zabbix --enable-agent --enable-ipv6 --with-openssl=/usr/local/opt/
make
make install
```

Compile el agente con GnuTLS:

```
cd zabbix-source/
./bootstrap.sh
./configure --sysconfdir=/usr/local/etc/zabbix --enable-agent --enable-ipv6 --with-gnutls=/usr/local/opt/g
```

```
make
make install
```

Compilar binarios del agente con bibliotecas estáticas sin TLS

Supongamos que las bibliotecas estáticas de PCRE se instalarán en `$HOME/static-libs`. Usaremos PCRE2 10.39.

```
PCRE_PREFIX="$HOME/static-libs/pcre2-10.39"
```

Descargue y compile PCRE con soporte para propiedades Unicode:

```
mkdir static-libs-source
cd static-libs-source
curl --remote-name https://github.com/PhilipHazel/pcre2/releases/download/pcre2-10.39/pcre2-10.39.tar.gz
tar xf pcre2-10.39.tar.gz
cd pcre2-10.39
./configure --prefix="$PCRE_PREFIX" --disable-shared --enable-static --enable-unicode-properties
make
make check
make install
```

Descargue el código fuente de Zabbix y compile el agente:

```
git clone https://git.zabbix.com/scm/zbx/zabbix.git
cd zabbix
./bootstrap.sh
./configure --sysconfdir=/usr/local/etc/zabbix --enable-agent --enable-ipv6 --with-libpcre2="$PCRE_PREFIX"
make
make install
```

Compilando binarios del agente con bibliotecas estáticas con OpenSSL

Al compilar OpenSSL, se recomienda ejecutar `make test` después de una compilación exitosa. Incluso si la compilación fue exitosa, a veces las pruebas fallan. Si este es el caso, se deben investigar y resolver los problemas antes de continuar.

Supongamos que las bibliotecas estáticas de PCRE y OpenSSL se instalarán en `$HOME/static-libs`. Usaremos PCRE2 10.39 y OpenSSL 1.1.1a.

```
PCRE_PREFIX="$HOME/static-libs/pcre2-10.39"
OPENSSL_PREFIX="$HOME/static-libs/openssl-1.1.1a"
```

Compilaremos las bibliotecas estáticas en `static-libs-source`:

```
mkdir static-libs-source
cd static-libs-source
```

Descargue y compile PCRE con soporte para propiedades Unicode:

```
curl --remote-name https://github.com/PhilipHazel/pcre2/releases/download/pcre2-10.39/pcre2-10.39.tar.gz
tar xf pcre2-10.39.tar.gz
cd pcre2-10.39
./configure --prefix="$PCRE_PREFIX" --disable-shared --enable-static --enable-unicode-properties
make
make check
make install
cd ..
```

Descargue y compile OpenSSL:

```
curl --remote-name https://www.openssl.org/source/openssl-1.1.1a.tar.gz
tar xf openssl-1.1.1a.tar.gz
cd openssl-1.1.1a
./Configure --prefix="$OPENSSL_PREFIX" --openssldir="$OPENSSL_PREFIX" --api=1.1.0 no-shared no-capieng no-
make
make test
make install_sw
cd ..
```

Descargue el código fuente de Zabbix y compile el agente:

```
git clone https://git.zabbix.com/scm/zbx/zabbix.git
cd zabbix
```

```
./bootstrap.sh
./configure --sysconfdir=/usr/local/etc/zabbix --enable-agent --enable-ipv6 --with-libpcre2="$PCRE_PREFIX"
make
make install
```

Compilación de binarios del agente con bibliotecas estáticas con GnuTLS

GnuTLS depende del backend criptográfico Nettle y de la biblioteca aritmética GMP. En lugar de usar la biblioteca GMP completa, esta guía usará mini-gmp que está incluida en Nettle.

Al compilar GnuTLS y Nettle, se recomienda ejecutar `make check` después de una compilación exitosa. Incluso si la compilación fue exitosa, las pruebas a veces fallan. Si este es el caso, se deben investigar y resolver los problemas antes de continuar.

Supongamos que las bibliotecas estáticas de PCRE, Nettle y GnuTLS se instalarán en `$HOME/static-libs`. Usaremos PCRE2 10.39, Nettle 3.4.1 y GnuTLS 3.6.5.

```
PCRE_PREFIX="$HOME/static-libs/pcre2-10.39"
NETTLE_PREFIX="$HOME/static-libs/nettle-3.4.1"
GNUTLS_PREFIX="$HOME/static-libs/gnutls-3.6.5"
```

Compilaremos las bibliotecas estáticas en `static-libs-source`:

```
mkdir static-libs-source
cd static-libs-source
```

Descargar y compilar Nettle:

```
curl --remote-name https://ftp.gnu.org/gnu/nettle/nettle-3.4.1.tar.gz
tar xf nettle-3.4.1.tar.gz
cd nettle-3.4.1
./configure --prefix="$NETTLE_PREFIX" --enable-static --disable-shared --disable-documentation --disable-a
make
make check
make install
cd ..
```

Descargar y compilar GnuTLS:

```
curl --remote-name https://www.gnupg.org/ftp/gcrypt/gnutls/v3.6/gnutls-3.6.5.tar.xz
tar xf gnutls-3.6.5.tar.xz
cd gnutls-3.6.5
PKG_CONFIG_PATH="$NETTLE_PREFIX/lib/pkgconfig" ./configure --prefix="$GNUTLS_PREFIX" --enable-static --dis
make
make check
make install
cd ..
```

Descargar el código fuente de Zabbix y compilar el agente:

```
git clone https://git.zabbix.com/scm/zbx/zabbix.git
cd zabbix
./bootstrap.sh
CFLAGS="-Wno-unused-command-line-argument -framework Foundation -framework Security" \
> LIBS="-lgnutls -lhogweed -lnettle" \
> LDFLAGS="-L$GNUTLS_PREFIX/lib -L$NETTLE_PREFIX/lib" \
> ./configure --sysconfdir=/usr/local/etc/zabbix --enable-agent --enable-ipv6 --with-libpcre2="$PCRE_PREFI
make
make install
```

Guía de implementación de AWS para Zabbix server

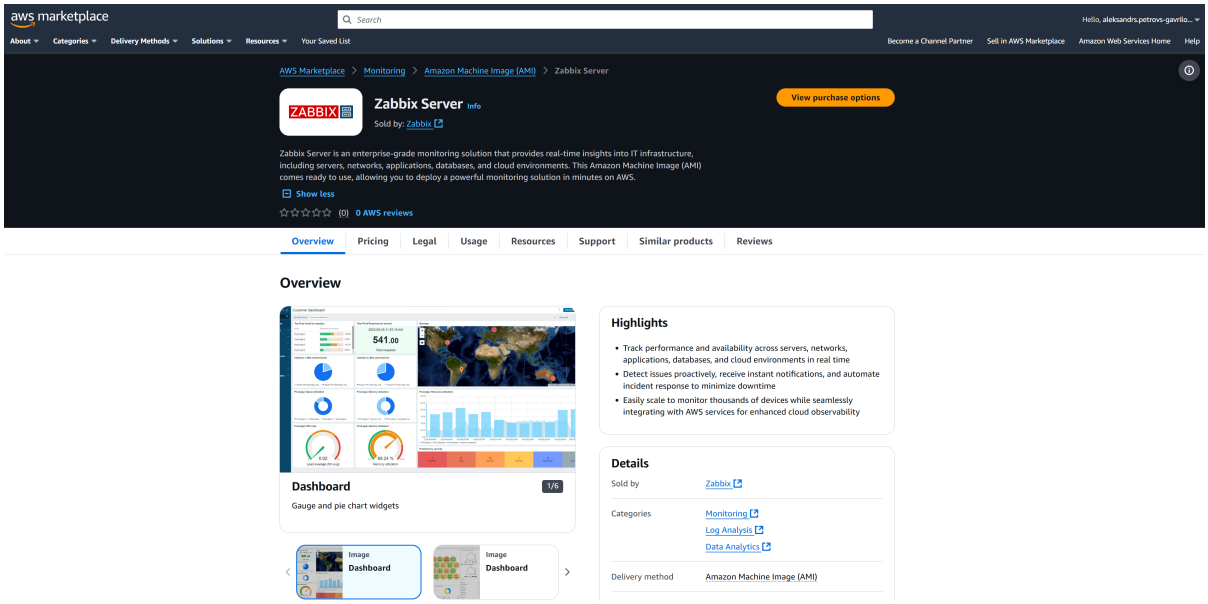
Descripción general

Esta página describe cómo implementar Zabbix server mediante AWS.

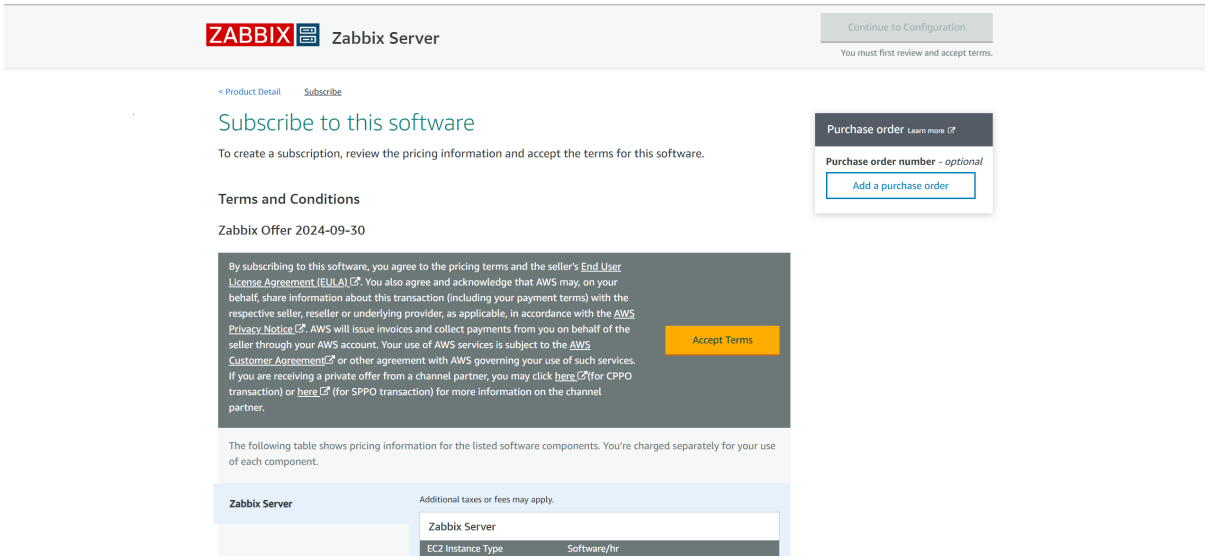
> **Attention:**
> Hay varias versiones de Zabbix disponibles para su implementación en la plataforma. Asegúrese de leer la documentación correspondiente a la misma versión de Zabbix que ha implementado. Puede seleccionar la documentación de distintas versiones de Zabbix en el menú desplegable de versiones.

Implementación del appliance

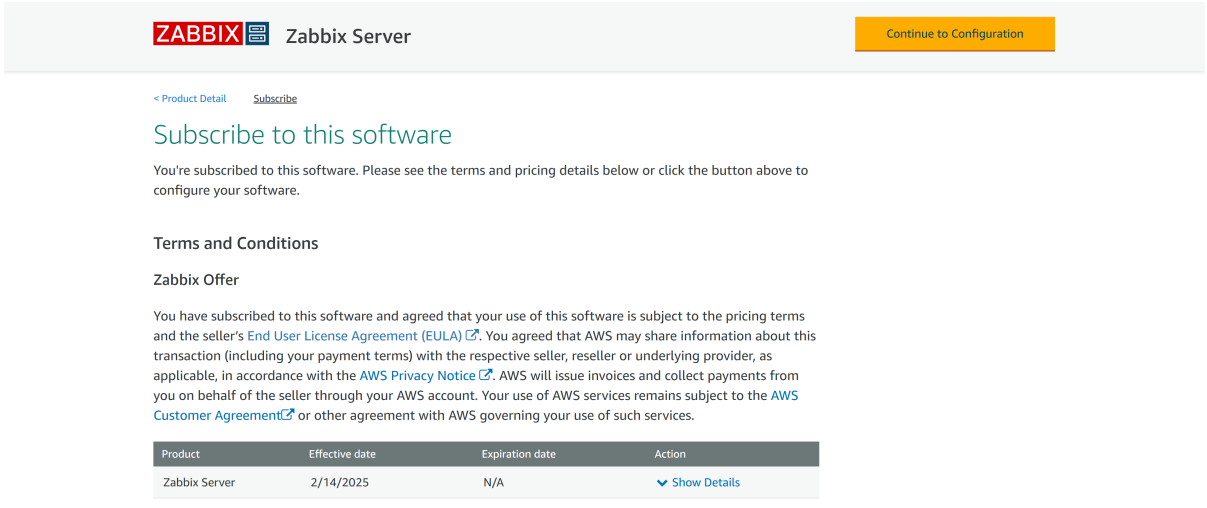
Para implementar el appliance de Zabbix, primero, vaya a la página de Zabbix en AWS Marketplace. A continuación, haga clic en *View purchase options* para implementar la imagen y continuar a la página de suscripción.



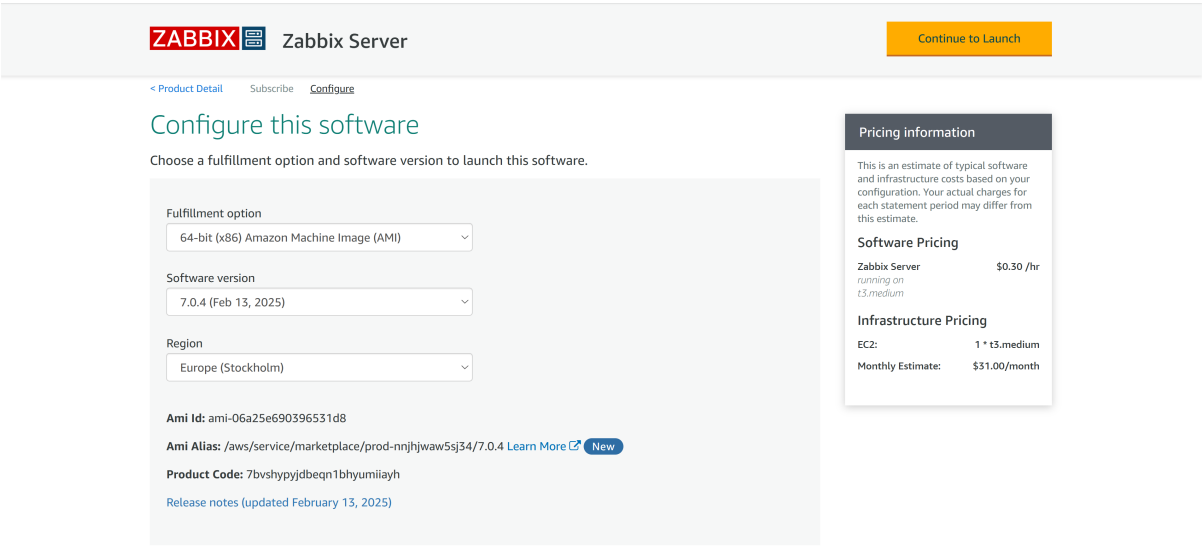
Haga clic en el botón *Accept terms* para aceptar los *Terms and Conditions*.



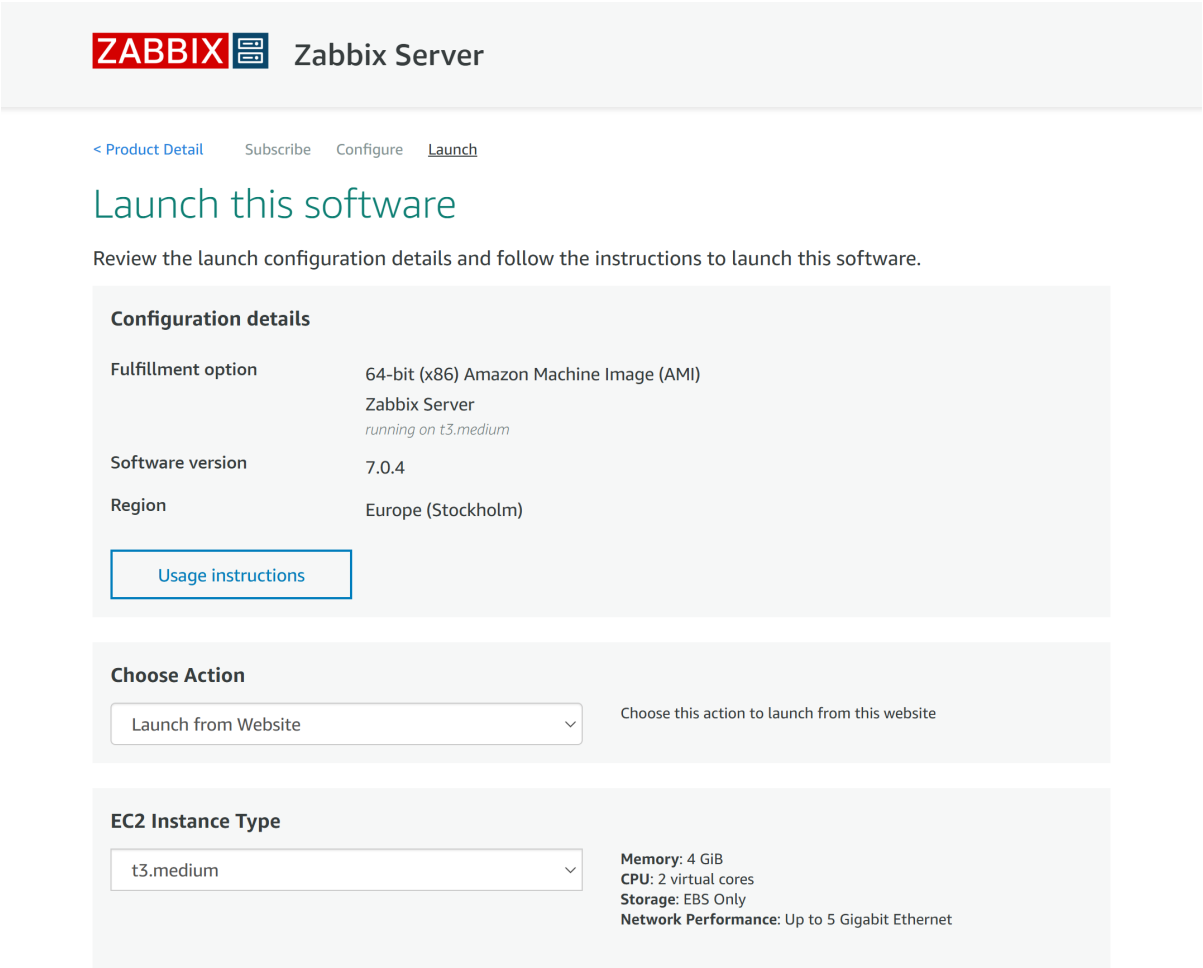
Haga clic en el botón *Continue to Configuration* para pasar a la sección Configuration del appliance de Zabbix.



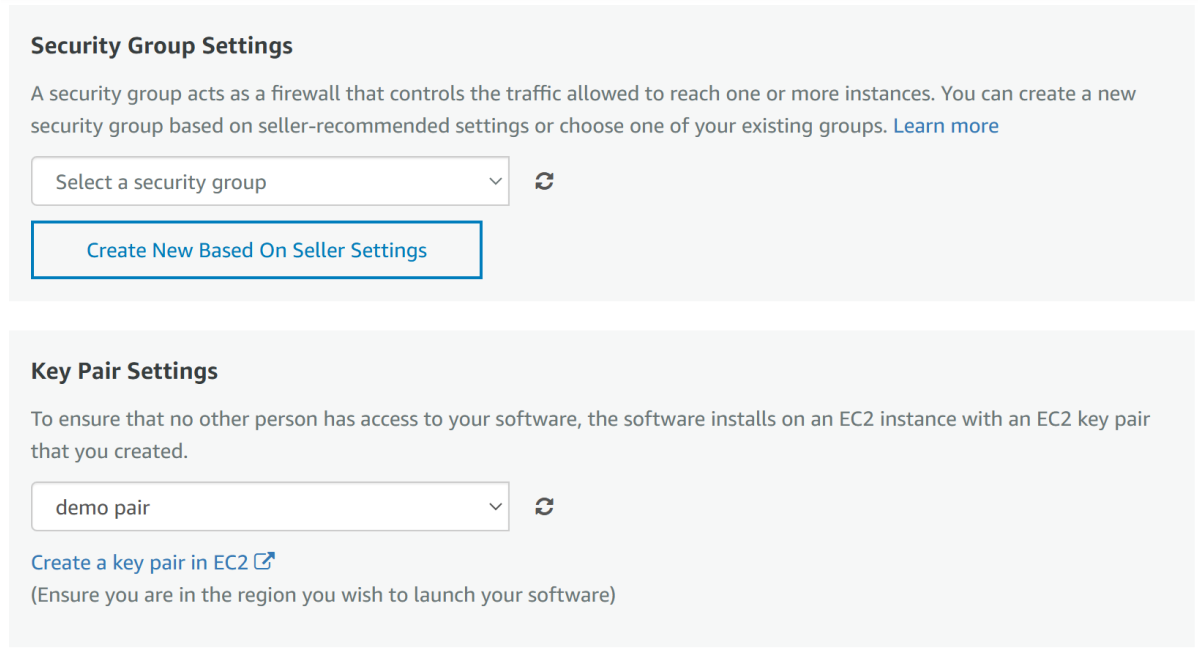
Seleccione la opción Fulfilment, la versión de Zabbix y la Región que prefiera.



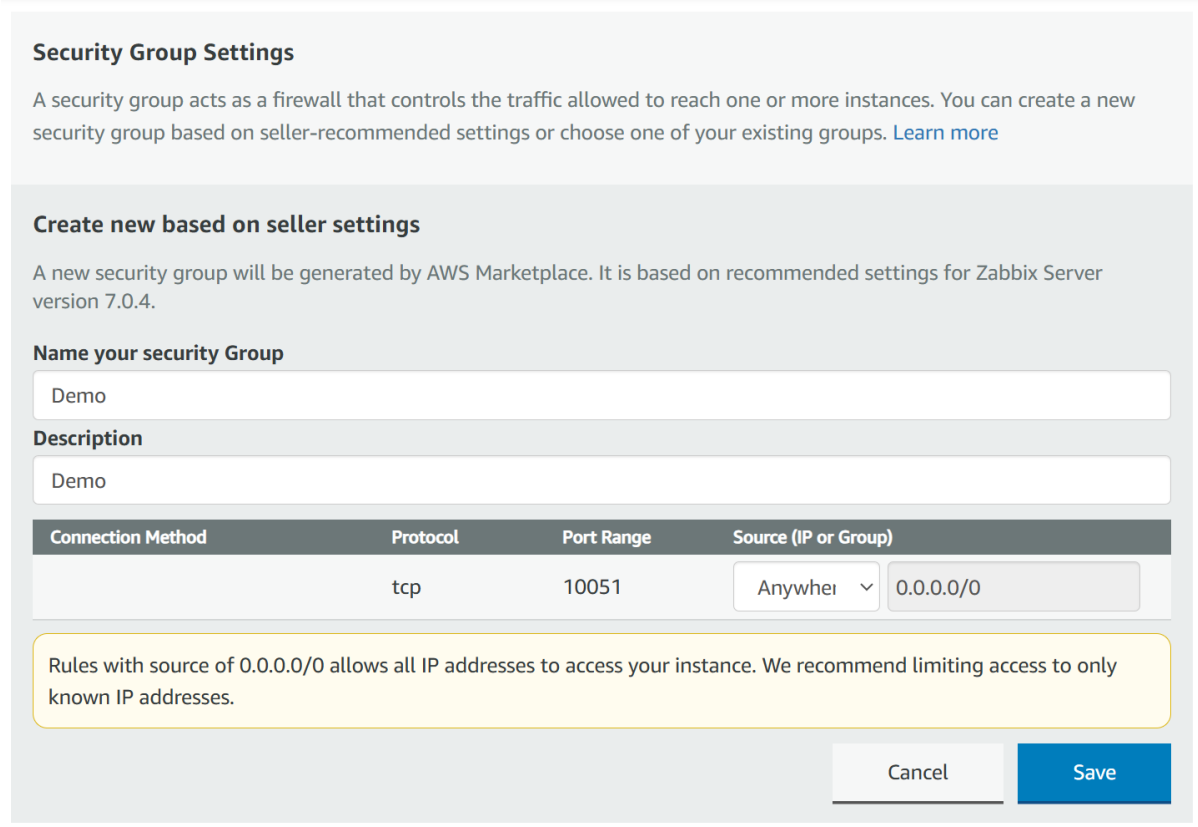
Haga clic en el botón *Continue to Launch*. En la página Launch, seleccione la opción Launch que prefiera, el tipo de instancia EC2 y la configuración de VPC y Subnet.



En la parte inferior de la página Launch, especifique la configuración de Security Group y seleccione o cree un par de claves de EC2. ¡Asegúrese de guardar su clave privada en un lugar seguro!



Si selecciona *Create a new Security Group based on seller settings*, se permitirá el puerto TCP 10051 (Zabbix server). Nota: Si desea acceder a la instancia mediante SSH y HTTP/HTTPS (web frontend), también deberá permitir las conexiones SSH y HTTP/HTTPS en la configuración de su Security Group.



Enhorabuena, su appliance de Zabbix se ha implementado correctamente.

ZABBIX Zabbix Server

Congratulations! An instance of this software is successfully deployed on EC2!

AMI ID: ami-06a25e690396531d8 (View Launch Configuration Details)
You can view this instance on EC2 Console. You can also view all instances on Your Software. Software and AWS hourly usage fees apply when the instance is running and will appear on your monthly bill.

You can launch this configuration again below or go to the configuration page to start a new one.

Configuration details

Fulfillment option	64-bit (x86) Amazon Machine Image (AMI) Zabbix Server *running on t3.medium*
Software version	7.0.4
Region	Europe (Stockholm)

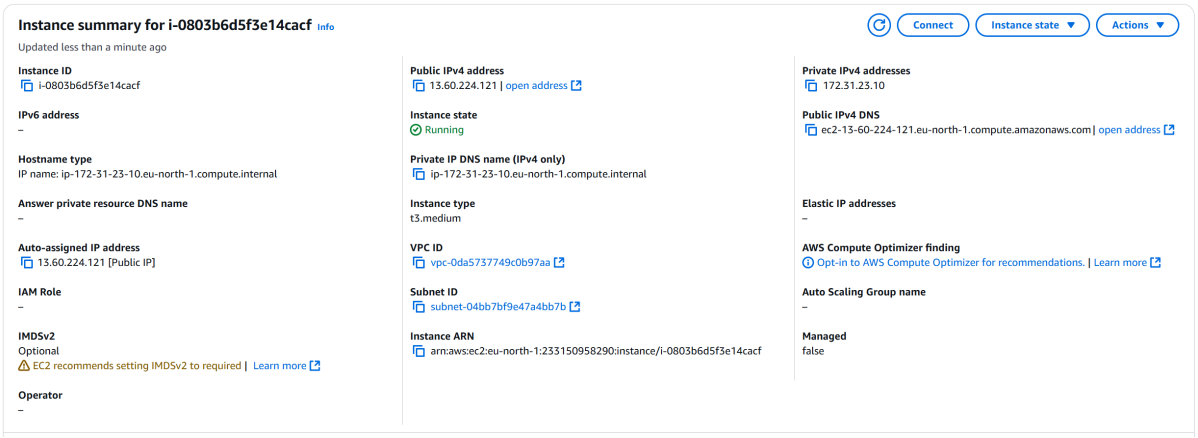
Usage instructions

Choose Action

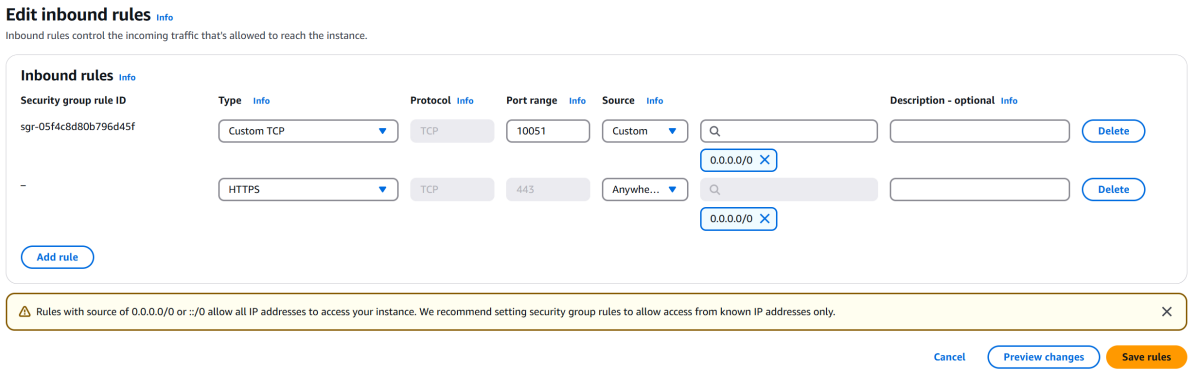
Select a launch action

Acceso al appliance

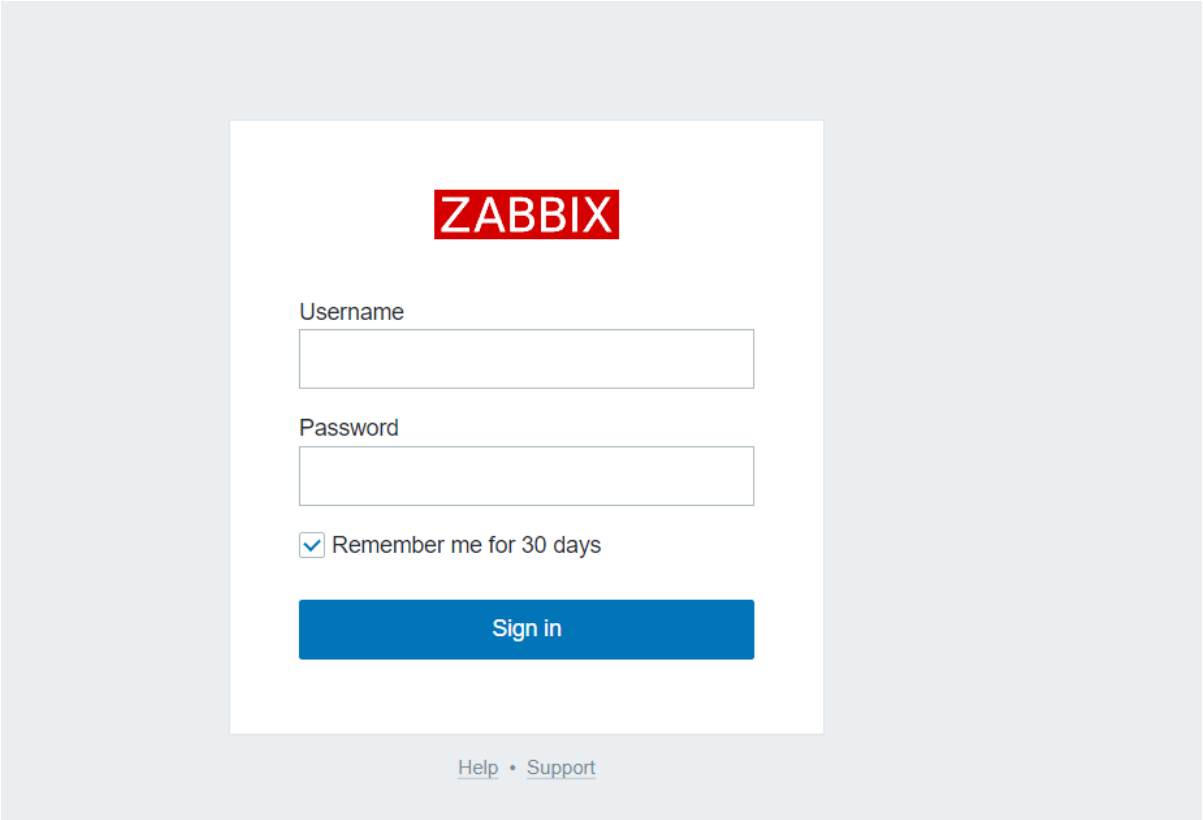
Abra la sección *Instances* y seleccione la instancia de Zabbix appliance recién implementada.



Para acceder al frontend de Zabbix mediante la dirección IPv4 pública o el DNS IPv4 público de la instancia, permita el tráfico HTTPS entrante en el puerto 443 en el grupo de seguridad.



Ahora podemos acceder al frontend de Zabbix abriendo la dirección IPv4 pública o el DNS IPv4 público de Zabbix appliance.



Para obtener la contraseña, debemos conectarnos a la máquina mediante la consola serie o SSH. El nombre de usuario SSH predeterminado es rocky. No olvide utilizar el par de claves que seleccionó o generó anteriormente. Tenga en cuenta que la generación de las credenciales puede tardar. La generación de credenciales puede tardar unos minutos. Espere unos minutos antes de intentar conectarse.

```
Last login: Tue Nov 16 15:16:07 2021 from ec2-13-48-4-200.eu-north-1.compute.amazonaws.com
********************************************************************************

Zabbix frontend credentials:

Username: Admin

Password: gJ8f7XpgJbsnc2BU48g

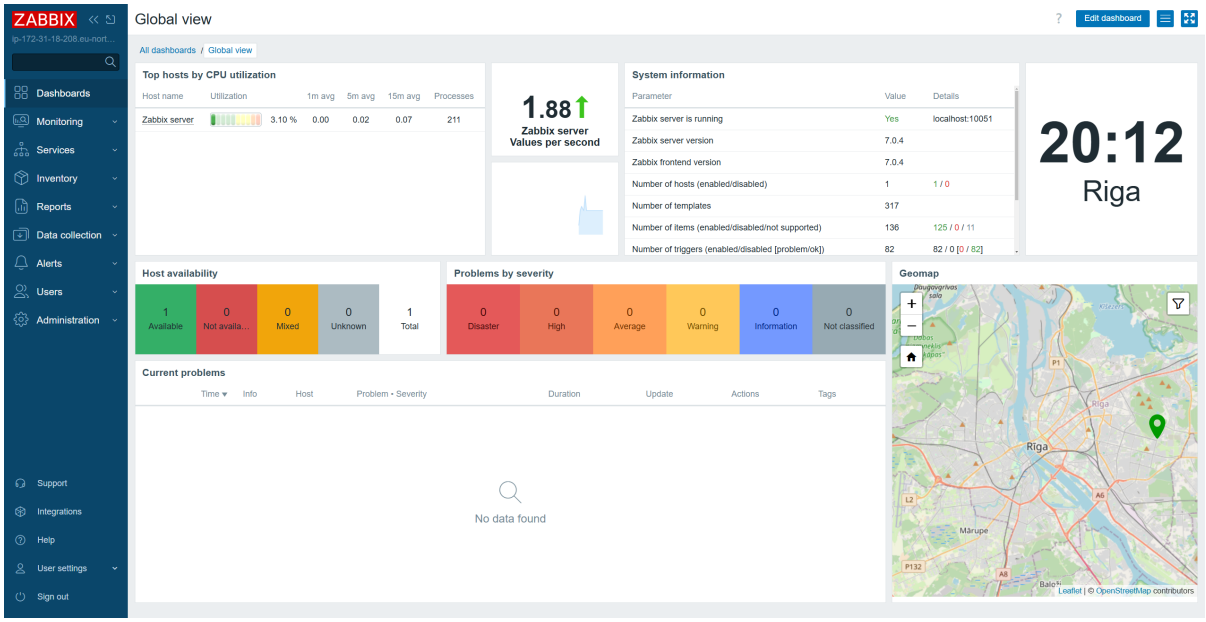
To learn about available professional services, including technical suppport and training, please visit https://www.zabbix.com/services

Official Zabbix documentation available at https://www.zabbix.com/documentation/current/

Note! Do not forget to change timezone PHP variable in /etc/php.d/99-zabbix.ini file.

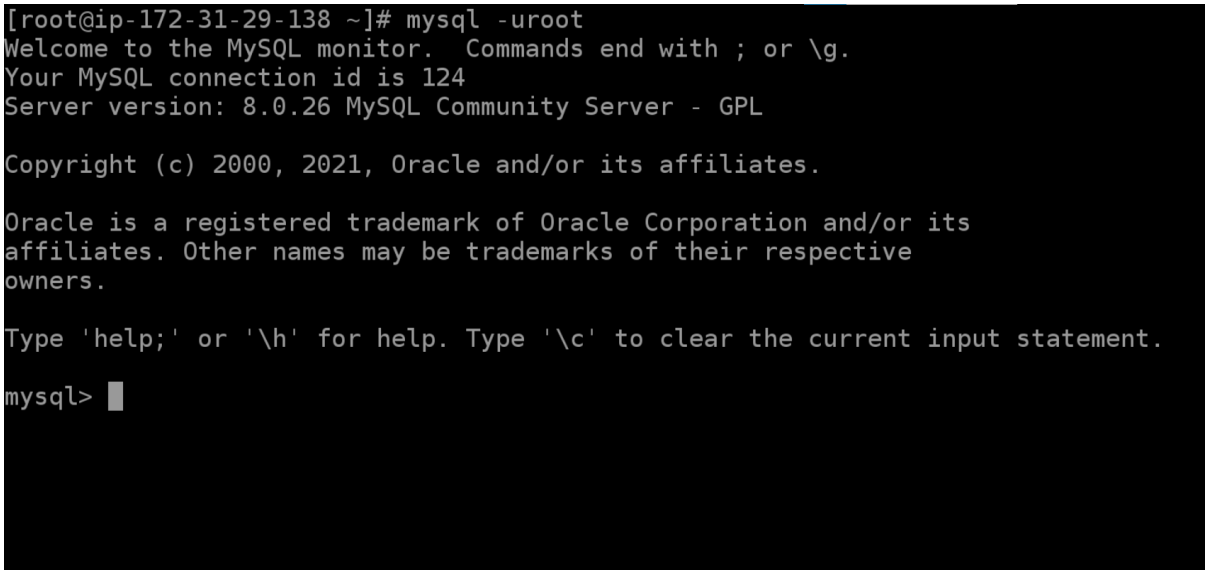
********************************************************************************
```

Ahora puede utilizar esta información para iniciar sesión en su frontend de Zabbix.



Acceso a la base de datos

Puede acceder a la base de datos de Zabbix utilizando el usuario `root` sin introducir una contraseña.



Las credenciales se almacenan en el archivo de configuración `/root/.my.cnf`.

La zona horaria predeterminada del frontend de Zabbix se puede cambiar en *Administración* > *General* > *GUI*

Guía de implementación de AWS para el proxy de Zabbix

Descripción general

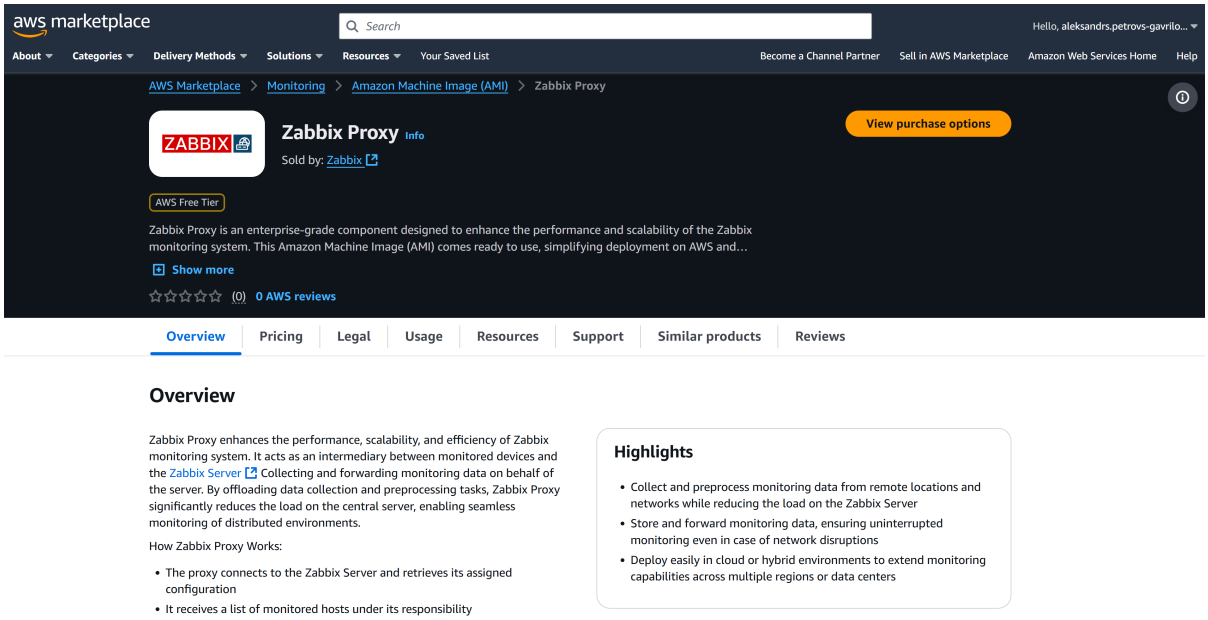
Esta página describe cómo implementar un proxy de Zabbix mediante AWS.

> **Attention:**
> Hay varias versiones de Zabbix disponibles para su implementación en la plataforma. Asegúrese de leer la documentación correspondiente a la misma versión de Zabbix que ha implementado. Puede seleccionar la documentación de diferentes versiones de Zabbix en el menú desplegable de versiones.

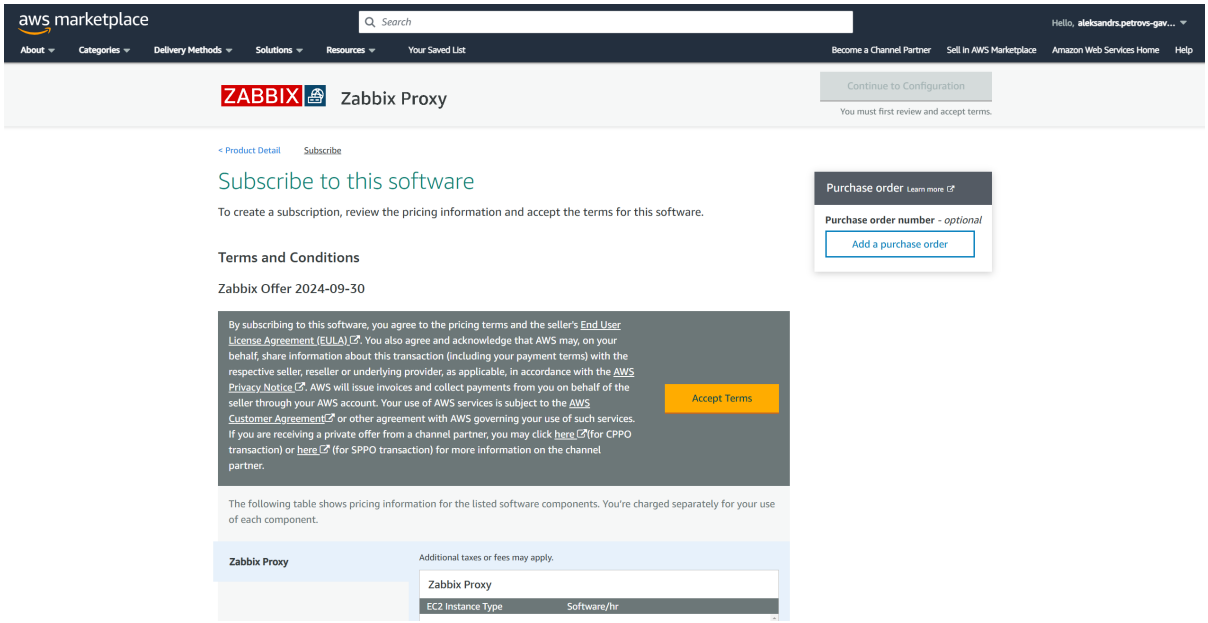
Implementación de Zabbix proxy

> **Attention:**
> Se requiere una instancia operativa de Zabbix server para utilizar el componente Zabbix proxy.

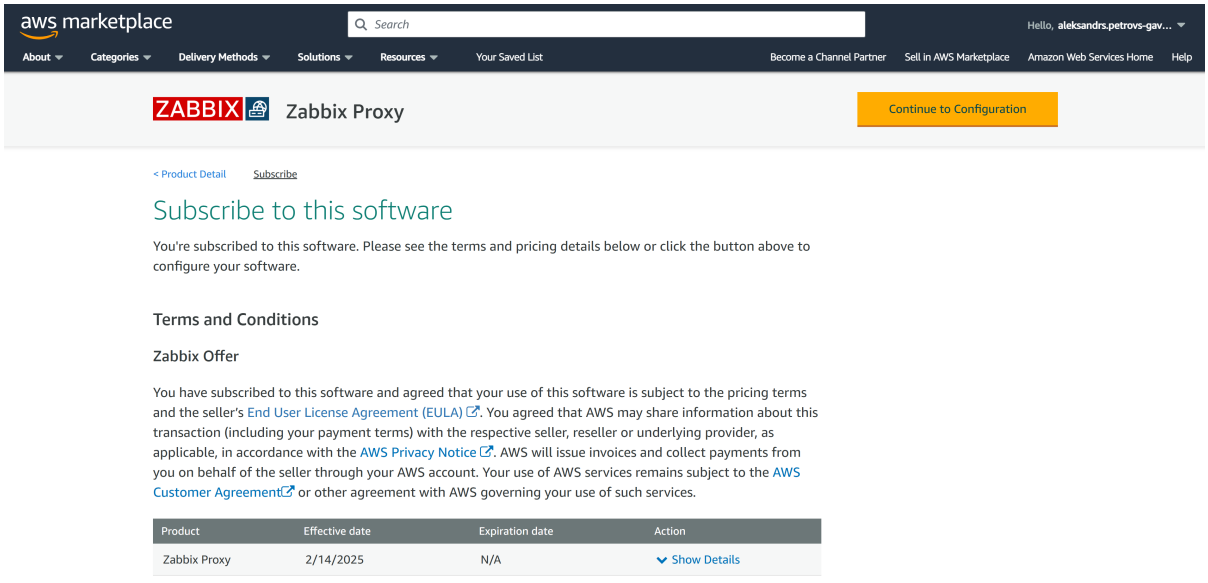
Para implementar Zabbix proxy, primero vaya a la página de Zabbix en AWS Marketplace. A continuación, haga clic en *View purchase options* para implementar la imagen y continuar a la página de suscripción.



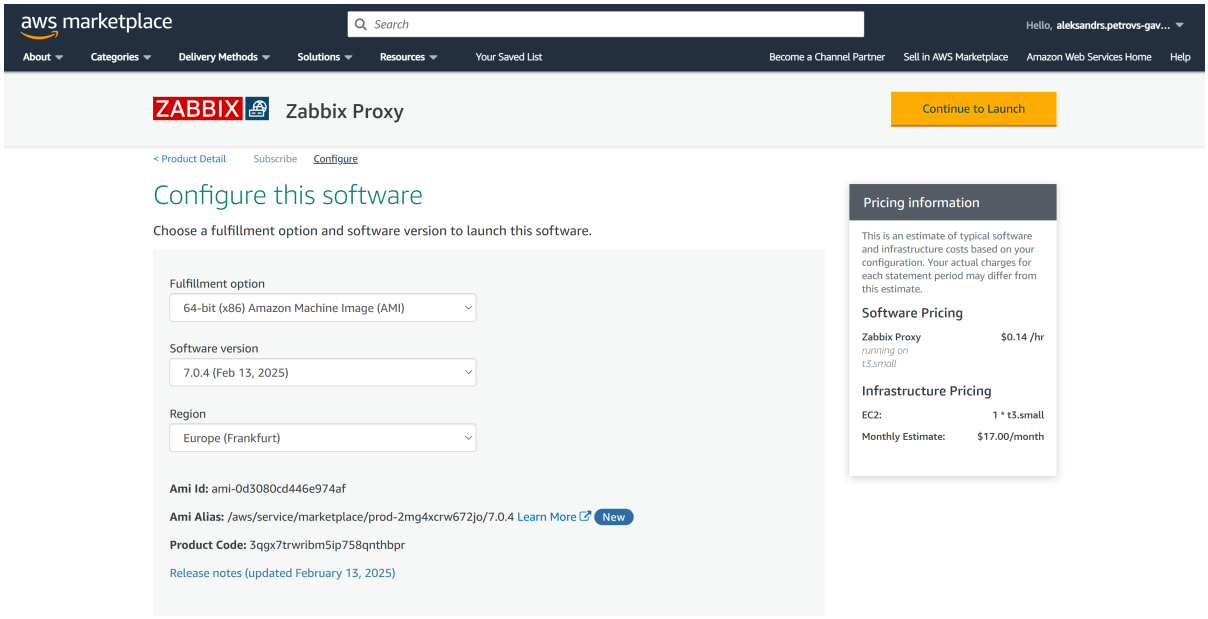
Haga clic en el botón *Accept terms* para aceptar los *Terms and Conditions*.



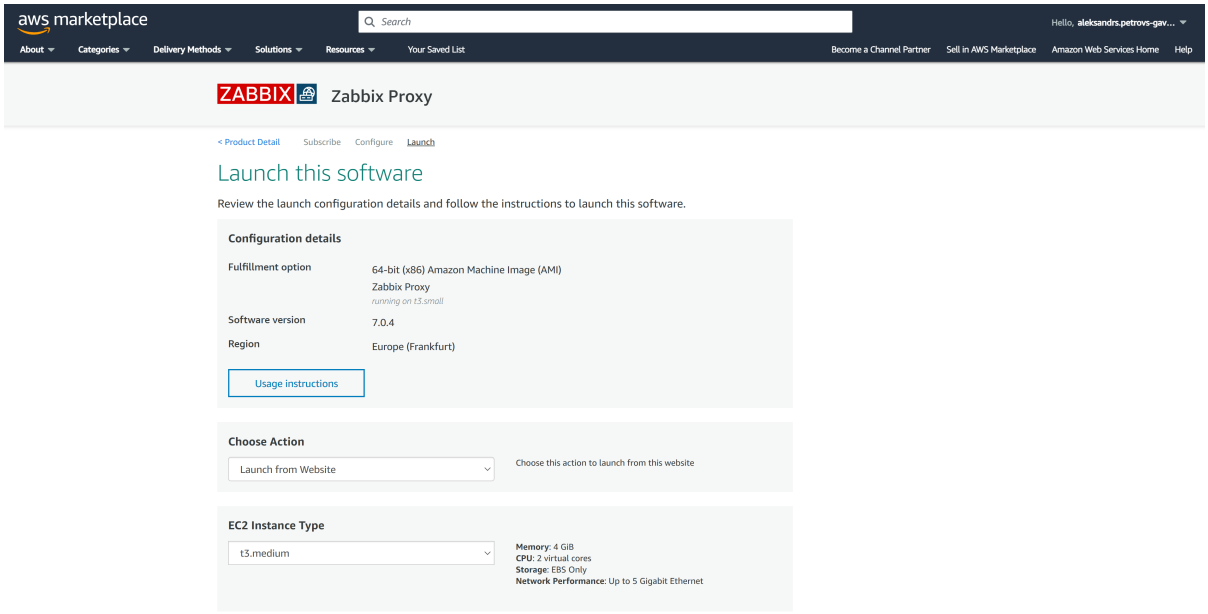
Haga clic en el botón *Continue to Configuration* para pasar a la sección de configuración de Zabbix proxy.



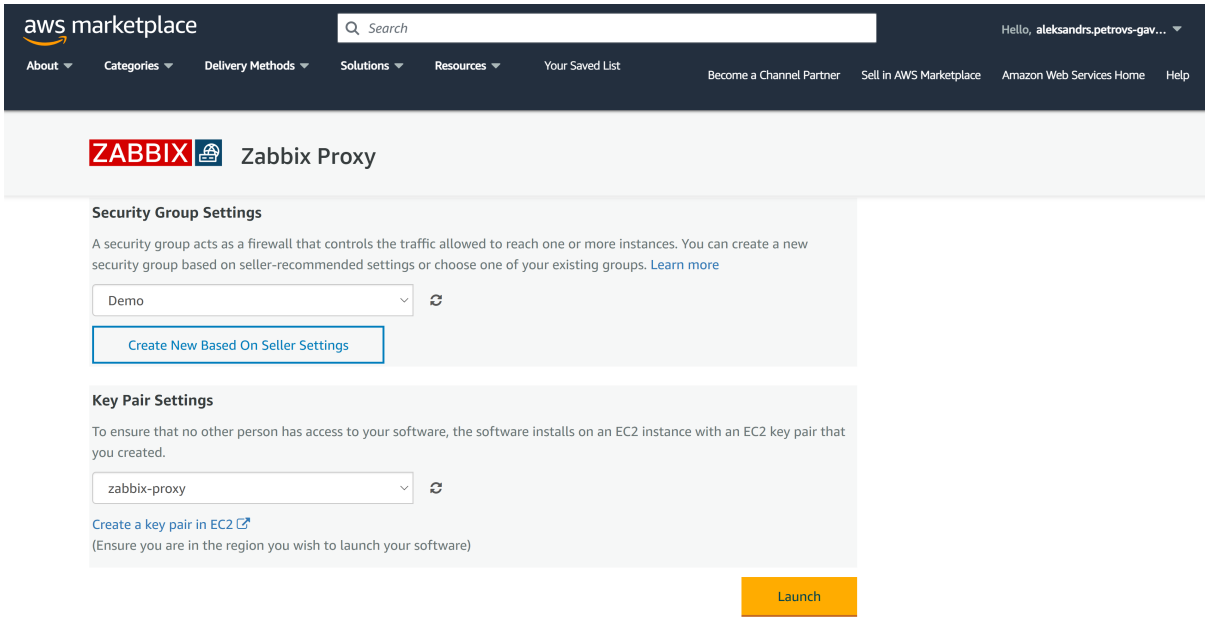
Seleccione la opción de Fulfilment, la versión de Zabbix y la Region que prefiera.



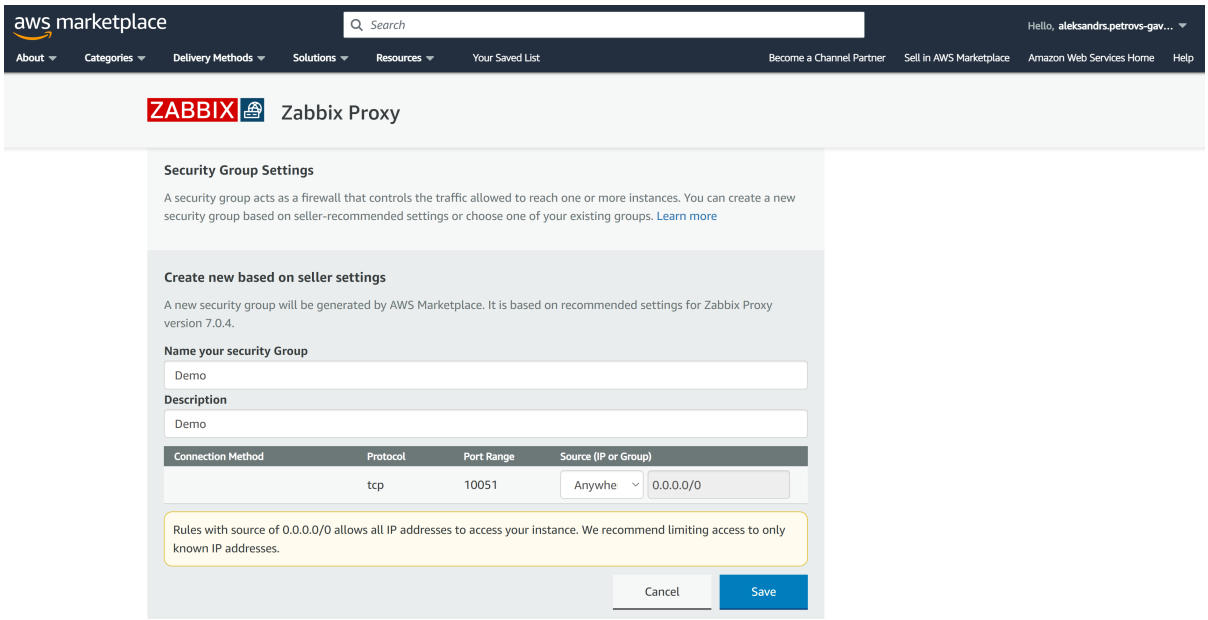
Haga clic en el botón *Continue to Launch*. En la página Launch, seleccione la opción Launch, el EC2 Instance Type, la VPC y la configuración de Subnet que prefiera.



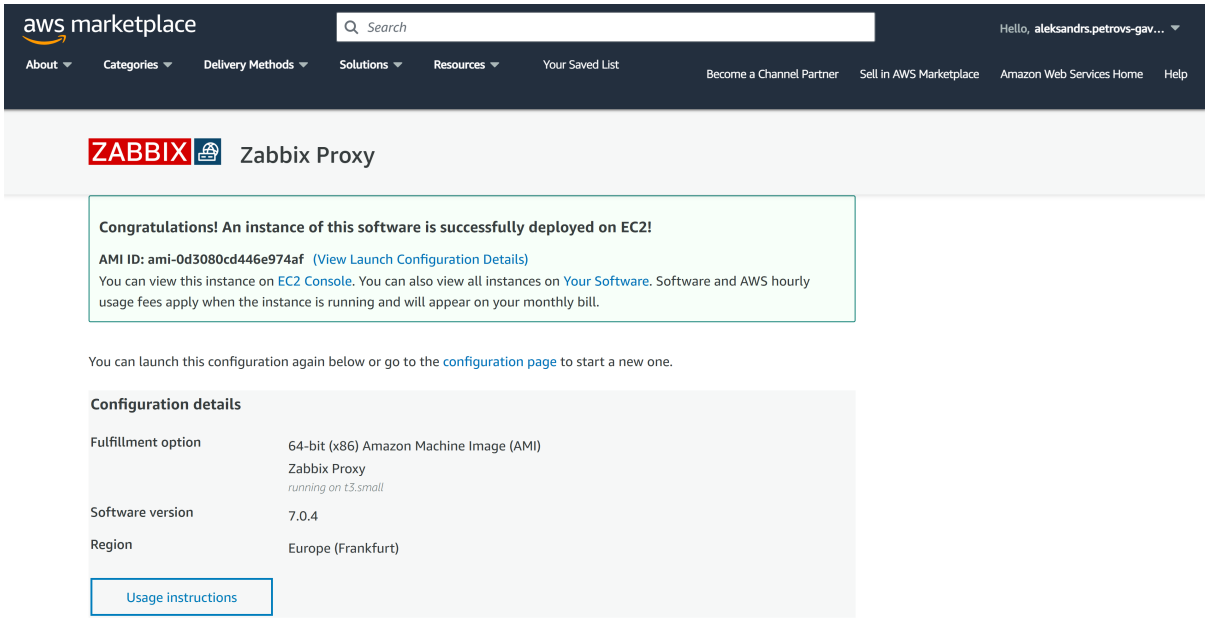
En la parte inferior de la página Launch, especifique la configuración de Security Group y seleccione o cree un par de claves de EC2. Asegúrese de guardar su clave privada en un lugar seguro.



Si selecciona *Create a new Security Group based on the Seller settings*, se permitirá el puerto TCP 10051 (Zabbix server). Nota: Si desea acceder a la instancia mediante SSH, también deberá permitir las conexiones SSH en la configuración de Security Group.

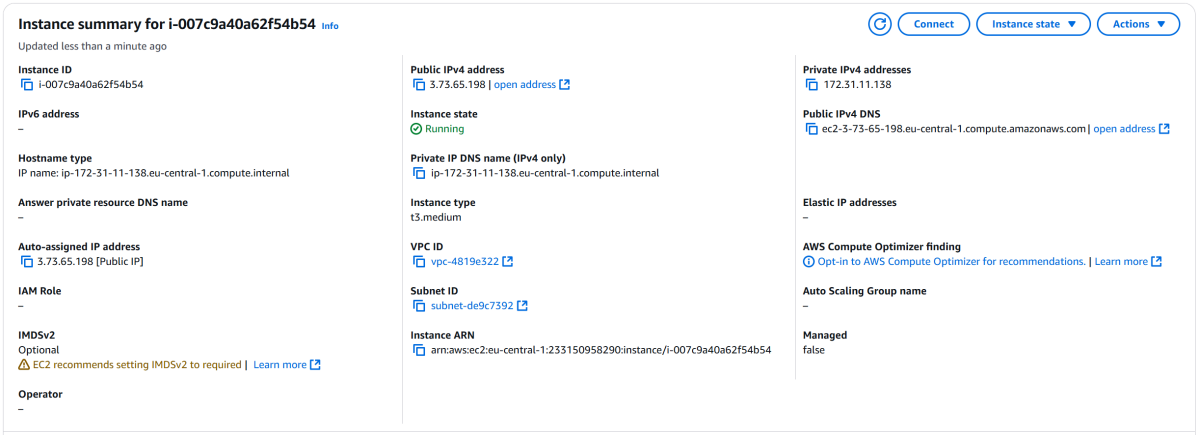


Por último, haga clic en el botón *Launch*. Enhorabuena, su Zabbix proxy se está implementando.



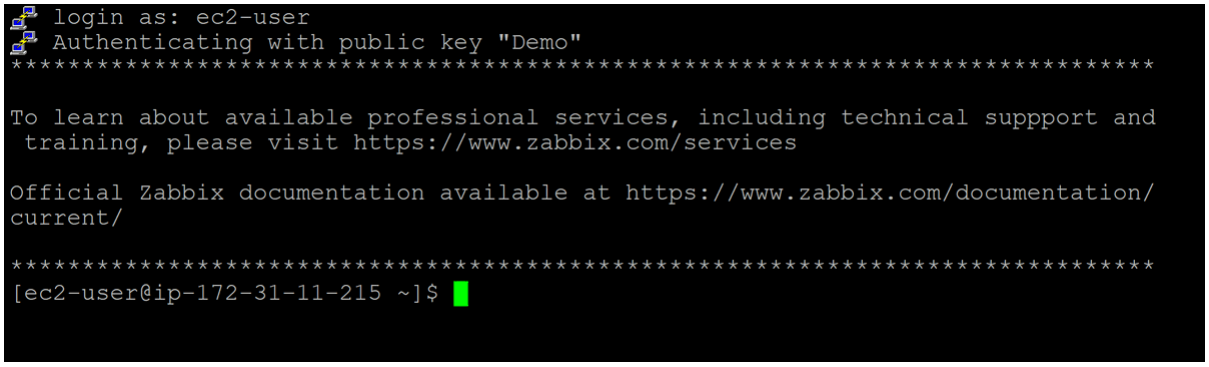
Configuración y acceso al proxy de Zabbix

Abra la sección Instances en la consola de EC2 y seleccione la instancia de proxy de Zabbix recién implementada.



Conéctese a la dirección IPv4 pública o al DNS IPv4 público del proxy de Zabbix mediante SSH. El nombre de usuario SSH predeterminado es `rocky`. No olvide utilizar el par de claves que generó anteriormente.

La generación de credenciales puede tardar unos minutos. Espere unos minutos antes de intentar conectarse.



El archivo de configuración del proxy de Zabbix se encuentra en `/etc/zabbix/zabbix_proxy.conf`.

El archivo de configuración se puede abrir con cualquier editor de texto, por ejemplo, vi:

```
sudo vi /etc/zabbix/zabbix_proxy.conf
```

La configuración inicial del proxy consiste en modificar los parámetros `ProxyMode`, `Server` y `Hostname` en el archivo de configuración del proxy de Zabbix.

El modo del proxy se puede cambiar editando el parámetro `ProxyMode`. Establezca `ProxyMode=0` para el modo activo o `ProxyMode=1` para el modo pasivo.

El parámetro `Server` debe contener la dirección IP del server de Zabbix. Por ejemplo:

```
Server=192.168.50.141
```

El parámetro `Hostname` debe contener el nombre de host del proxy de Zabbix. El nombre de host definido aquí debe coincidir con el nombre del proxy proporcionado durante la configuración del proxy en el frontend de Zabbix. Por ejemplo:

```
Hostname=EC2-Frankfurt
```

Una vez completada la configuración, reinicie el proxy de Zabbix con el siguiente comando:

```
sudo systemctl restart zabbix-proxy
```

Nota: utilice `sudo` para editar archivos de configuración y ejecutar comandos desde la línea de comandos.

```
### Option: ProxyMode
#       Proxy operating mode.
#       0 - proxy in the active mode
#       1 - proxy in the passive mode
#
# Mandatory: no
# Default:
# ProxyMode=0

### Option: Server
#       If ProxyMode is set to active mode:
#               IP address or DNS name of Zabbix server to get configuration data from and send data to.
#       If ProxyMode is set to passive mode:
#               List of comma delimited IP addresses, optionally in CIDR notation, or DNS names of Zabbix
server.
#               Incoming connections will be accepted only from the addresses listed here.
#               If IPv6 support is enabled then '127.0.0.1', '::127.0.0.1', '::ffff:127.0.0.1' are treated
 equally
#               and '::/0' will allow any IPv4 or IPv6 address.
#               '0.0.0.0/0' can be used to allow any IPv4 address.
#               Example: Server=127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.example.com
#
# Mandatory: yes
# Default:
# Server=

Server=127.0.0.1
```

```
### Option: Hostname
#       Unique, case sensitive Proxy name. Make sure the Proxy name is known to the server!
#       Value is acquired from HostnameItem if undefined.
#
# Mandatory: no
# Default:
# Hostname=

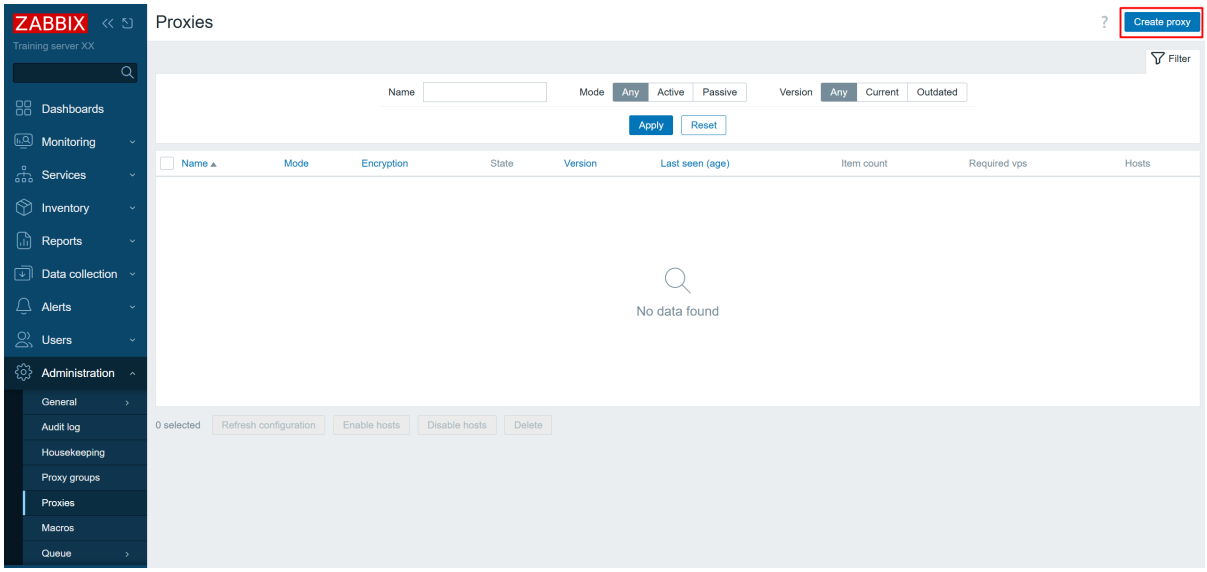
Hostname=EC2-Frankfurt
```

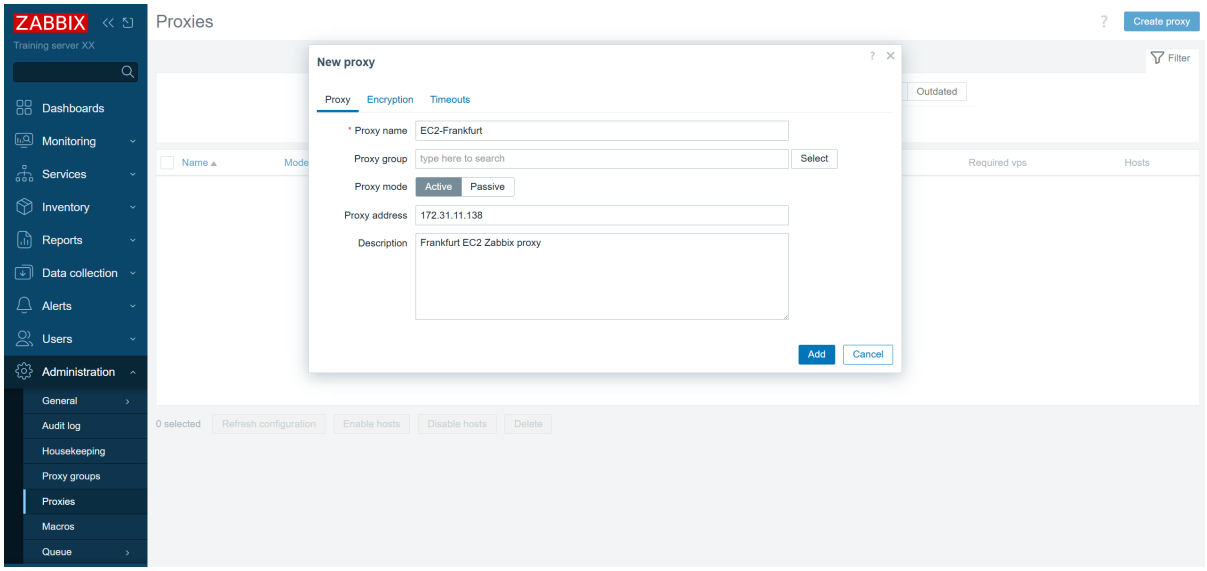
Abra el frontend de Zabbix y vaya a *Administration* > *Proxies*. Haga clic en el botón *Create proxy*.

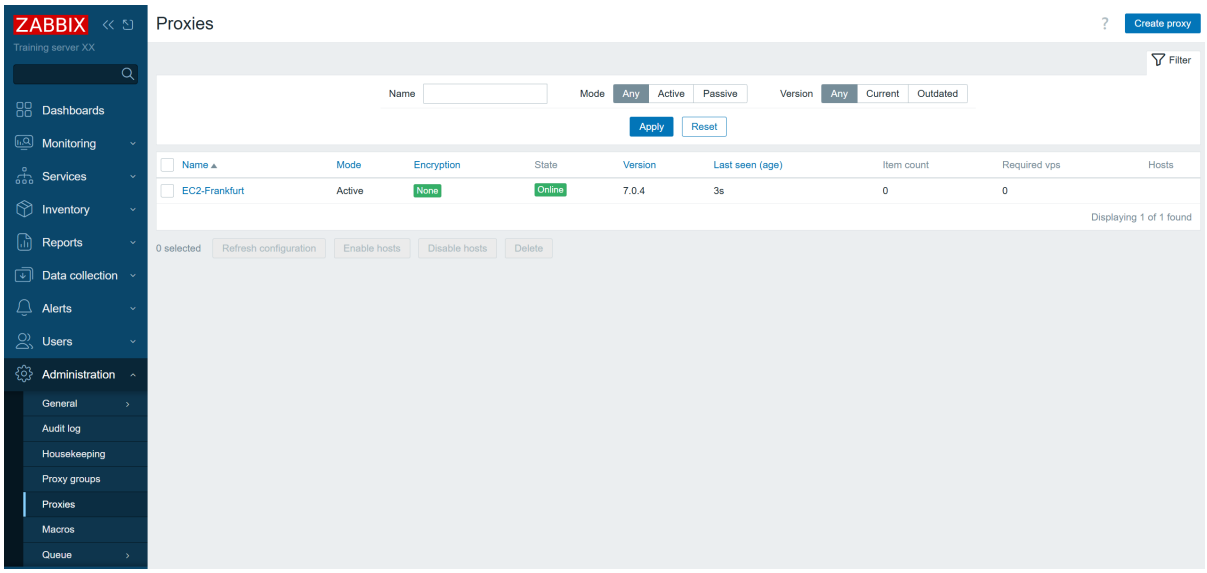
Introduzca el mismo nombre del proxy que el valor del parámetro `Hostname` en el archivo de configuración del proxy.

Seleccione el modo del proxy y proporcione la dirección IP del proxy (opcional para los proxies activos).

Haga clic en *Create proxy*.







Ubicación de la base de datos del proxy

La base de datos SQLite del proxy se almacena en `/tmp/zabbix_proxy.sqlite3` y se especifica mediante el parámetro `DBName` en el archivo de configuración del proxy de Zabbix.

Guía de implementación de Zabbix server en Azure

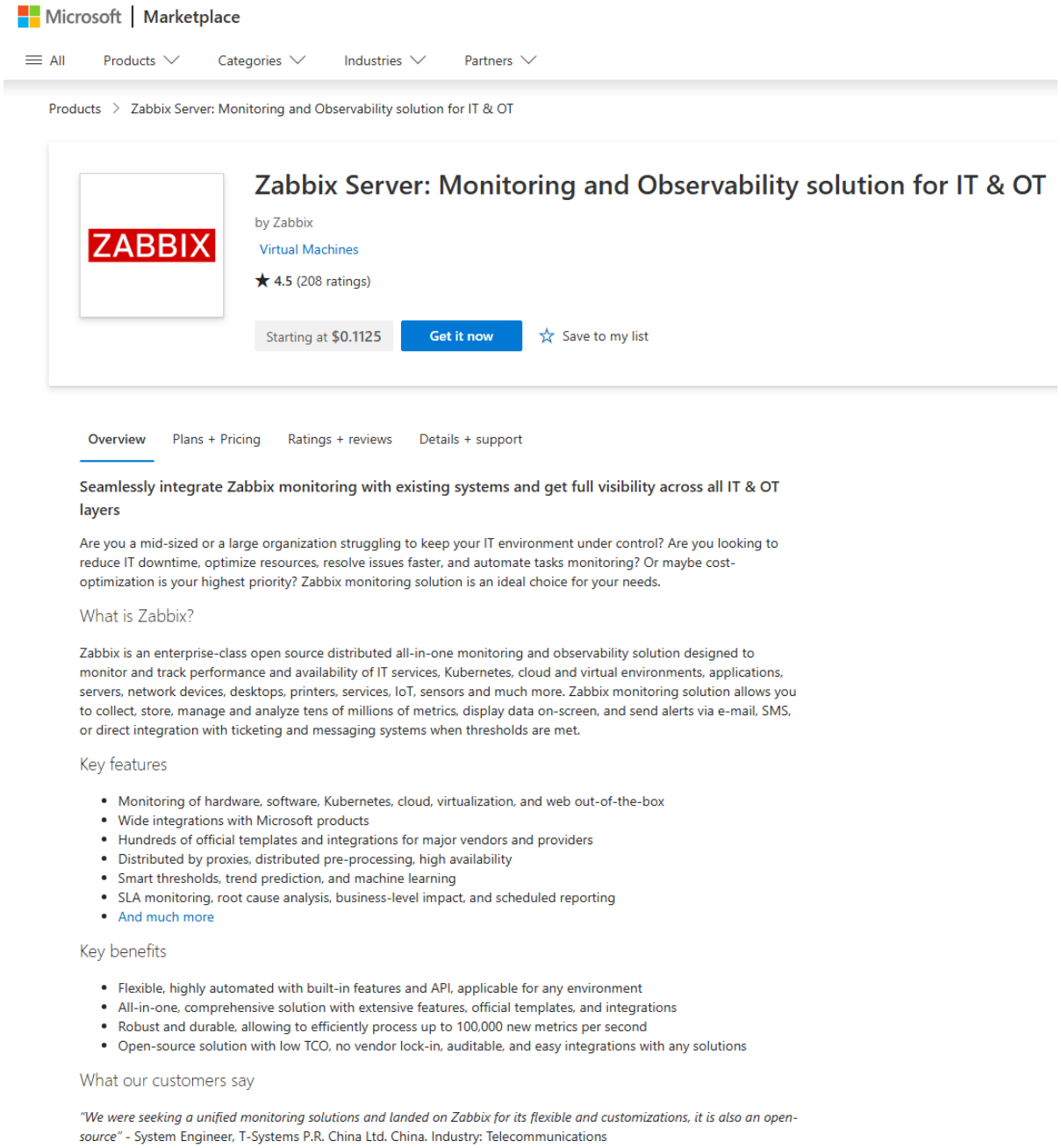
Descripción general

Esta página describe cómo implementar Zabbix server mediante Azure.

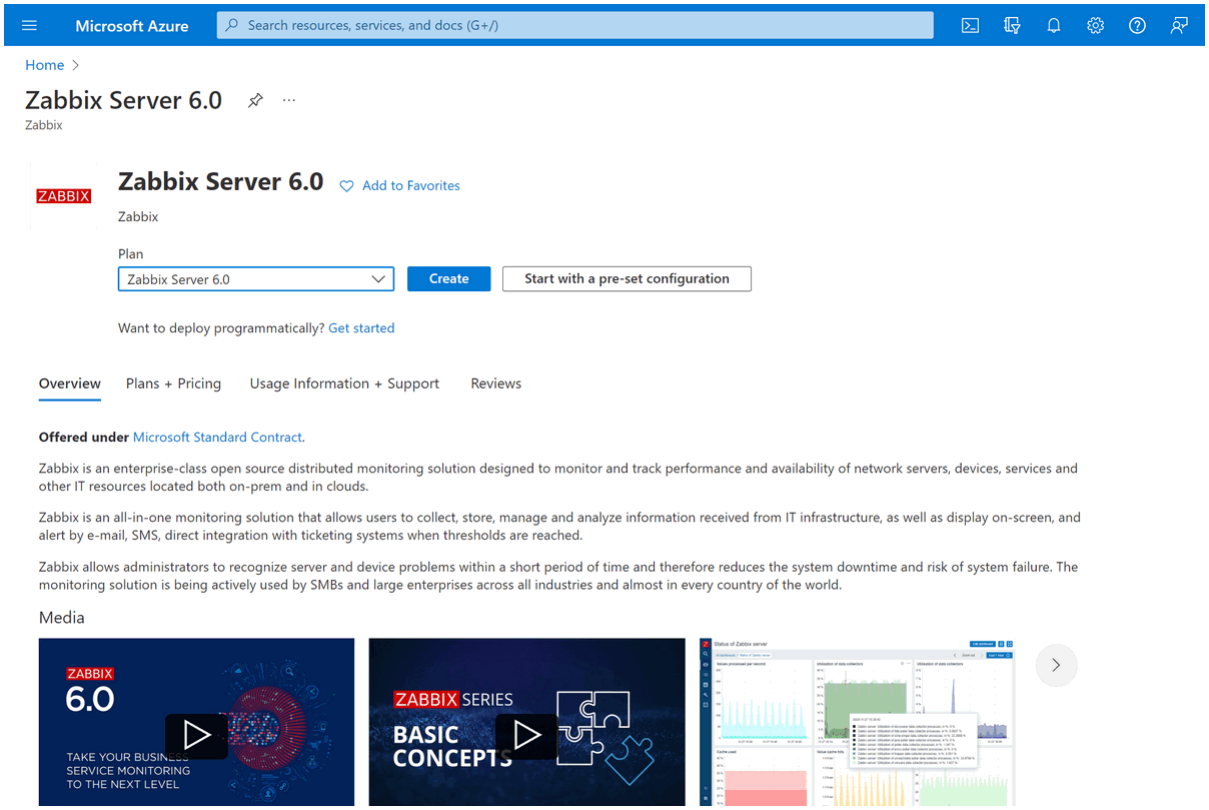
> **Attention:**
> Hay varias versiones de Zabbix disponibles para su implementación en la plataforma. Asegúrese de leer la documentación correspondiente a la misma versión de Zabbix que ha implementado. La documentación de las distintas versiones de Zabbix se puede seleccionar en el menú desplegable de versiones.

Implementación del appliance

Para implementar el appliance de Zabbix, primero vaya a la página de Zabbix en Azure Marketplace. Revise la descripción general y, a continuación, haga clic en *Get it now*.



Revise la descripción general de la implementación, los planes de precios y otros detalles. A continuación, haga clic en *Create* o en *Start with a pre-set configuration* para implementar su appliance de Zabbix.

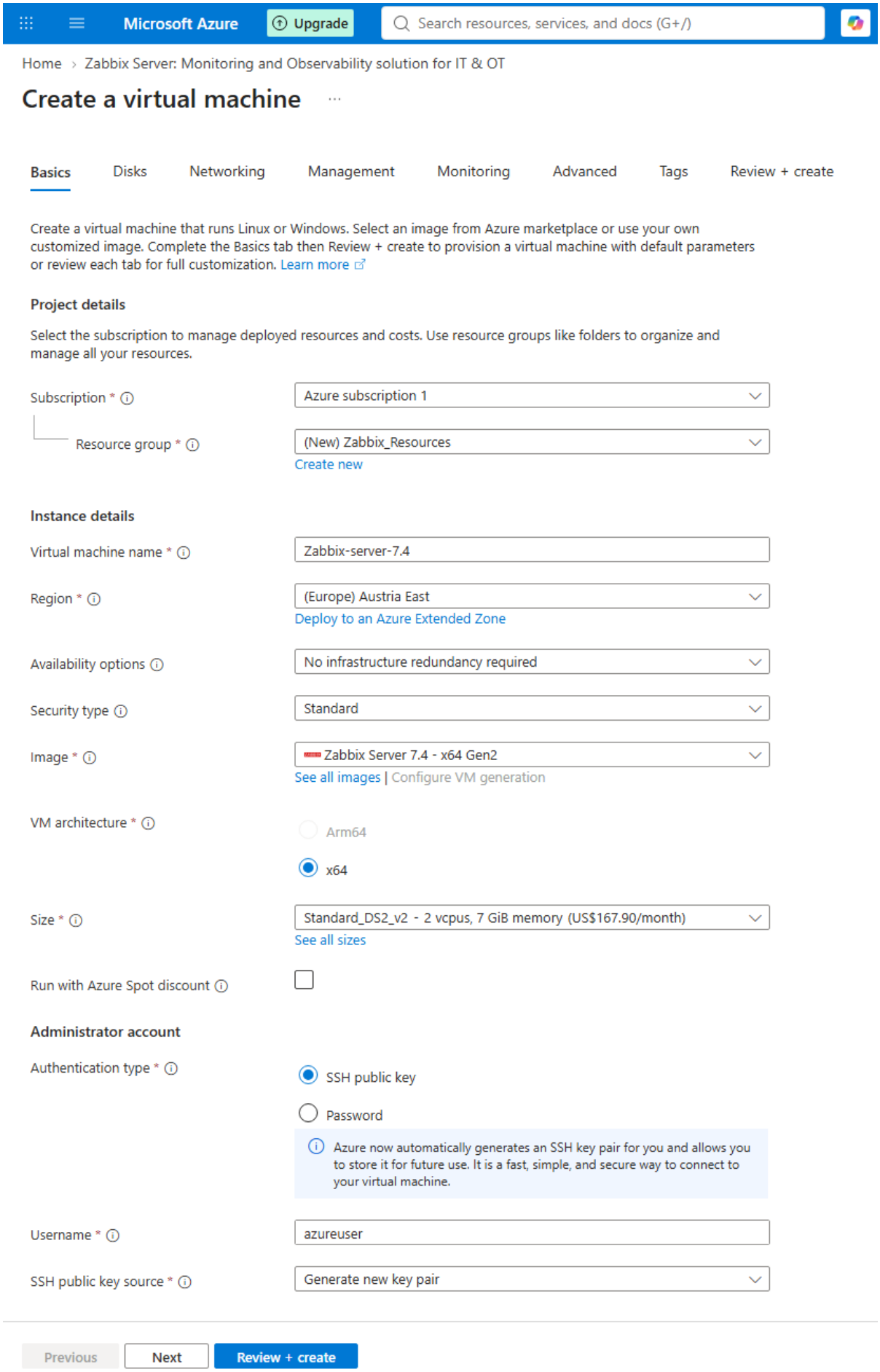


Configure la VM seleccionando el tamaño de la VM, creando una cuenta de administrador y definiendo la configuración del disco, la red, la administración y otros parámetros. Esta configuración puede variar según los requisitos de su infraestructura.

Puede utilizar la siguiente configuración como ejemplo:

Pestaña *Basics*:

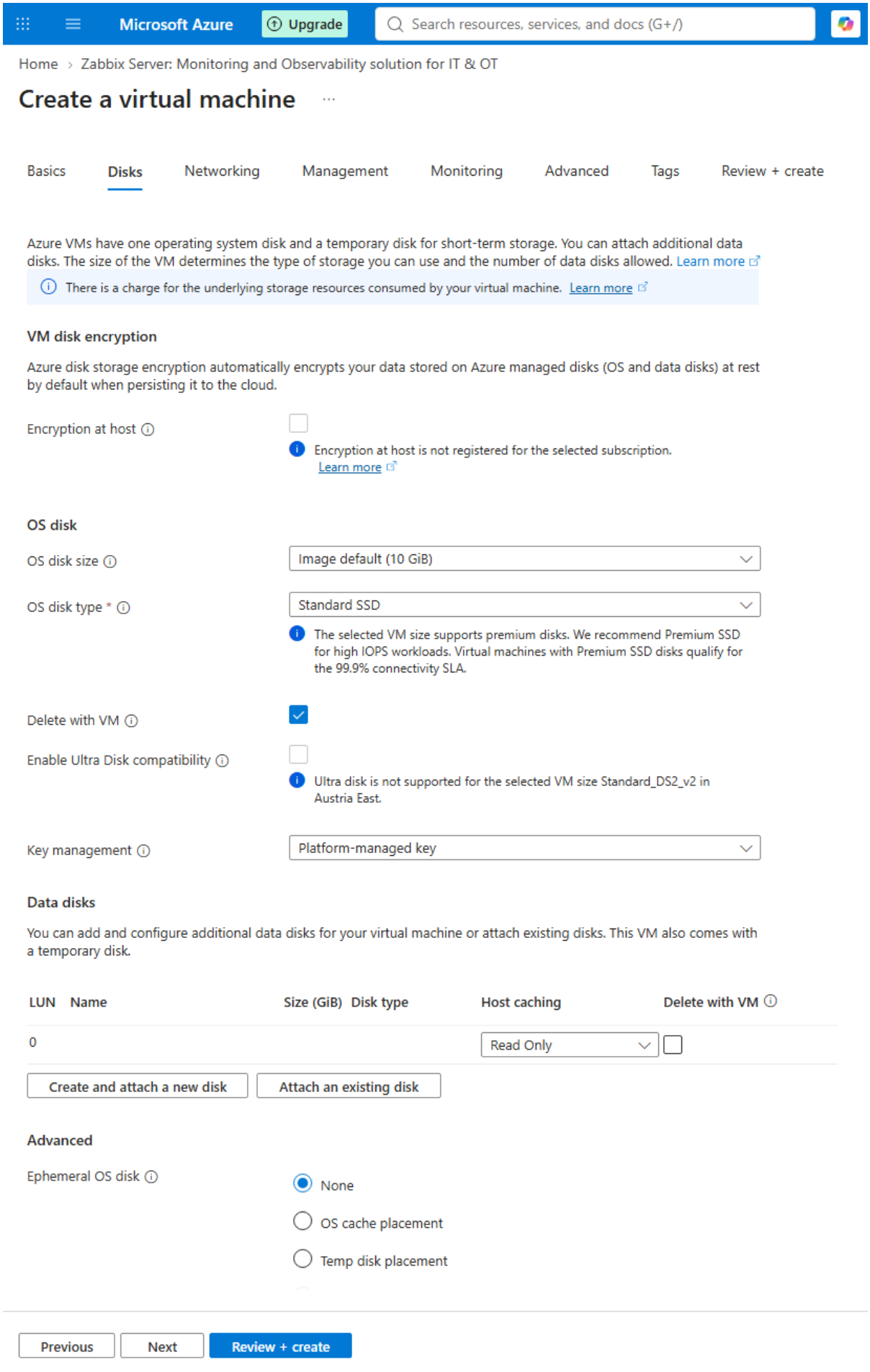
- *Subscription*: Azure subscription 1
- *Resource group*: Zabbix_Resources
- *Virtual machine name*: Zabbix-server--7.4
- *Availability options*: No infrastructure redundancy required
- *Security type*: Standard
- *Image*: Zabbix Server 7.4 - x64 Gen2
- *Size*: Standard_DS2_V2 - 2 vcpus, 7 GiB memory
- *Authentication type*: SSH public key
- *Username*: azureuser
- *SSH public key source*: Generate new key pair



Pestaña *Disks*:

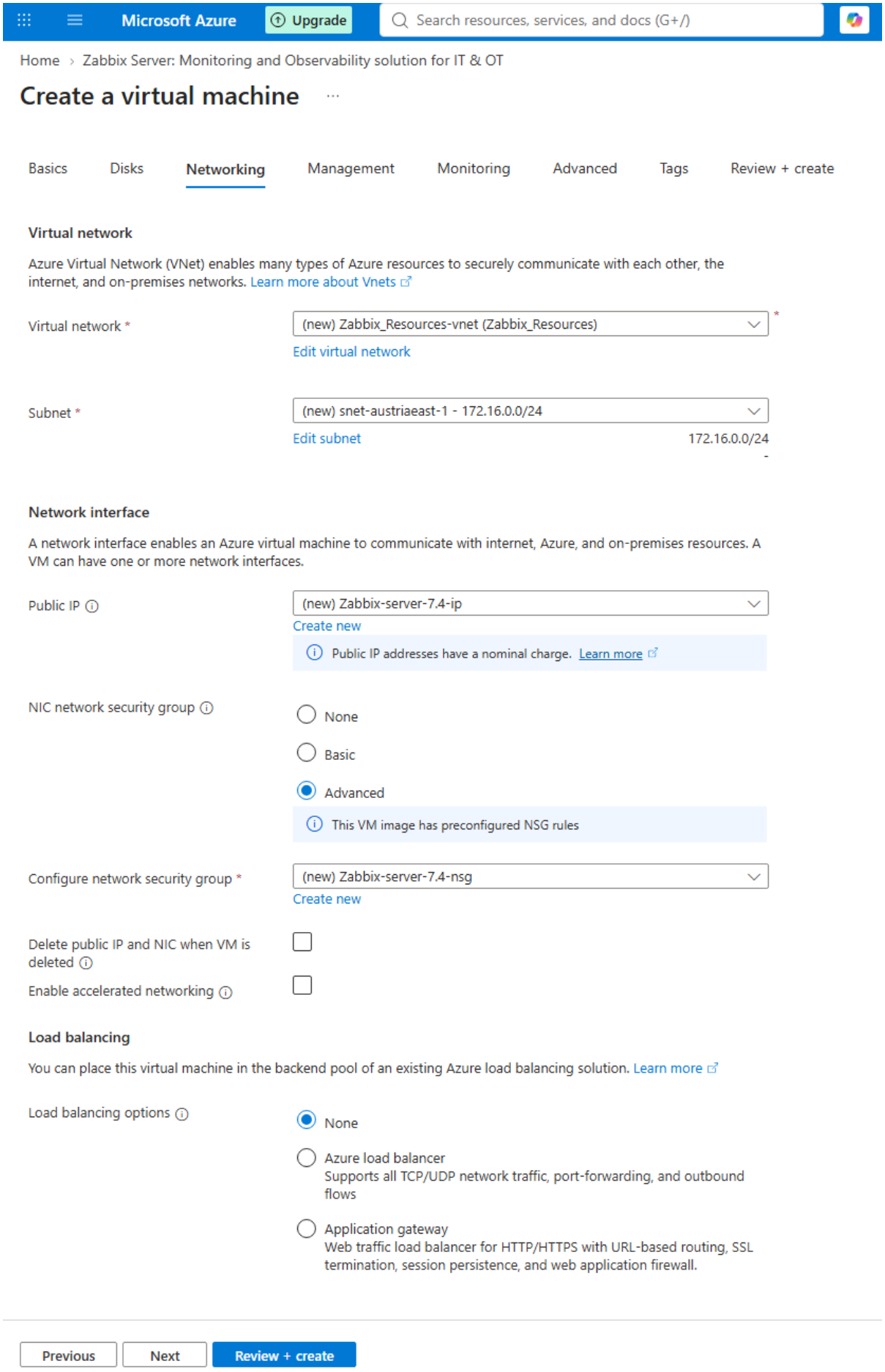
- *Encryption at host*: unchecked
- *OS disk type*: Standard SSD

- *Delete with VM*: checked
- *Enable Ultra Disk compatibility*: unchecked

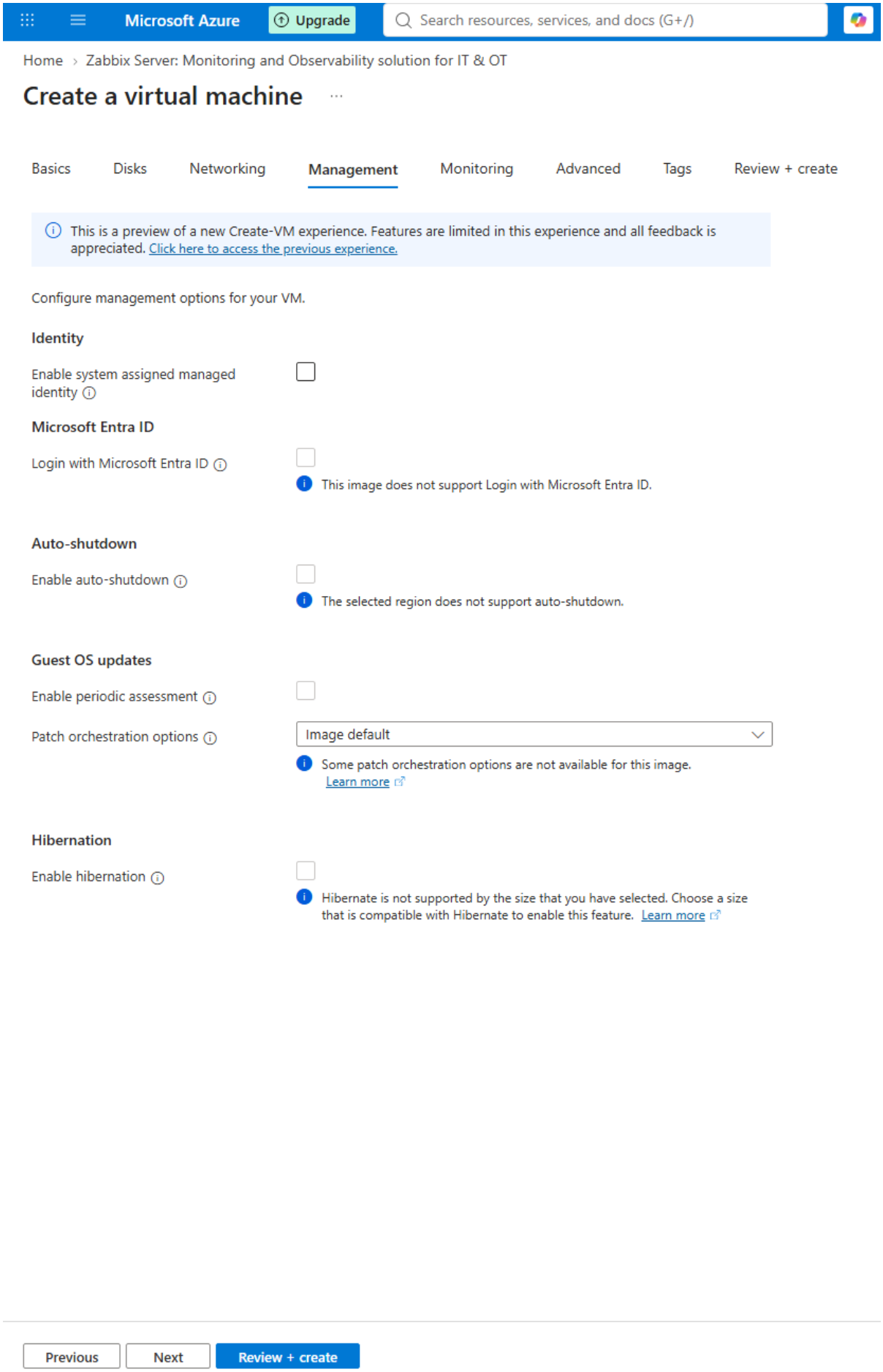


Pestaña *Networking*:

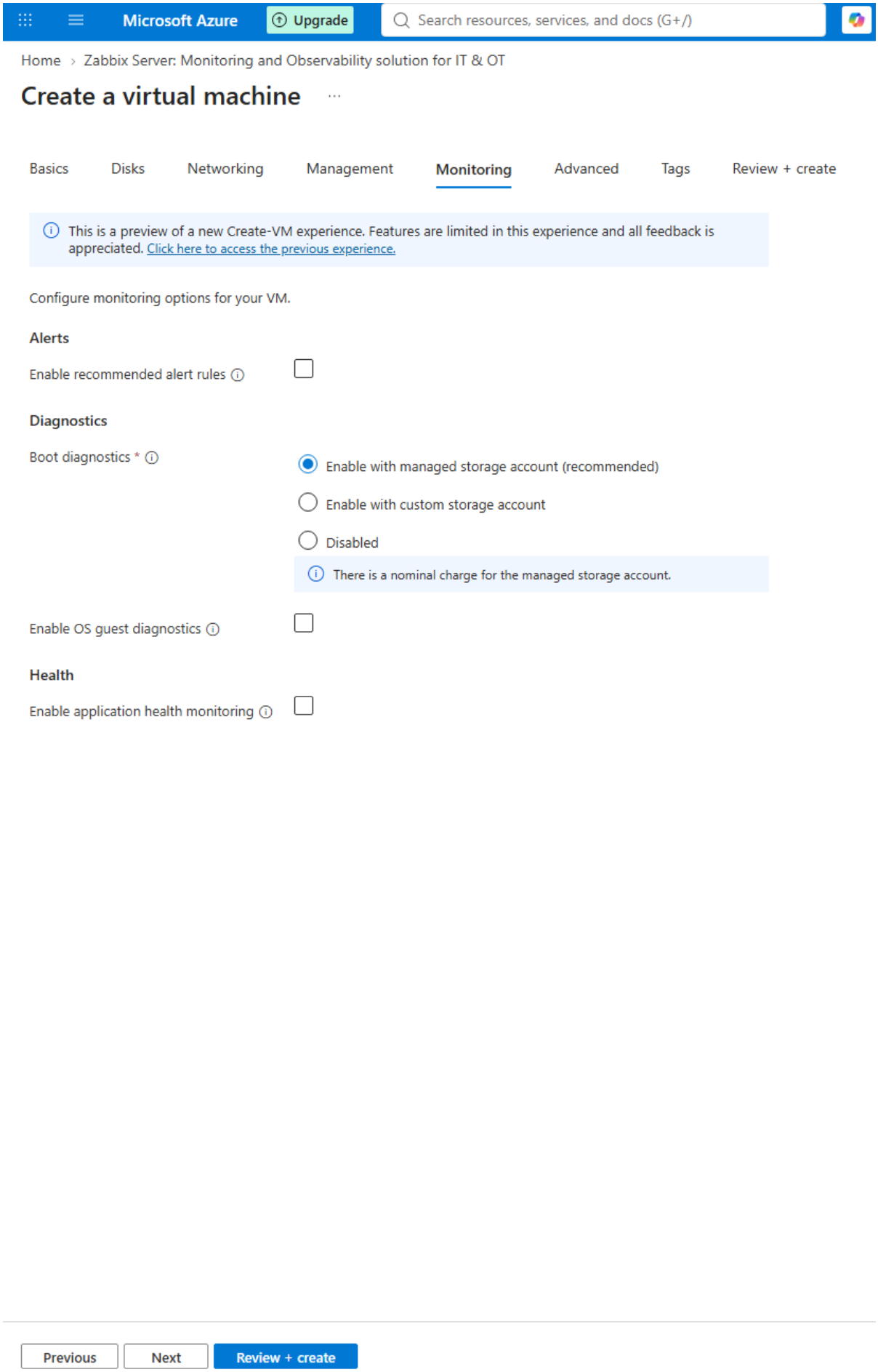
- *Virtual network*: Zabbix_Resources-vnet
- *Public IP*: Zabbix-server-7.4-ip
- *NIC network security group*: Advanced
- *Configure network security group*: Zabbix-server-7.4-nsg



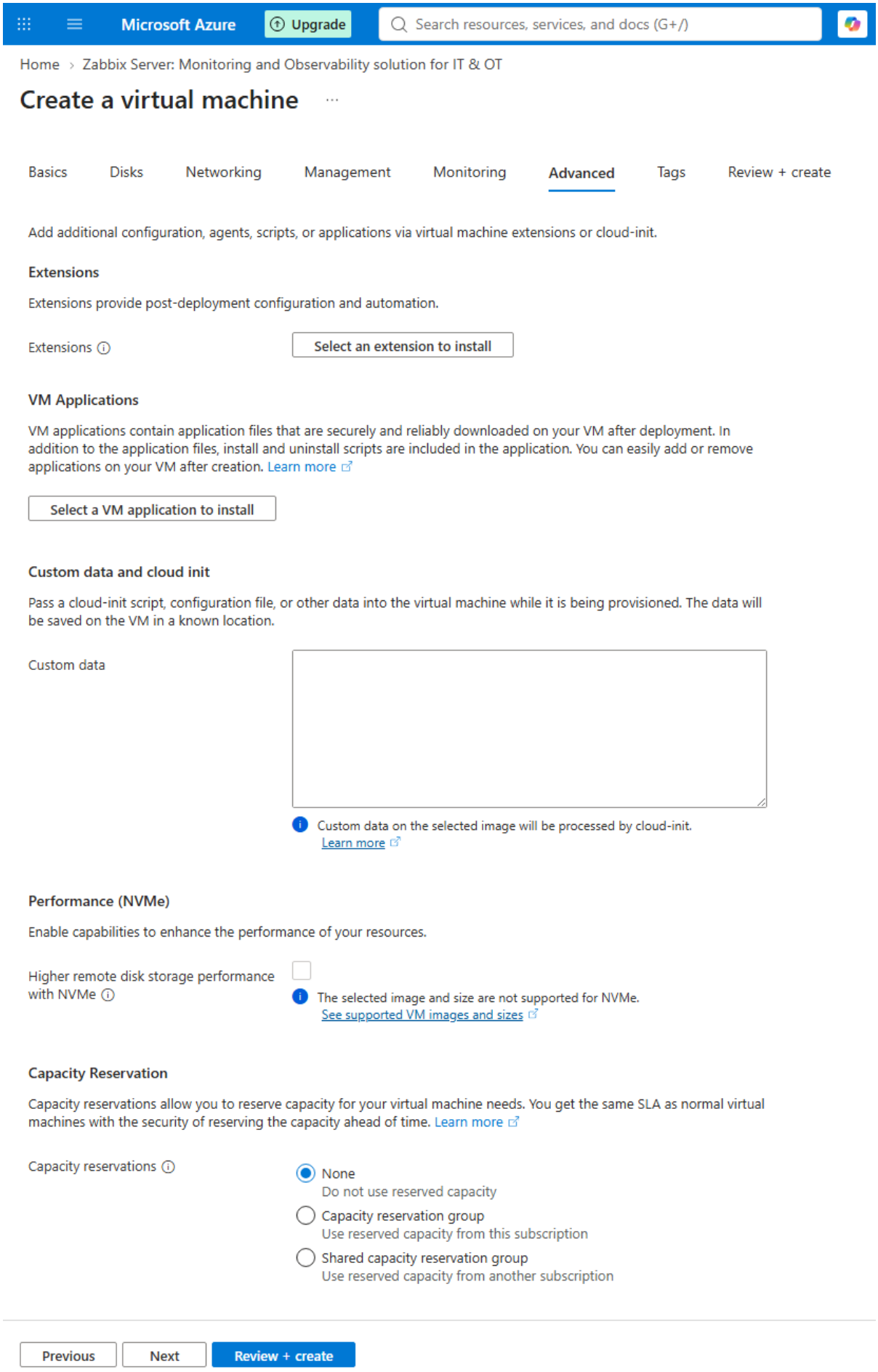
Pestaña *Management*: los parámetros pueden dejarse con sus valores predeterminados.



Pestaña *Monitoring*: los parámetros pueden dejarse con sus valores predeterminados.



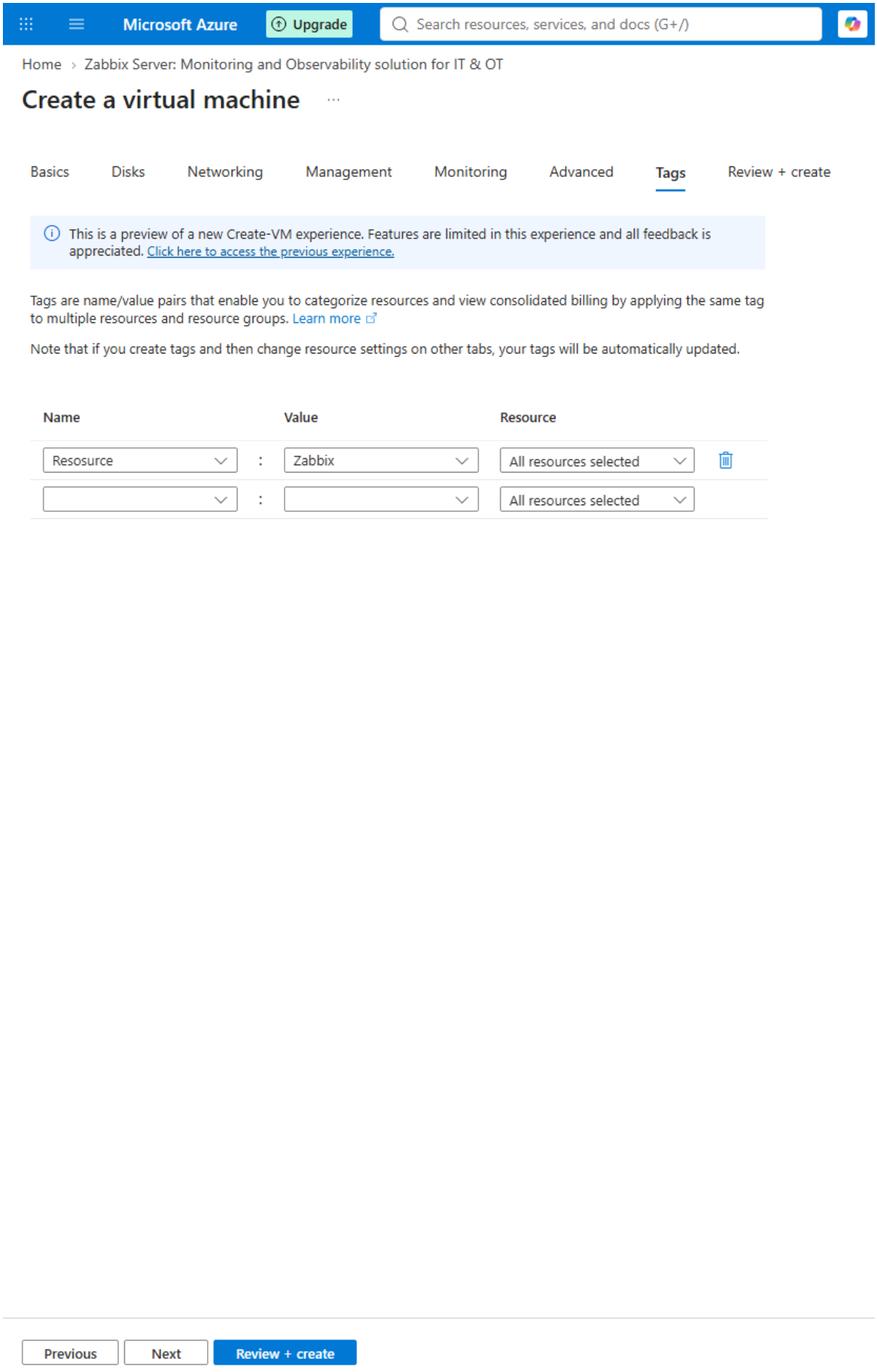
Pestaña *Advanced*: los parámetros pueden dejarse con sus valores predeterminados.



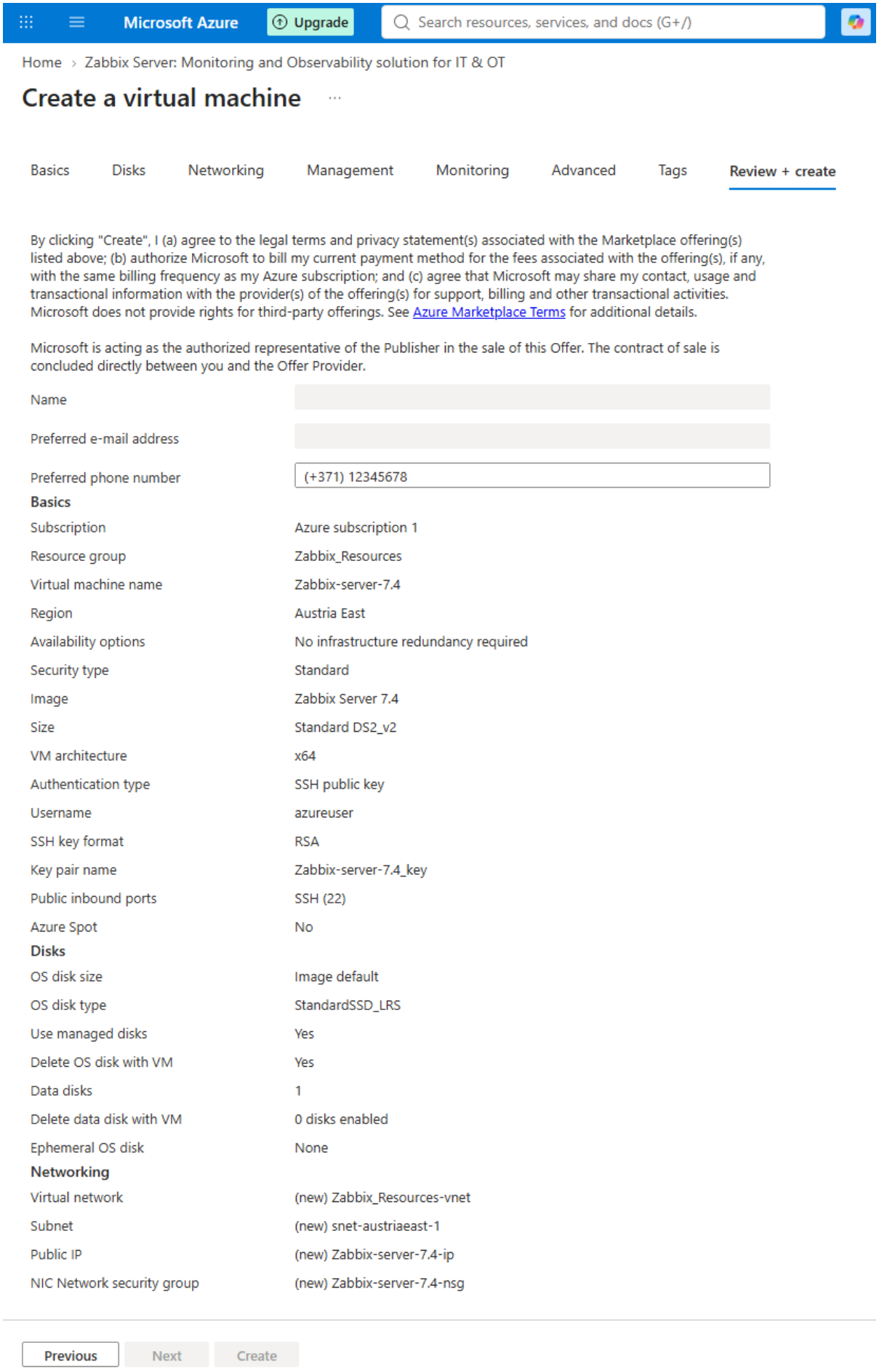
Pestaña *Tags*:

- *Name*: Resource
- *Value*: Zabbix

- *Resource*: All resource selected



Confirme los detalles de la VM y cree la VM.



Se le redirigirá a la página de estado de implementación de la VM. Espere hasta que se complete la implementación.

Una vez completada la implementación, abra el recurso y anote su dirección IP pública. Ahora deberá conectarse al Zabbix server mediante SSH (u otro método disponible) para obtener las credenciales de inicio de sesión del Zabbix frontend.

```
********************************************************************************

Zabbix frontend credentials:

Username: Admin

Password: zfpvagcs0s+srUUK+EM

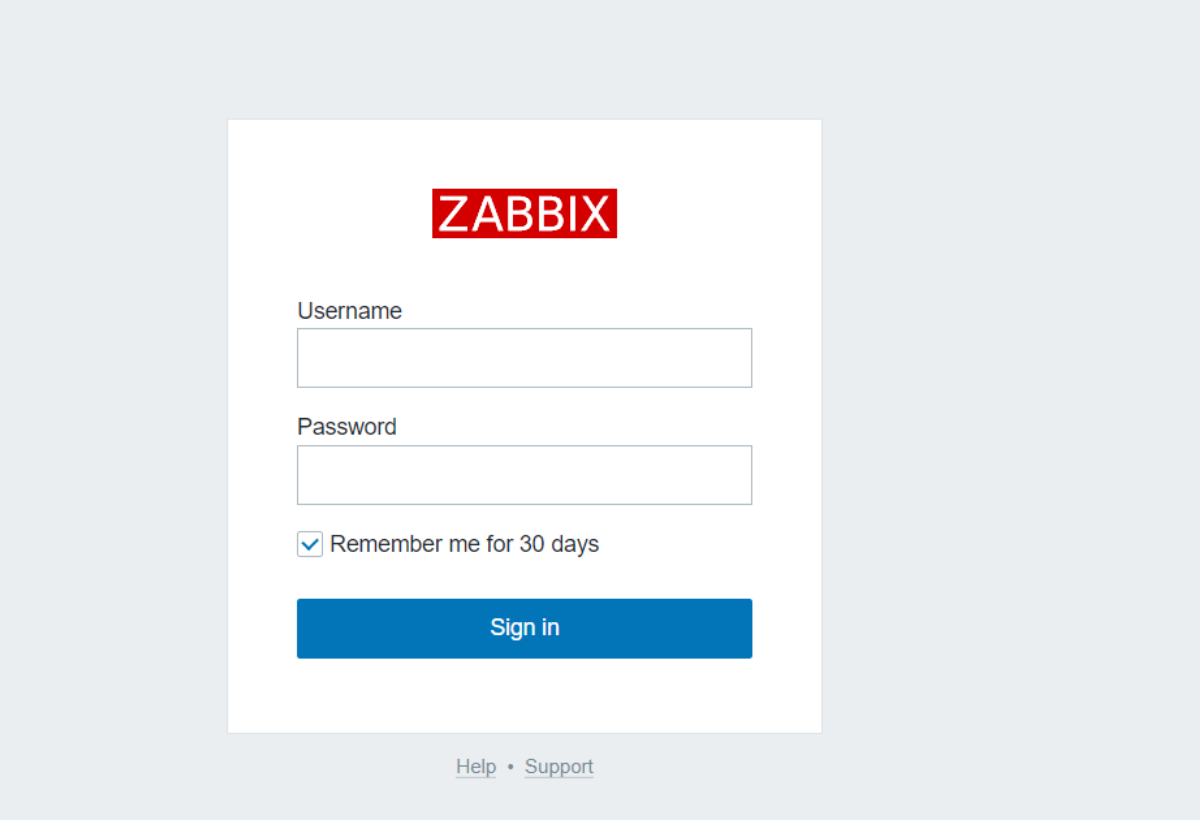

To learn about available professional services, including technical suppport and
 training, please visit https://www.zabbix.com/services

Official Zabbix documentation available at https://www.zabbix.com/documentation/
current/

Note! Do not forget to change timezone PHP variable in /etc/php-fpm.d/zabbix.con
f file.


********************************************************************************
[azureuser@Zabbix-server-6 ~]$
```

Acceda al Zabbix frontend abriendo la dirección IP pública del appliance de Zabbix en un navegador web. Para iniciar sesión, utilice el nombre de usuario y la contraseña obtenidos en el paso anterior.



Acceso a la base de datos

Puede acceder a la base de datos de Zabbix utilizando el usuario `root` sin introducir una contraseña. Las credenciales se almacenan en el archivo de configuración `/root/.my.cnf`.

Guía de implementación de Azure para el proxy de Zabbix

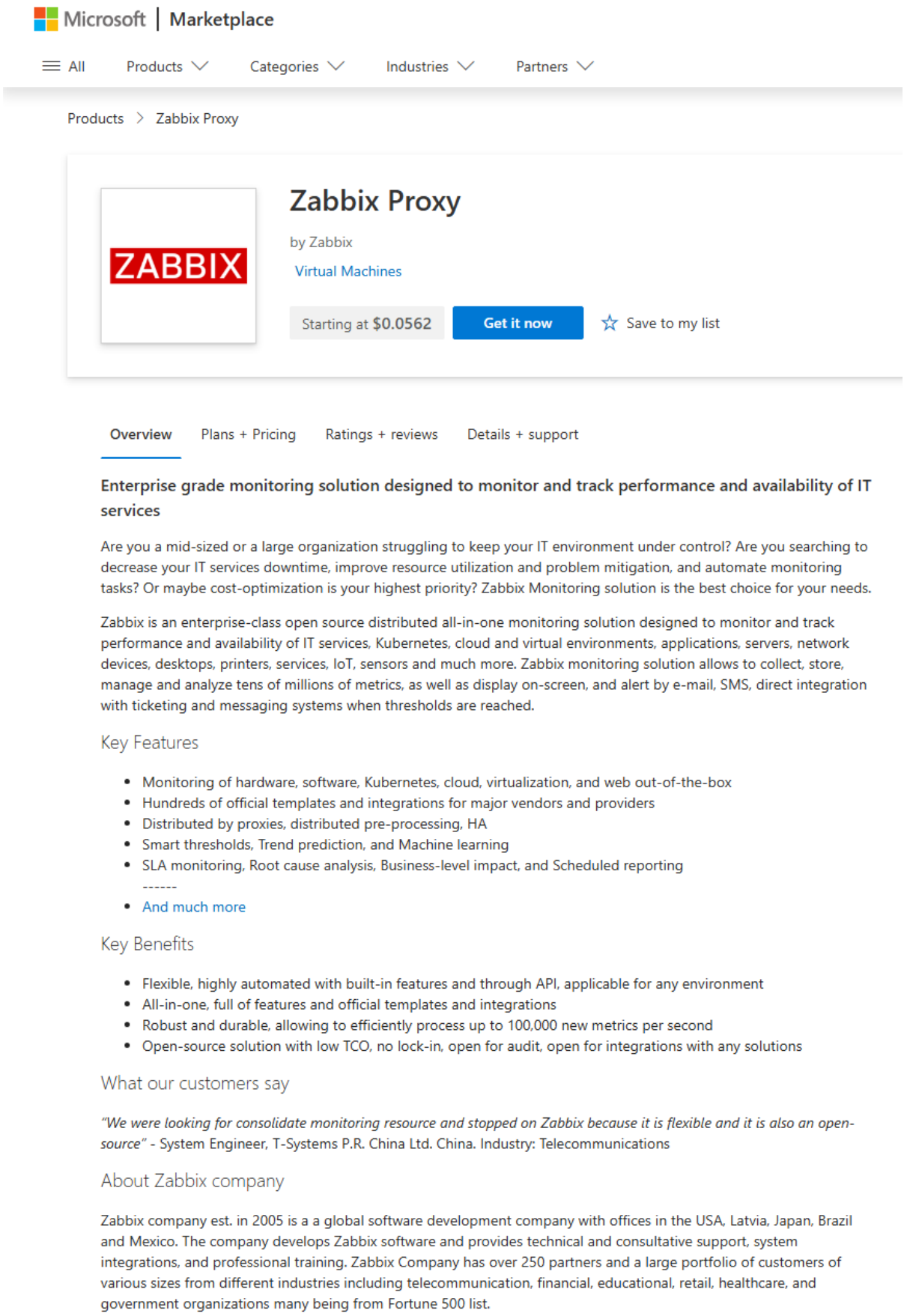
Descripción general

Esta página describe cómo implementar un proxy de Zabbix mediante Azure.

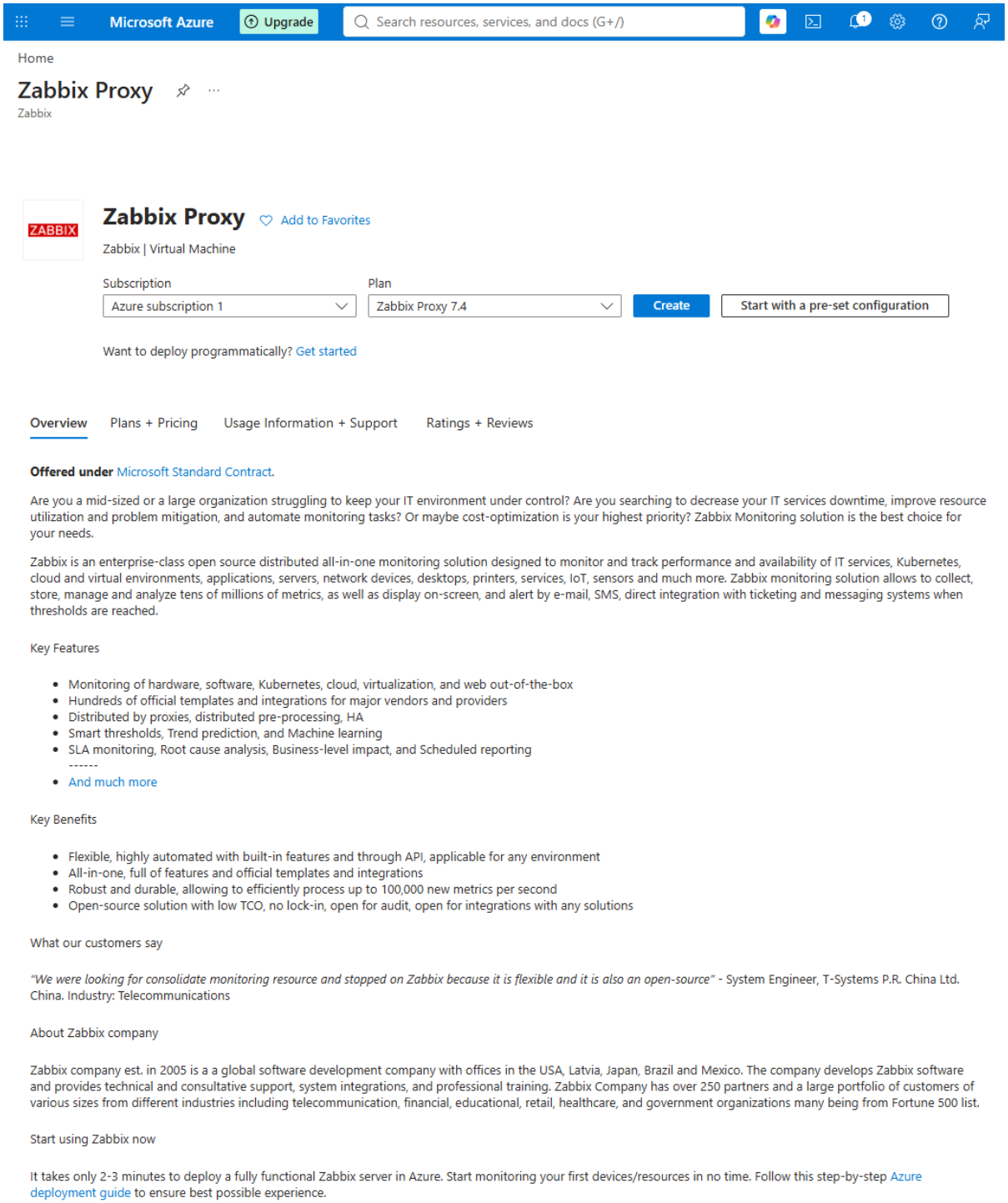
Attention:
Hay varias versiones de Zabbix disponibles para su implementación en la plataforma. Asegúrese de leer la documentación correspondiente a la misma versión de Zabbix que ha implementado. Puede seleccionar la documentación de diferentes versiones de Zabbix en el menú desplegable de versiones.

Implementación de Zabbix proxy

Para implementar Zabbix proxy, primero vaya a la página de Zabbix en Azure Marketplace. Haga clic en *Get it now*.



Aquí puede consultar información adicional sobre la implementación actual, como la descripción general, los planes de precios y mucho más. Haga clic en *Create* o en *Start with a pre-set configuration*.

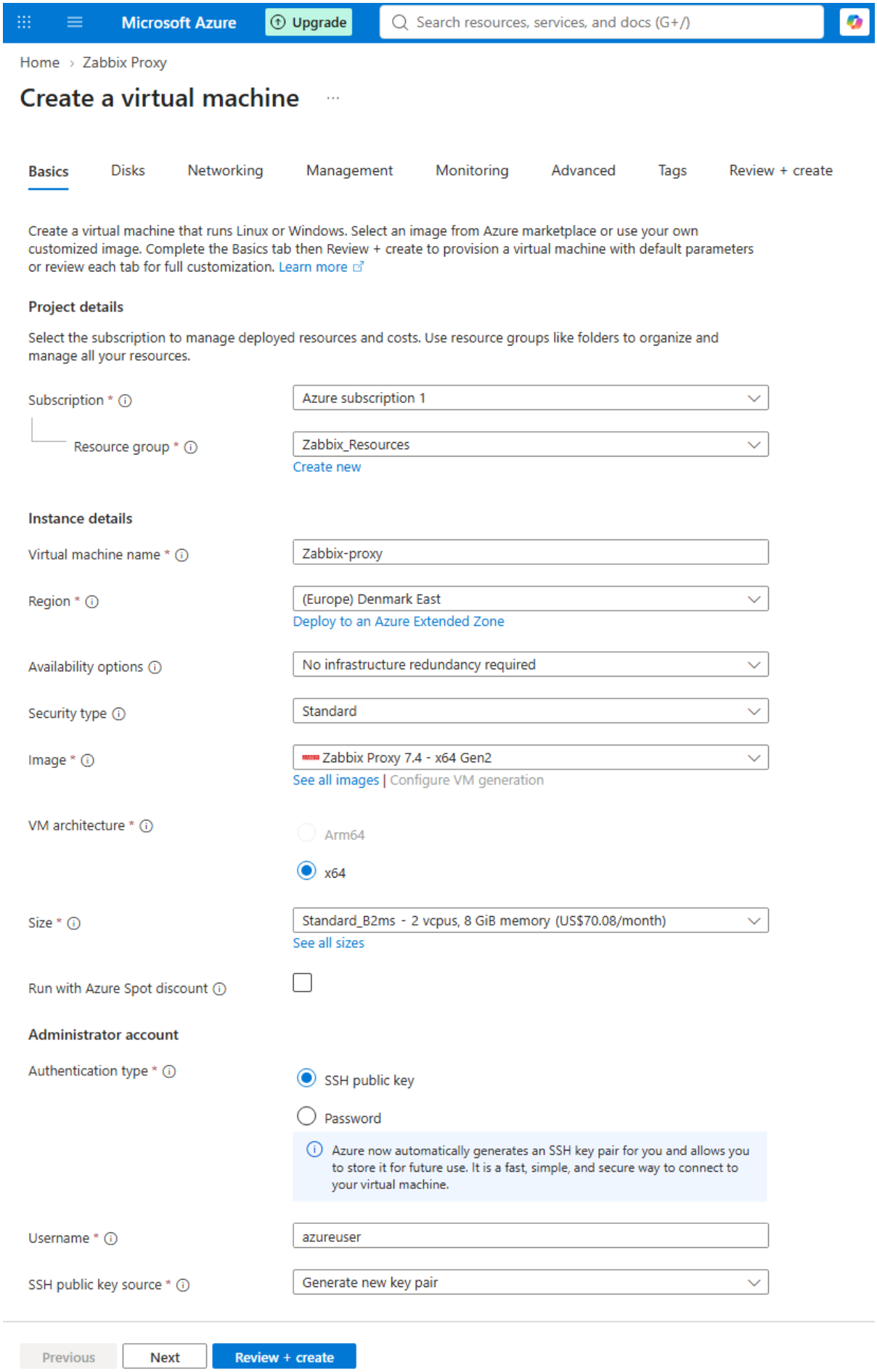


Continúe con la configuración detallada de la VM: seleccione el tamaño de la VM, la cuenta de administrador, el disco, la red, la administración y otros ajustes. Estos ajustes pueden variar según los requisitos de su infraestructura.

Puede utilizar la siguiente configuración como ejemplo:

Pestaña *Basics*:

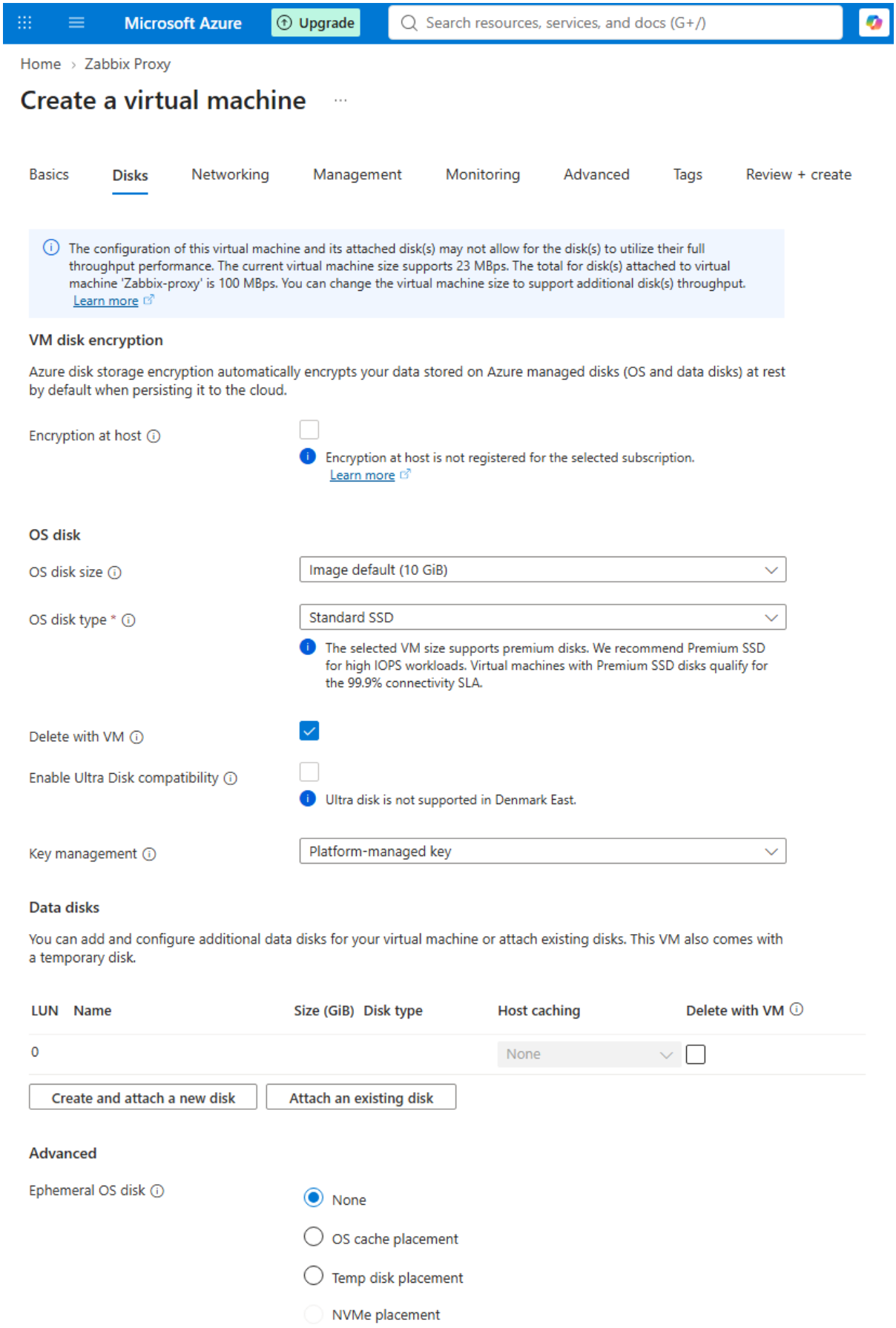
- *Subscription*: Azure subscription 1
- *Resource group*: Zabbix_Resources
- *Virtual machine name*: Zabbix-proxy
- *Availability options*: No infrastructure redundancy required
- *Security type*: Standard
- *Image*: Zabbix Proxy 7.4 - x64 Gen2
- *Size*: Standard_B2ms - 2 vcpus, 8 GiB memory
- *Authentication type*: SSH public key
- *Username*: azureuser
- *SSH public key source*: Generate new key pair



Pestaña *Disks*:

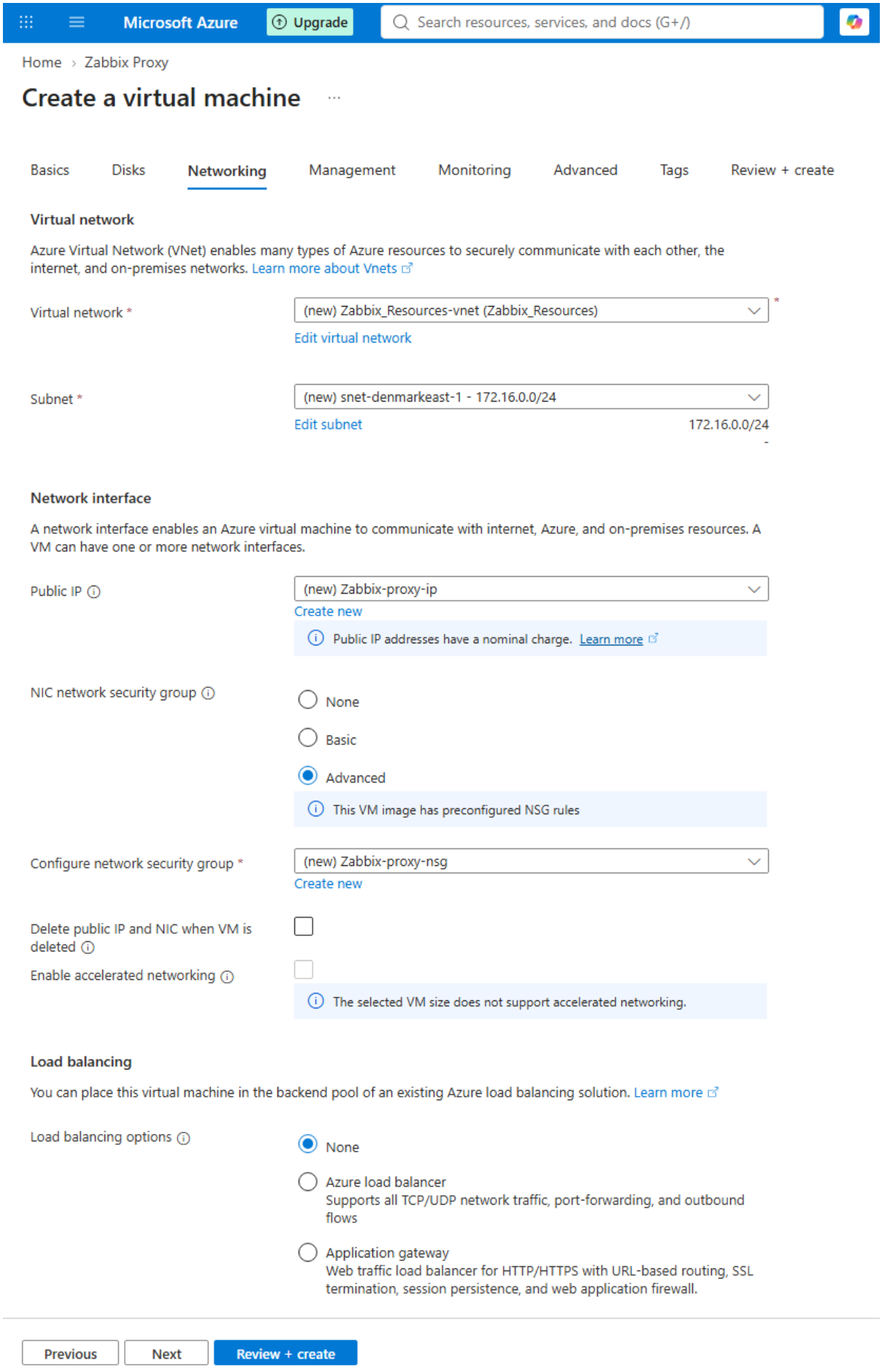
- *Encryption at host*: unchecked

- *OS disk type*: Standard SSD
- *Delete with VM*: checked
- *Enable Ultra Disk compatibility*: unchecked

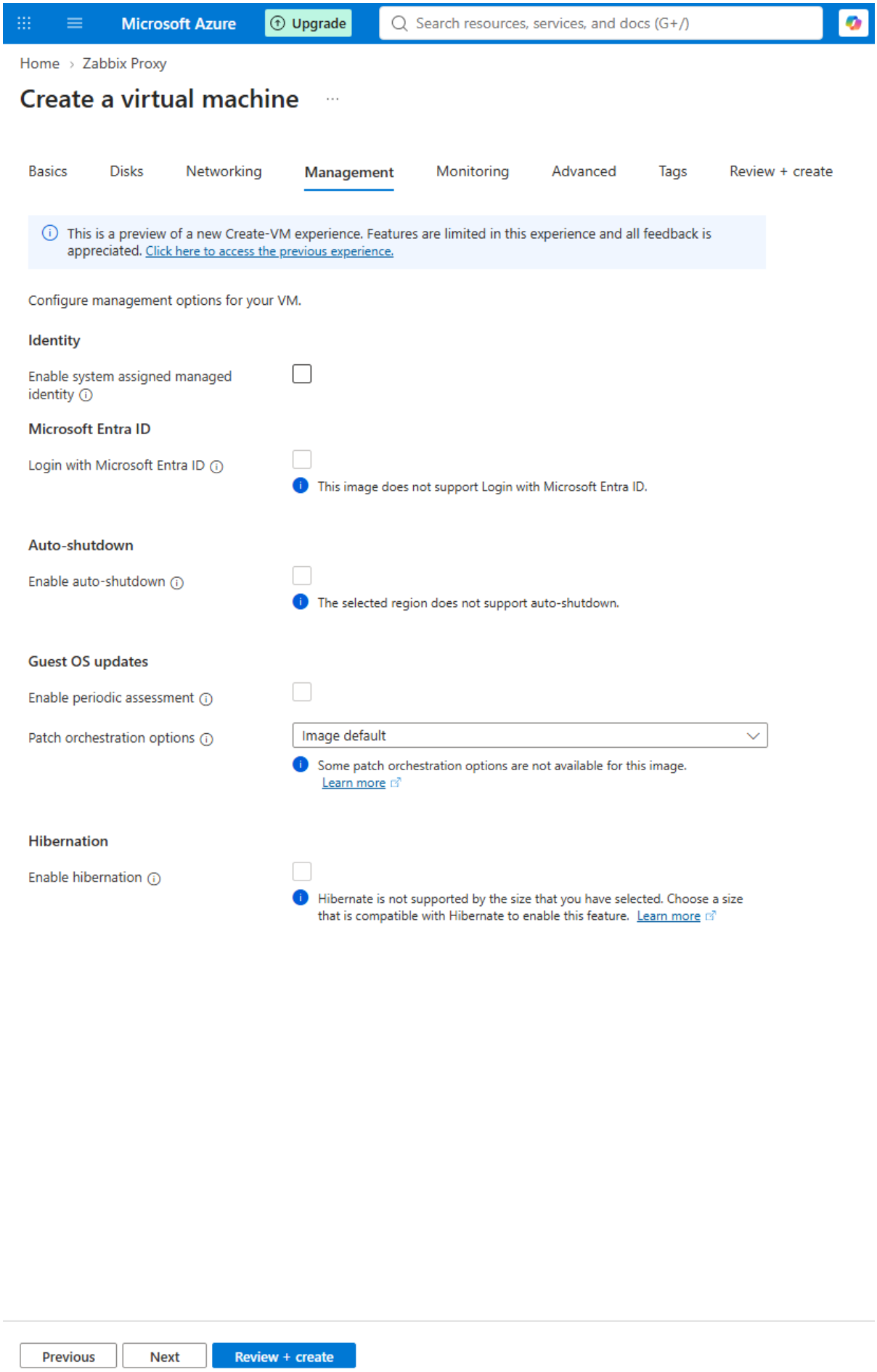


Pestaña *Networking*:

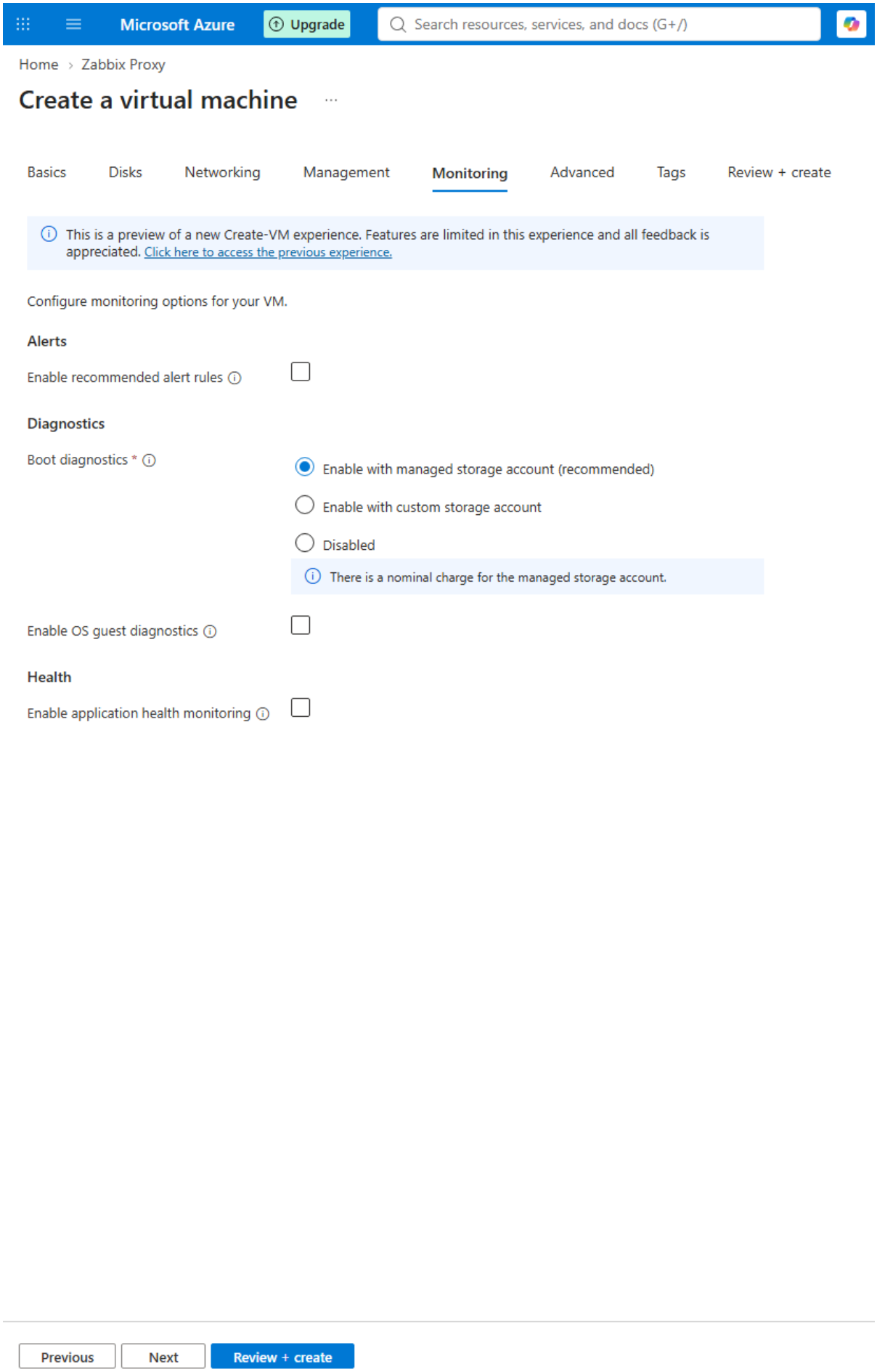
- *Virtual network*: Zabbix_Resources-vnet
- *Public IP*: Zabbix-proxy-ip
- *NIC network security group*: Advanced
- *Configure network security group*: Zabbix-proxy-nsg



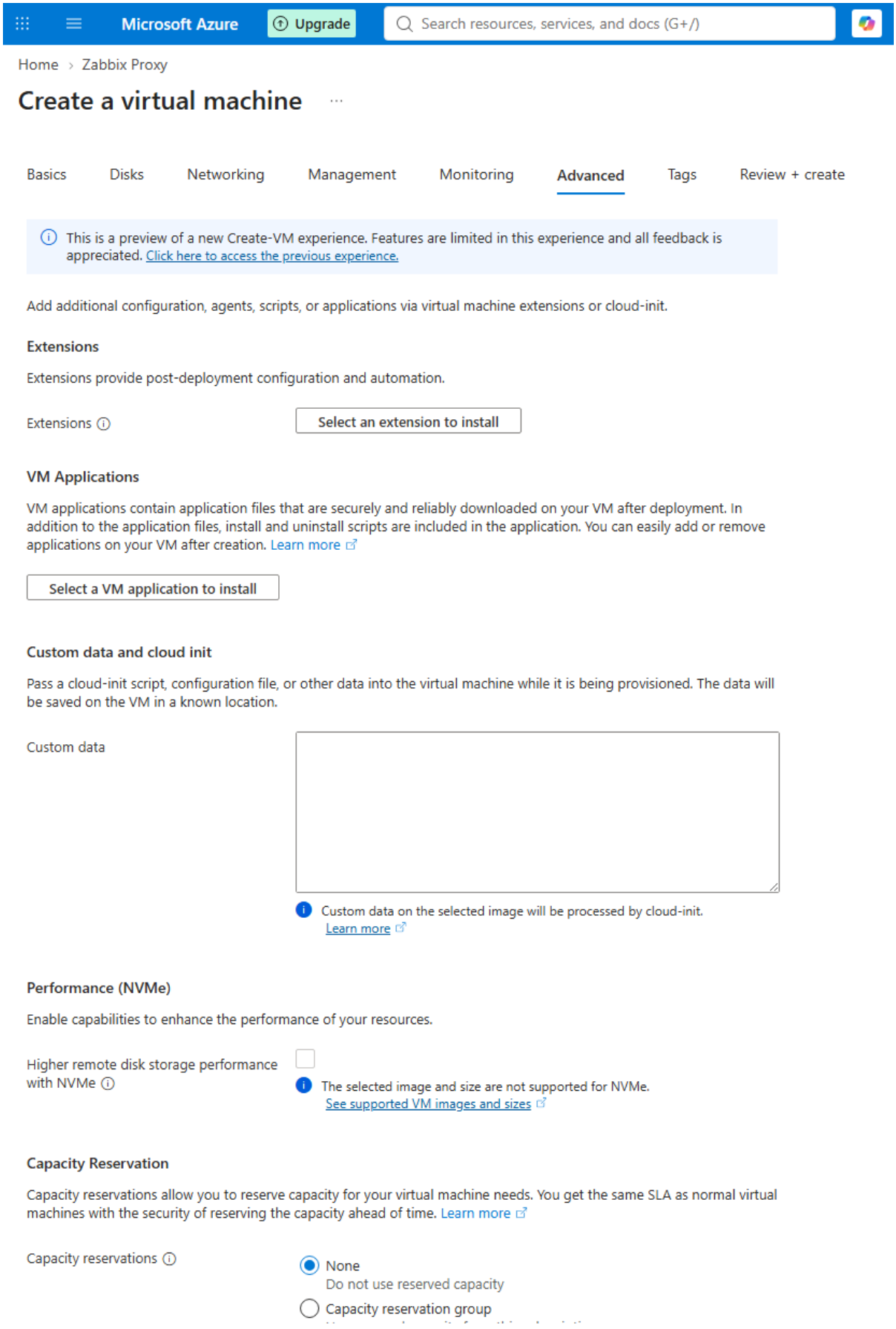
Pestaña *Management*: los parámetros pueden dejarse con sus valores predeterminados.



Pestaña *Monitoring*: los parámetros pueden dejarse con sus valores predeterminados.



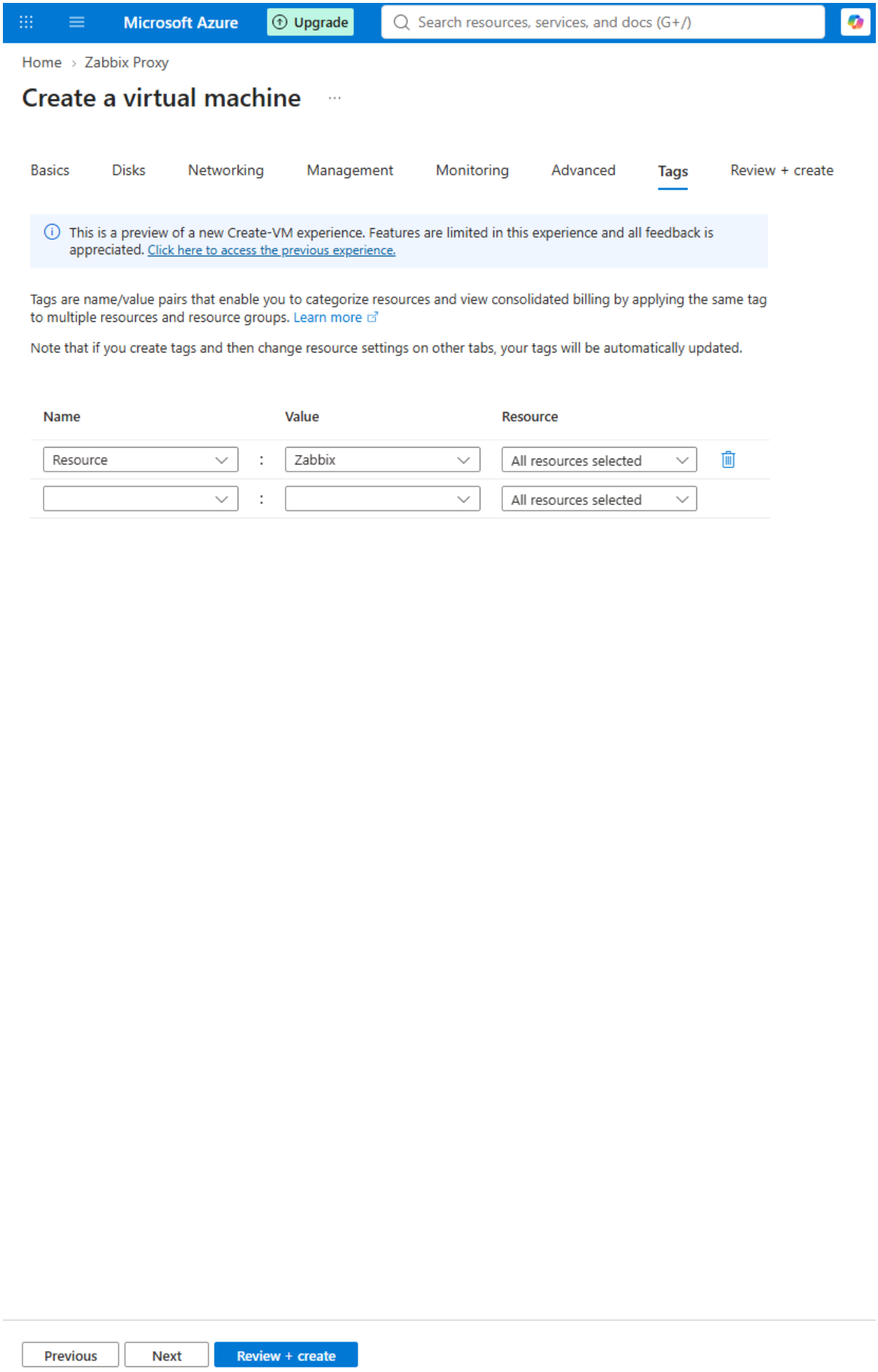
Pestaña *Advanced*: los parámetros pueden dejarse con sus valores predeterminados.



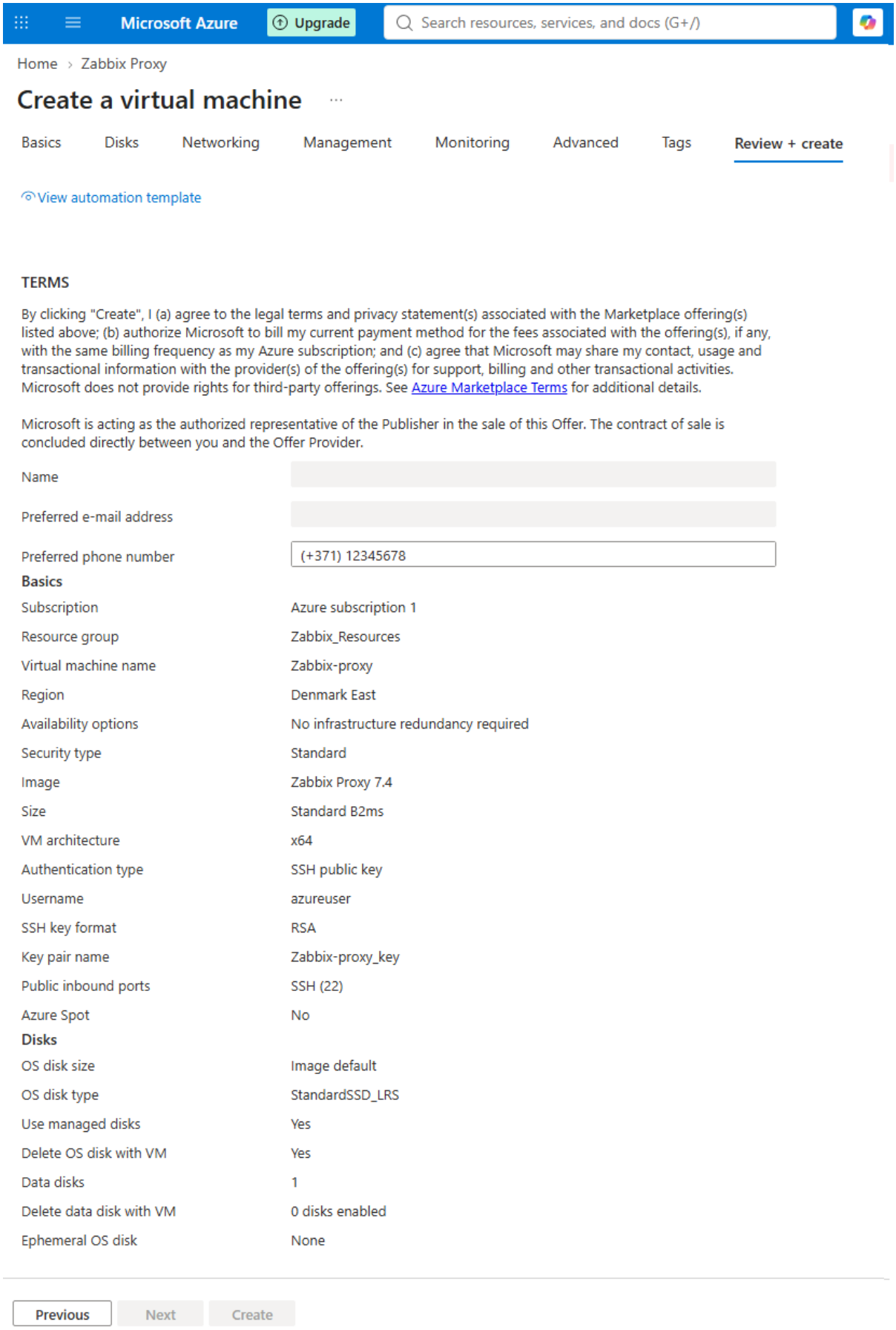
Pestaña *Tags*:

- *Name*: Resource
- *Value*: Zabbix

- *Resource*: All resource selected



Confirme los detalles de la VM y cree la VM.



Se le redirigirá a la página de estado de implementación de la VM. Espere hasta que finalice la implementación.

Una vez finalizada la implementación, abra el recurso y anote su dirección IP pública. Conéctese a Zabbix proxy mediante SSH y las credenciales de administrador creadas durante la implementación de la VM para configurar el proxy.

```
********************************************************************************

To learn about available professional services, including technical suppport and
 training, please visit https://www.zabbix.com/services

Official Zabbix documentation available at https://www.zabbix.com/documentation/
current/

********************************************************************************
[azureuser@Zabbix-proxy ~]$
```

El archivo de configuración de Zabbix proxy se encuentra en `/etc/zabbix/zabbix_proxy.conf`.

La configuración inicial del proxy consiste en modificar los parámetros `ProxyMode`, `Server` y `Hostname` en el archivo de configuración de Zabbix proxy.

El modo del proxy se puede cambiar editando el parámetro `ProxyMode`. Establezca `ProxyMode=0` para el modo activo o `ProxyMode=1` para el modo pasivo.

El parámetro `Server` debe contener la dirección IP del Zabbix server.

El parámetro `Hostname` debe contener el nombre de host de Zabbix proxy. El nombre de host definido aquí debe coincidir con el nombre del proxy proporcionado durante la configuración del proxy en el frontend de Zabbix.

```
############ GENERAL PARAMETERS #################

### Option: ProxyMode
#       Proxy operating mode.
#       0 - proxy in the active mode
#       1 - proxy in the passive mode
#
# Mandatory: no
# Default:
# ProxyMode=0

### Option: Server
#       If ProxyMode is set to active mode:
#               IP address or DNS name (address:port) or cluster (address:port;address2:port) of Zabbix server to get
configuration data from and send data to.
#               If port is not specified, default port is used.
#               Cluster nodes need to be separated by semicolon.
#       If ProxyMode is set to passive mode:
#               List of comma delimited IP addresses, optionally in CIDR notation, or DNS names of Zabbix server.
#               Incoming connections will be accepted only from the addresses listed here.
#               If IPv6 support is enabled then '127.0.0.1', '::127.0.0.1', '::ffff:127.0.0.1' are treated equally
#               and '::/0' will allow any IPv4 or IPv6 address.
#               '0.0.0.0/0' can be used to allow any IPv4 address.
#               Example: Server=127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.example.com
#
# Mandatory: yes
# Default:
# Server=

Server=127.0.0.1
```

```
### Option: Hostname
#       Unique, case sensitive Proxy name. Make sure the Proxy name is known to the server!
#       Value is acquired from HostnameItem if undefined.
#
# Mandatory: no
# Default:
# Hostname=

Hostname=Zabbix proxy
```

Abra el frontend de Zabbix y vaya a *Administration* > *Proxies*. Haga clic en el botón *Create proxy*.

Introduzca el mismo nombre del proxy que el valor del parámetro *Hostname* en el archivo de configuración del proxy.

Seleccione el modo del proxy y proporcione la dirección IP del proxy (opcional para proxies activos).

Haga clic en el botón *Create proxy* para crear un proxy.

Ubicación de la base de datos del proxy

La base de datos SQLite del proxy se encuentra en `/tmp/zabbix_proxy.sqlite3` y se especifica mediante el parámetro `DBName` en el archivo de configuración del proxy de Zabbix.

Guía de implementación de Google Cloud

Descripción general

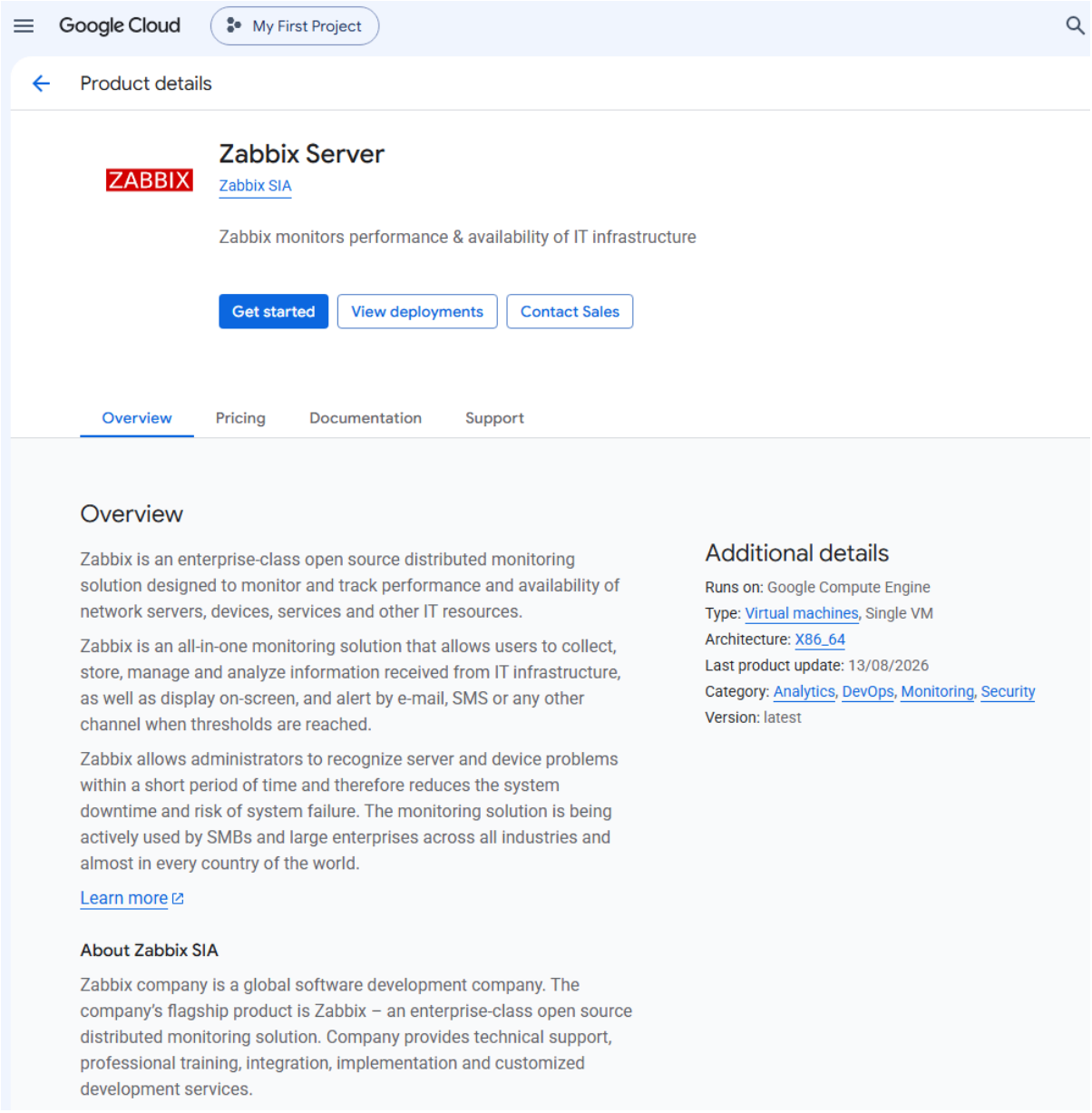
Esta página describe cómo implementar Zabbix mediante Google Cloud Marketplace.

Attention:
Hay varias versiones de Zabbix disponibles para su implementación en la plataforma. Asegúrese de leer la documentación correspondiente a la misma versión de Zabbix que ha implementado. Puede seleccionar la documentación de diferentes versiones de Zabbix en el menú desplegable de versiones.

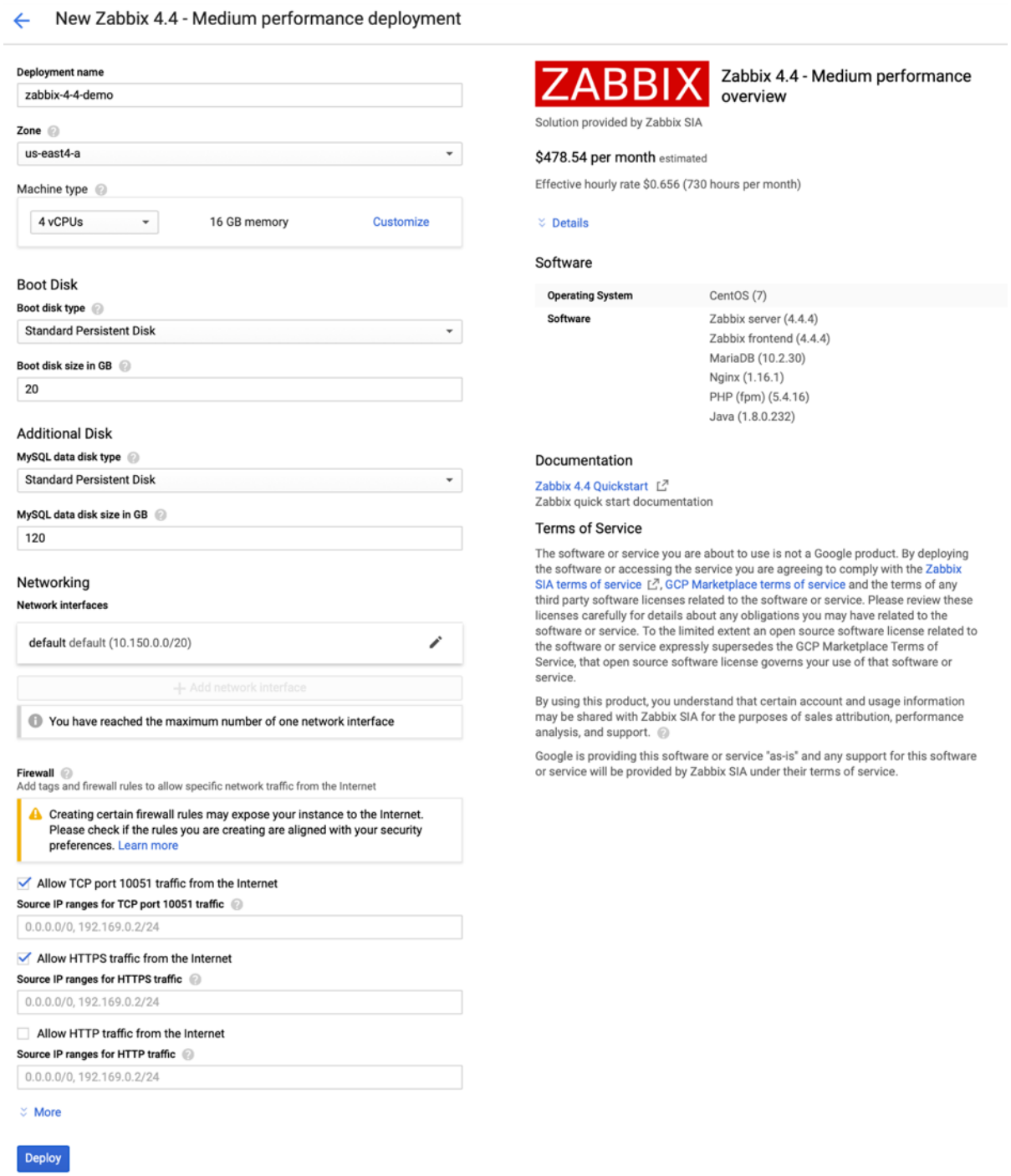
Implementación del appliance

Primero, implemente el appliance de Zabbix.
Es muy fácil hacerlo con Google Cloud Marketplace.
Simplemente vaya al producto Zabbix Server en el marketplace.

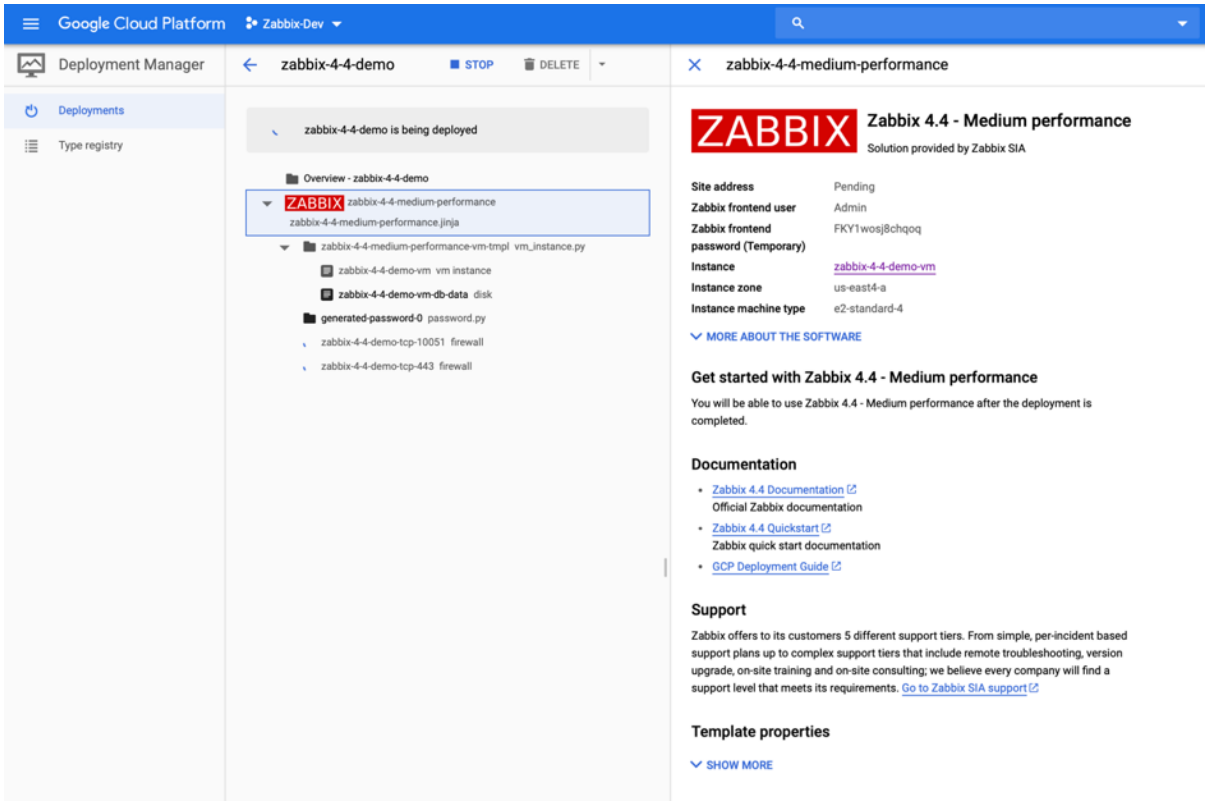
Haga clic en *Get started*.



Puede utilizar la configuración predeterminada o personalizarla.
Cuando haya terminado, haga clic en *Deploy*.



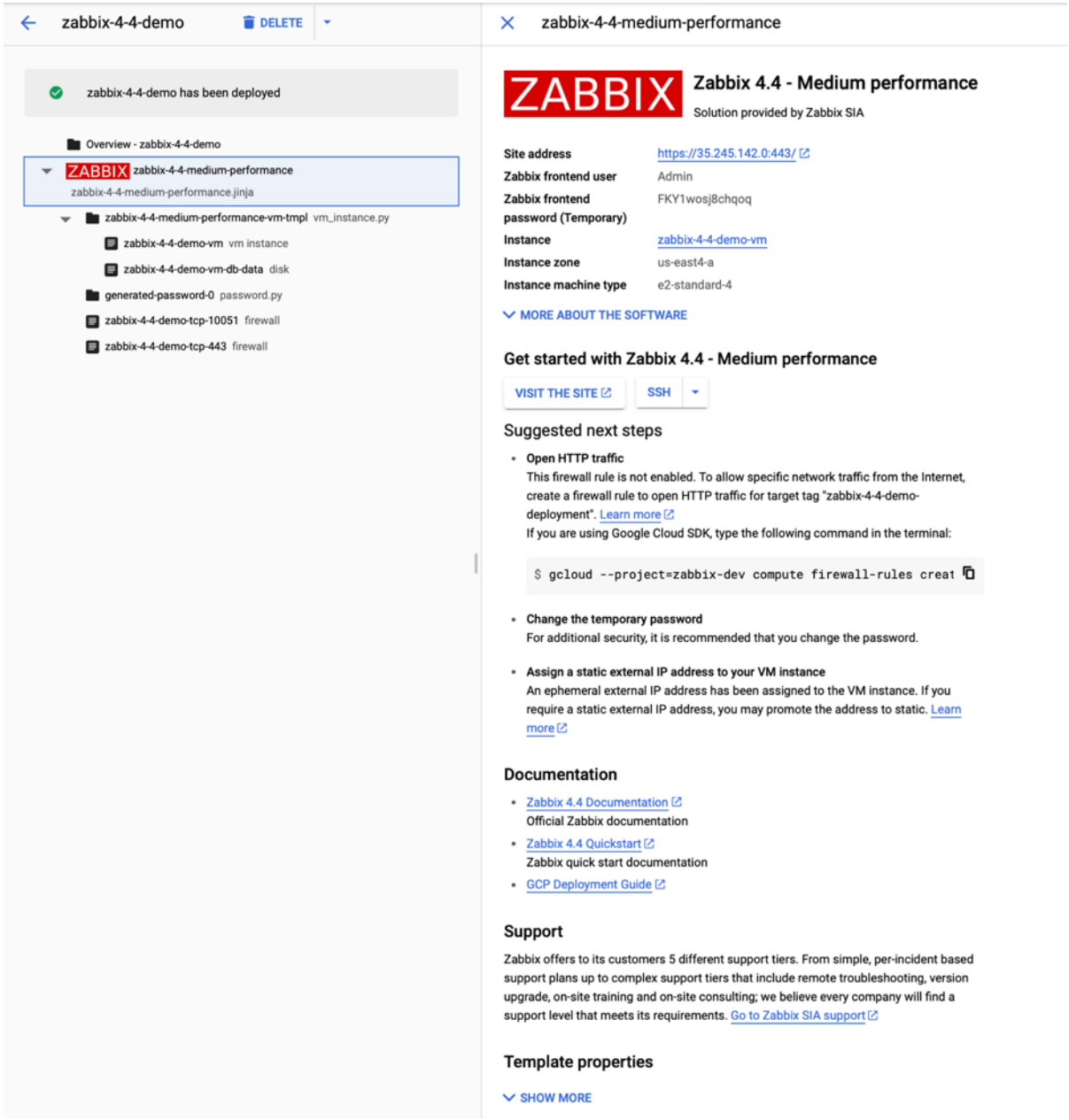
La implementación está en curso.
Espere a que finalice.



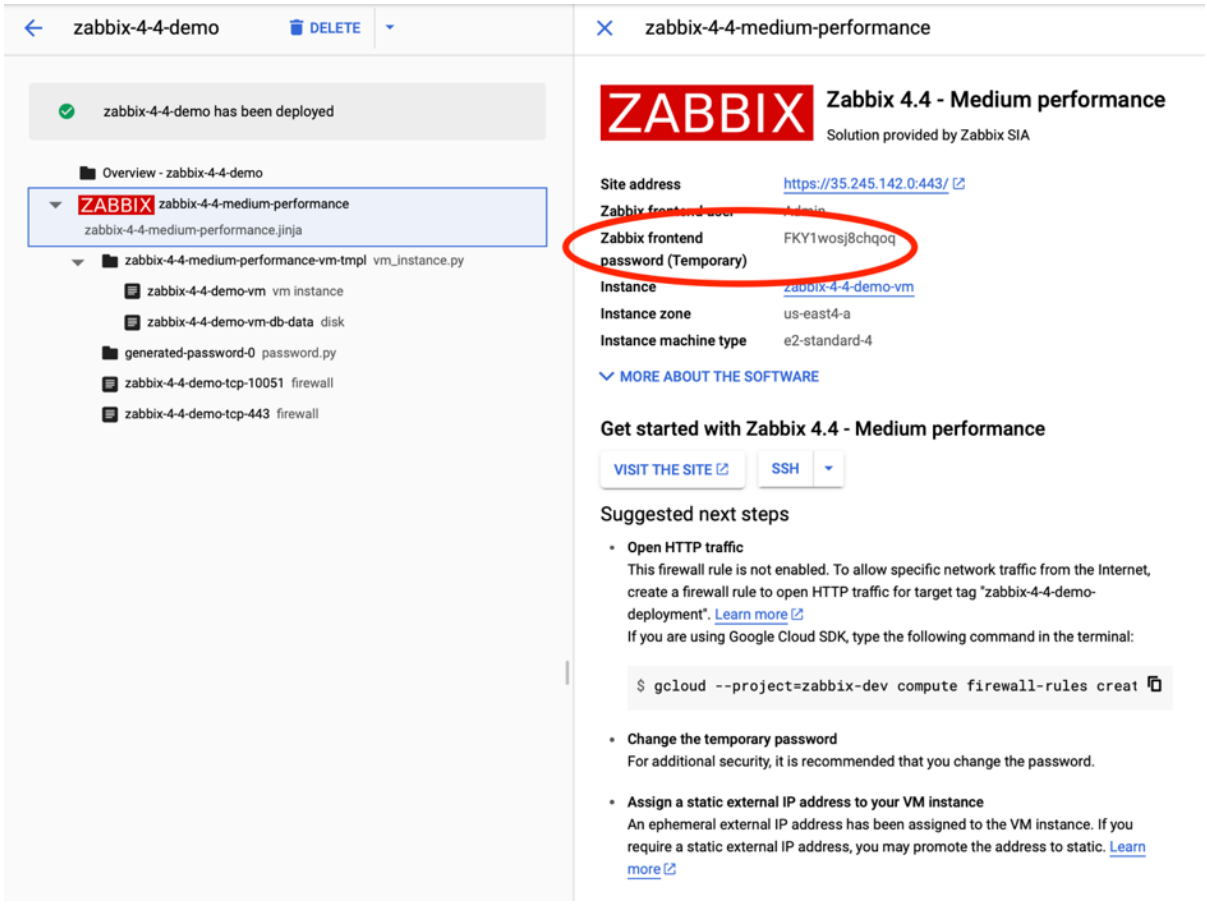
¡Eso es todo! La implementación ya ha comenzado.

Acceso al appliance

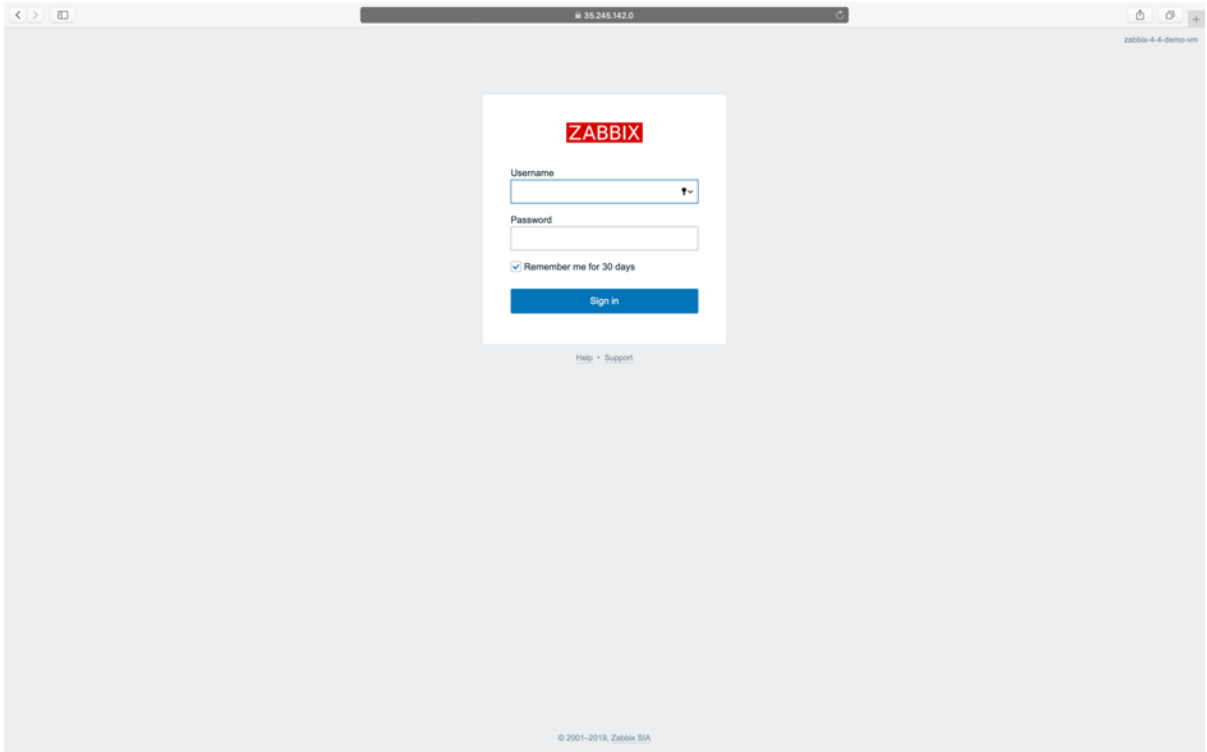
Cuando se complete la implementación, debería aparecer un mensaje que indique que el appliance se está implementando.

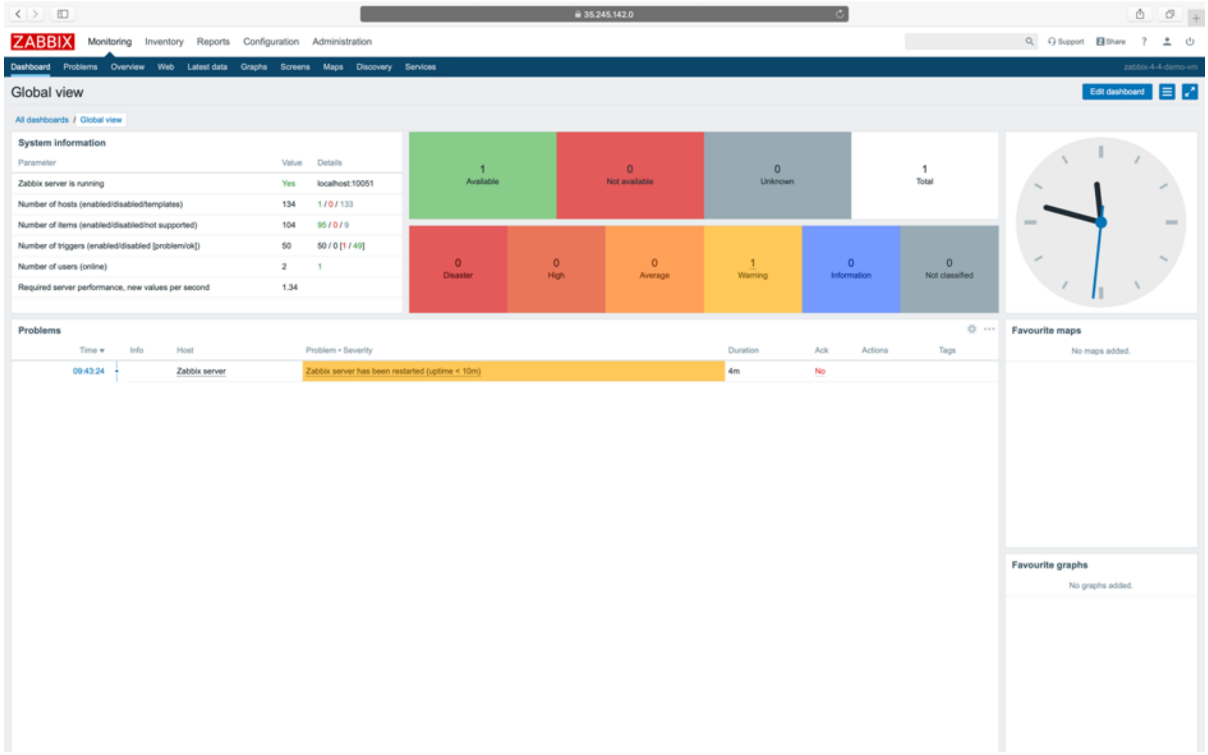


Abra un navegador web en *Site Address* para ver el frontend de Zabbix. Antes de hacerlo, obtenga la contraseña del usuario *Admin* de Zabbix para iniciar sesión en su appliance de Zabbix.

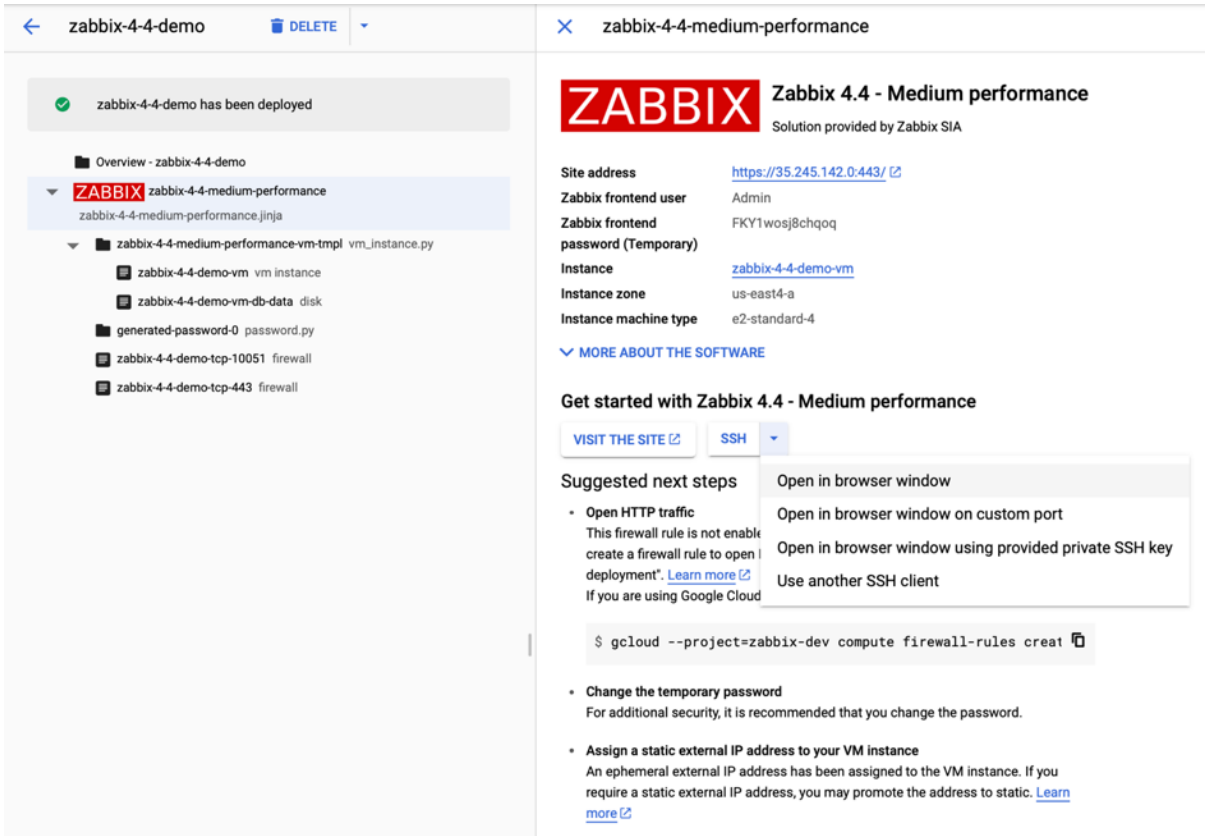


¡Perfecto! Ahora dispone de un appliance de Zabbix.





También puede iniciar sesión en la instancia para interactuar con la base de datos. Para ello, vuelva a la consola de Google y siga la flecha roja que se muestra a continuación para iniciar una sesión SSH mediante la opción *Open in browser window*.



MySQL está disponible sin una contraseña de `root`. La contraseña se almacena en el archivo de configuración `/root/.my.cnf`. De forma predeterminada, Zabbix appliance utiliza la zona horaria UTC. Si necesita cambiarla, edite la directiva de configuración de PHP `date.timezone` en `/etc/php.d/99-zabbix.ini` y reinicie el proceso php-fpm: `systemctl restart php-fpm`.

Actualización

Antes de actualizar, se recomienda encarecidamente consultar todas las notas de actualización relevantes.

También será útil saber qué componentes debe actualizar.

Descripción general

Hay instrucciones de actualización paso a paso disponibles para:

- Red Hat Enterprise Linux (mediante paquetes)
- Debian/Ubuntu (mediante paquetes)
- Actualización mediante contenedores
- Actualización mediante código fuente

Instrucciones relacionadas:

- Para servers en un clúster de alta disponibilidad (HA), consulte Actualización del clúster de HA
- Para la base de datos TimescaleDB, consulte Actualización del esquema de TimescaleDB

Componentes que se actualizarán

Se recomienda actualizar los agentes de Zabbix, pero no es obligatorio.

Se recomienda encarecidamente actualizar los proxies de Zabbix.
El server de Zabbix admite completamente los proxies que tienen la misma versión principal que el server.
El server de Zabbix también admite proxies que **no sean más antiguos** que la versión LTS anterior del server de Zabbix, pero con funcionalidad limitada (recopilación de datos, ejecución de comandos remotos, comprobaciones inmediatas de los valores de los items).
La actualización de la configuración también está deshabilitada, y los proxies obsoletos solo funcionarán con la configuración antigua.

> **Attention:**
> No se admiten proxies con una versión anterior a la versión LTS anterior del server de Zabbix o posterior a la versión principal del server de Zabbix. El server de Zabbix ignorará los datos de los proxies no compatibles y toda comunicación con el server de Zabbix fallará con una advertencia. Para obtener más información, consulte Compatibilidad de versiones.

Para minimizar el tiempo de inactividad y la pérdida de datos durante la actualización, se recomienda detener, actualizar e iniciar el server de Zabbix y, a continuación, detener, actualizar e iniciar los proxies de Zabbix uno tras otro. Durante el tiempo de inactividad del server, los proxies en ejecución continuarán recopilando datos. Una vez que el server esté operativo, los proxies desactualizados enviarán los datos al server más reciente (aunque la configuración del proxy no se actualizará) y seguirán funcionando parcialmente. Las notificaciones de los problemas ocurridos durante el tiempo de inactividad del server de Zabbix solo se generarán después de iniciar el server actualizado.

Si Zabbix proxy se inicia por primera vez y falta el archivo de base de datos SQLite, proxy lo crea automáticamente.

Nota: si Zabbix proxy utiliza SQLite3 y, durante el inicio, detecta que la versión del archivo de base de datos existente es anterior a la requerida, **eliminará automáticamente el archivo de base de datos** y creará uno nuevo. Por lo tanto, se perderán los datos históricos almacenados en el archivo de base de datos SQLite. Si la versión de Zabbix proxy es anterior a la versión del archivo de base de datos, Zabbix registrará un error y finalizará.

Según el tamaño de la base de datos, la actualización de la base de datos a la versión 8.0 puede tardar mucho tiempo.

Notas de actualización

La actualización directa a Zabbix 8.0.x es compatible desde las versiones **2.0**.x.
Para actualizar desde versiones anteriores, consulte la documentación de Zabbix para la versión 2.0 y anteriores.

> **Note:**
> Tenga en cuenta que, después de la actualización, algunas integraciones de software de terceros en Zabbix podrían verse afectadas si el software externo no es compatible con la versión actualizada de Zabbix.

Las siguientes notas de actualización están disponibles:

Actualizar desde	Leer las notas de actualización completas	Cambios más importantes entre versiones
7.4.x	Para: Zabbix 8.0	Los Zabbix agents de versiones anteriores a la 1.6 ya no son compatibles. Se han aumentado las versiones mínimas requeridas de la base de datos. La versión mínima requerida de PHP se ha aumentado de 8.0.0 a 8.2.0. El carácter % se ha añadido a la lista `UnsafeUserParameters` para Zabbix agent y Zabbix agent 2.
7.2.x	Para: Zabbix 7.4 Zabbix 8.0	Se ha eliminado la compatibilidad con la biblioteca PCRE.
7.0.x	Para: Zabbix 7.2 Zabbix 7.4 Zabbix 8.0	Se ha eliminado la compatibilidad con Oracle DB.
6.4.x	Para: Zabbix 7.0 Zabbix 7.2 Zabbix 7.4 Zabbix 8.0	La versión mínima requerida de PHP se ha aumentado de 7.4.0 a 8.0.0. Se han añadido pollers asíncronos para las comprobaciones de agent, HTTP agent, SNMP walk[OID] y SNMP get[OID]. Tabla de base de datos independiente para los proxies. Se ha cambiado la ubicación predeterminada del archivo de configuración del agent para Windows. Oracle DB está obsoleto. Se ha eliminado el tipo de valor numérico (float) antiguo.
6.2.x	Para: Zabbix 6.4 Zabbix 7.0 Zabbix 7.2 Zabbix 7.4 Zabbix 8.0	La versión mínima requerida de MySQL se ha aumentado de 8.0.0 a 8.0.30. La biblioteca 'libevent_pthreads' es necesaria para Zabbix server/proxy. Durante el primer inicio después de una actualización, un Zabbix proxy con SQLite3 elimina automáticamente la versión antigua de la base de datos (con todo el historial) y crea una nueva.
6.0.x LTS	Para: Zabbix 6.2 Zabbix 6.4 Zabbix 7.0 Zabbix 7.2 Zabbix 7.4 Zabbix 8.0	La versión mínima requerida de PHP se ha aumentado de 7.2.5 a 7.4.0. La monitorización de servicios se ha rediseñado considerablemente. Durante la actualización deben crearse triggers deterministas. Si el registro binario está habilitado para MySQL/MariaDB, se requieren privilegios de superusuario o establecer la variable/parámetro de configuración *log_bin_trust_function_creators = 1*. Consulte Scripts de creación de la base de datos para obtener instrucciones sobre cómo establecer la variable.
5.4.x	Para: Zabbix 6.0 Zabbix 6.2 Zabbix 6.4 Zabbix 7.0 Zabbix 7.2 Zabbix 7.4 Zabbix 8.0	Se han aumentado las versiones mínimas requeridas de la base de datos. El server/proxy no se iniciará si la base de datos está desactualizada. Se han perdido registros del registro de auditoría debido a un cambio en la estructura de la base de datos.
5.2.x	Para: Zabbix 5.4 Zabbix 6.0 Zabbix 6.2 Zabbix 6.4 Zabbix 7.0 Zabbix 7.2 Zabbix 7.4 Zabbix 8.0	Se han aumentado las versiones mínimas requeridas de la base de datos. Los items agregados se han eliminado como tipo independiente.

Actualizar desde	Leer las notas de actualización completas	Cambios más importantes entre versiones
5.0.x LTS	Para: Zabbix 5.2 Zabbix 5.4 Zabbix 6.0 Zabbix 6.2 Zabbix 6.4 Zabbix 7.0 Zabbix 7.2 Zabbix 7.4 Zabbix 8.0	La versión mínima requerida de PHP se ha aumentado de 7.2.0 a 7.2.5. El algoritmo de hash de contraseñas ha cambiado de MD5 a bcrypt.
4.4.x	Para: Zabbix 5.0 Zabbix 5.2 Zabbix 5.4 Zabbix 6.0 Zabbix 6.2 Zabbix 6.4 Zabbix 7.0 Zabbix 7.2 Zabbix 7.4 Zabbix 8.0	Se ha eliminado la compatibilidad con IBM DB2. La versión mínima requerida de PHP se ha aumentado de 5.4.0 a 7.2.0. Se han aumentado las versiones mínimas requeridas de la base de datos. Se ha cambiado el directorio de archivos PHP de Zabbix.
4.2.x	Para: Zabbix 4.4 Zabbix 5.0 Zabbix 5.2 Zabbix 5.4 Zabbix 6.0 Zabbix 6.2 Zabbix 6.4 Zabbix 7.0 Zabbix 7.2 Zabbix 7.4 Zabbix 8.0	Se han eliminado los tipos de medio Jabber y Ez Texting.
4.0.x LTS	Para: Zabbix 4.2 Zabbix 4.4 Zabbix 5.0 Zabbix 5.2 Zabbix 5.4 Zabbix 6.0 Zabbix 6.2 Zabbix 6.4 Zabbix 7.0 Zabbix 7.2 Zabbix 7.4 Zabbix 8.0	Los proxies antiguos ya no pueden enviar datos a un server actualizado. Los agents más recientes ya no podrán trabajar con un server de Zabbix antiguo.
3.4.x	Para: Zabbix 4.0 Zabbix 4.2 Zabbix 4.4 Zabbix 5.0 Zabbix 5.2 Zabbix 5.4 Zabbix 6.0 Zabbix 6.2 Zabbix 6.4 Zabbix 7.0 Zabbix 7.2 Zabbix 7.4 Zabbix 8.0	Las bibliotecas ’libpthread’ y ’zlib’ ahora son obligatorias. Se ha eliminado la compatibilidad con el protocolo de texto sin formato y el encabezado es obligatorio. Los Zabbix agents de versiones anteriores a la 1.4 ya no son compatibles. El parámetro Server en la configuración del proxy pasivo ahora es obligatorio.

Actualizar desde	Leer las notas de actualización completas	Cambios más importantes entre versiones
3.2.x	Para: Zabbix 3.4 Zabbix 4.0 Zabbix 4.2 Zabbix 4.4 Zabbix 5.0 Zabbix 5.2 Zabbix 5.4 Zabbix 6.0 Zabbix 6.2 Zabbix 6.4 Zabbix 7.0 Zabbix 7.2 Zabbix 7.4 Zabbix 8.0	Se ha eliminado la compatibilidad con SQLite como base de datos backend para Zabbix server/frontend. Se admiten expresiones regulares compatibles con Perl (PCRE) en lugar de las extendidas POSIX. Las bibliotecas 'libpcre' y 'libevent' son obligatorias para Zabbix server. Se han añadido comprobaciones del código de salida para los parámetros de usuario, los comandos remotos y los items system.run[] sin la marca 'nowait', así como para los scripts ejecutados por Zabbix server. Es necesario actualizar Zabbix Java gateway para admitir nuevas funcionalidades.
3.0.x LTS	Para: Zabbix 3.2 Zabbix 3.4 Zabbix 4.0 Zabbix 4.2 Zabbix 4.4 Zabbix 5.0 Zabbix 5.2 Zabbix 5.4 Zabbix 6.0 Zabbix 6.2 Zabbix 6.4 Zabbix 7.0 Zabbix 7.2 Zabbix 7.4 Zabbix 8.0	La actualización de la base de datos puede ser lenta, según el tamaño de la tabla de historial.
2.4.x	Para: Zabbix 3.0 Zabbix 3.2 Zabbix 3.4 Zabbix 4.0 Zabbix 4.2 Zabbix 4.4 Zabbix 5.0 Zabbix 5.2 Zabbix 5.4 Zabbix 6.0 Zabbix 6.2 Zabbix 6.4 Zabbix 7.0 Zabbix 7.2 Zabbix 7.4 Zabbix 8.0	La versión mínima requerida de PHP se ha aumentado de 5.3.0 a 5.4.0. El parámetro LogFile del agent debe especificarse.

Actualizar desde	Leer las notas de actualización completas	Cambios más importantes entre versiones
2.2.x LTS	Para: Zabbix 2.4 Zabbix 3.0 Zabbix 3.2 Zabbix 3.4 Zabbix 4.0 Zabbix 4.2 Zabbix 4.4 Zabbix 5.0 Zabbix 5.2 Zabbix 5.4 Zabbix 6.0 Zabbix 6.2 Zabbix 6.4 Zabbix 7.0 Zabbix 7.2 Zabbix 7.4 Zabbix 8.0	Se ha eliminado la monitorización distribuida basada en nodos.
2.0.x	Para: Zabbix 2.2 Zabbix 2.4 Zabbix 3.0 Zabbix 3.2 Zabbix 3.4 Zabbix 4.0 Zabbix 4.2 Zabbix 4.4 Zabbix 5.0 Zabbix 5.2 Zabbix 5.4 Zabbix 6.0 Zabbix 6.2 Zabbix 6.4 Zabbix 7.0 Zabbix 7.2 Zabbix 7.4 Zabbix 8.0	La versión mínima requerida de PHP se ha aumentado de 5.1.6 a 5.3.0. Se requiere una base de datos MySQL que distinga entre mayúsculas y minúsculas para el funcionamiento correcto del server; se requieren el juego de caracteres utf8 y la intercalación utf8_bin para que Zabbix server funcione correctamente con una base de datos MySQL. Consulte scripts de creación de la base de datos. Se requiere la extensión PHP 'mysqli' en lugar de 'mysql'.

Actualización desde paquetes

Descripción general

Esta sección proporciona los pasos necesarios para una actualización exitosa utilizando los paquetes oficiales RPM y DEB proporcionados por Zabbix para:

- Red Hat Enterprise Linux
- Debian/Ubuntu

Paquetes de Zabbix de los repositorios del sistema operativo

Algunas distribuciones de sistemas operativos (en particular, las distribuciones basadas en Debian) proporcionan sus propios paquetes de Zabbix. Estos paquetes **no están soportados por Zabbix** y pueden estar desactualizados o carecer de las últimas funciones y correcciones de errores. Se recomienda utilizar únicamente los paquetes oficiales del Repositorio Oficial de Zabbix.

Si está actualizando desde paquetes proporcionados por las distribuciones del sistema operativo (o los tuvo instalados en algún momento), siga este procedimiento para cambiar a los paquetes oficiales de Zabbix:

1. Desinstale los paquetes antiguos.
2. Compruebe y elimine cualquier archivo sobrante de la instalación anterior.
3. Instale los paquetes oficiales de Zabbix siguiendo las instrucciones de instalación proporcionadas por Zabbix.

No realice una actualización directa, ya que esto puede resultar en una instalación defectuosa.

Debian/Ubuntu

Descripción general

Esta sección proporciona instrucciones para actualizar de Zabbix **7.4.x** a la última versión de Zabbix **8.0.x** utilizando los paquetes oficiales de Zabbix para Debian/Ubuntu.

Warning:
Antes de actualizar, revise las notas de actualización relevantes y asegúrese de que su sistema cumpla con los requisitos para Zabbix 8.0.

Note:
Considere ejecutar dos sesiones SSH paralelas durante la actualización: una para ejecutar los pasos de actualización y otra para monitorear los registros del servidor/proxy. Por ejemplo, ejecute `tail -f zabbix_server.log` o `tail -f zabbix_proxy.log` en la segunda sesión para ver las últimas entradas del registro y posibles errores en tiempo real. Esto puede ser crítico para instancias en producción.

Para obtener instrucciones sobre cómo actualizar entre versiones menores de Zabbix 8.0.x (por ejemplo, de 8.0.1 a 8.0.3), consulte Actualizar entre versiones menores.

Procedimiento de actualización

1 Detener los procesos de Zabbix

Detenga el servidor de Zabbix para asegurarse de que no se inserten nuevos datos en la base de datos:

```
systemctl stop zabbix-server
```

Si está actualizando el proxy de Zabbix, el agente o el agente 2, detenga también estos componentes:

```
systemctl stop zabbix-proxy
systemctl stop zabbix-agent
systemctl stop zabbix-agent2
```

2 Realizar una copia de seguridad de la base de datos de Zabbix

Realice una copia de seguridad de su base de datos de Zabbix existente para protegerse contra fallos en la actualización (por ejemplo, problemas de espacio en disco, cortes de energía o problemas inesperados).

3 Hacer copia de seguridad de los archivos de configuración de Zabbix, los archivos PHP y los binarios de Zabbix

Haga una copia de seguridad de los archivos de configuración de Zabbix, los archivos PHP y los binarios de Zabbix existentes.

Para los archivos de configuración, ejecute:

```
mkdir /opt/zabbix-backup/
cp /etc/zabbix/zabbix_server.conf /opt/zabbix-backup/
##### For Apache:
cp /etc/apache2/conf-enabled/zabbix.conf /opt/zabbix-backup/
##### For Nginx:
cp /etc/nginx/conf.d/zabbix.conf /opt/zabbix-backup/
cp /etc/zabbix/nginx.conf /opt/zabbix-backup/
```

Para los archivos PHP y los binarios de Zabbix, ejecute:

```
cp -R /usr/share/zabbix/ /opt/zabbix-backup/
cp -R /usr/share/zabbix-* /opt/zabbix-backup/
```

4 Actualizar el paquete de configuración del repositorio

Antes de continuar con la actualización, desinstale el paquete actual del repositorio de Zabbix:

```
rm -Rf /etc/apt/sources.list.d/zabbix.list
```

También puede ser necesario eliminar manualmente cualquier paquete antiguo de Zabbix de su directorio de trabajo (por ejemplo, `rm zabbix-release_latest+debian12_all.deb`) antes de descargar el nuevo, para evitar que el gestor de paquetes reutilice una versión desactualizada durante el proceso de actualización.

A continuación, instale el paquete de configuración del repositorio más reciente para garantizar la compatibilidad con los paquetes más nuevos e incluir cualquier parche de seguridad o corrección de errores reciente.

En **Debian 13**, ejecute:

```
wget https://repo.zabbix.com/zabbix/8.0/release/debian/pool/main/z/zabbix-release/zabbix-release_latest+de
dpkg -i zabbix-release_latest+debian13_all.deb
```

En **Debian 12**, ejecute:

```
wget https://repo.zabbix.com/zabbix/8.0/release/debian/pool/main/z/zabbix-release/zabbix-release_latest+de
dpkg -i zabbix-release_latest+debian12_all.deb
```

Note:
Para versiones anteriores de Debian, sustituya el enlace anterior por el correcto desde el repositorio de Zabbix. Tenga en cuenta, sin embargo, que los paquetes para estas versiones pueden no incluir todos los componentes de Zabbix, y para actualizar esos componentes desde paquetes, considere actualizar su sistema operativo. Para ver una lista de los componentes incluidos, consulte Paquetes de Zabbix.

En **Ubuntu 26.04**, ejecute:

```
wget https://repo.zabbix.com/zabbix/8.0/release/ubuntu/pool/main/z/zabbix-release/zabbix-release_latest+ub
dpkg -i zabbix-release_latest+ubuntu26.04_all.deb
```

En **Ubuntu 24.04**, ejecute:

```
wget https://repo.zabbix.com/zabbix/8.0/release/ubuntu/pool/main/z/zabbix-release/zabbix-release_latest+ub
dpkg -i zabbix-release_latest+ubuntu24.04_all.deb
```

En **Ubuntu 22.04**, ejecute:

```
wget https://repo.zabbix.com/zabbix/8.0/release/ubuntu/pool/main/z/zabbix-release/zabbix-release_latest+ub
dpkg -i zabbix-release_latest+ubuntu22.04_all.deb
```

Note:
Para versiones anteriores de Ubuntu, sustituya el enlace anterior por el correcto desde el repositorio de Zabbix. Tenga en cuenta, sin embargo, que los paquetes para estas versiones pueden no incluir todos los componentes de Zabbix, y para actualizar esos componentes desde paquetes, considere actualizar su sistema operativo. Para ver una lista de los componentes incluidos, consulte Paquetes de Zabbix.

Es posible que vea un aviso sobre la configuración del repositorio de Zabbix:

```
Configuration file '/etc/apt/sources.list.d/zabbix.list'
==> Deleted (by you or by a script) since installation.
==> Package distributor has shipped an updated version.
What would you like to do about it ?  Your options are:
Y or I  : install the package maintainer's version
N or O  : keep your currently-installed version
D       : show the differences between the versions
Z       : start a shell to examine the situation
The default action is to keep your current version.
*** zabbix.list (Y/I/N/O/D/Z) [default=N] ?
```

Introduzca `Y` (o `I`) para instalar la versión del mantenedor del paquete de la configuración del repositorio de Zabbix.

Luego, actualice la información del repositorio:

```
apt update
```

5 Actualizar los componentes de Zabbix

Para actualizar los componentes de Zabbix, ejecute:

```
apt install --only-upgrade zabbix-server-mysql zabbix-frontend-php zabbix-agent
```

- Si utiliza PostgreSQL, sustituya `mysql` por `pgsql` en el comando.
- Si actualiza el proxy, sustituya `server` por `proxy` en el comando.
- Si actualiza Zabbix agent 2, sustituya `zabbix-agent` por `zabbix-agent2 zabbix-agent2-plugin-*` en el comando.

Attention:
Actualizar Zabbix agent 2 con el comando `apt install zabbix-agent2` podría provocar un error. Para obtener más información, consulte *Problemas conocidos*.

Es posible que aparezca un aviso sobre la configuración del server de Zabbix (o del proxy):

```
Configuration file '/etc/zabbix/zabbix_server.conf'
==> Modified (by you or by a script) since installation.
==> Package distributor has shipped an updated version.
What would you like to do about it ?  Your options are:
Y or I  : install the package maintainer's version
N or O  : keep your currently-installed version
D       : show the differences between the versions
Z       : start a shell to examine the situation
The default action is to keep your current version.
*** zabbix_server.conf (Y/I/N/O/D/Z) [default=N] ?
```

Introduzca la opción que mejor se adapte a su situación. Por ejemplo, introduzca `D` para comparar la configuración actual con la nueva y, a continuación, decida si desea instalar la versión del mantenedor del paquete (`Y` o `I`).

A continuación, actualice el frontend de Zabbix y reinicie el web server.

Para Apache, ejecute:

```
apt install zabbix-apache-conf
systemctl restart apache2
```

Para Nginx, ejecute:

```
apt install zabbix-nginx-conf
systemctl restart nginx
```

Para Nginx, es posible que aparezca un aviso sobre el archivo de configuración `nginx.conf`. Introduzca la opción que mejor se adapte a su situación:

```
Configuration file '/etc/zabbix/nginx.conf'
 ==> Modified (by you or by a script) since installation.
 ==> Package distributor has shipped an updated version.
   What would you like to do about it ?  Your options are:
    Y or I  : install the package maintainer's version
    N or O  : keep your currently-installed version
      D     : show the differences between the versions
      Z     : start a shell to examine the situation
 The default action is to keep your current version.
*** nginx.conf (Y/I/N/O/D/Z) [default=N] ?
```

Si, después de la actualización, necesita instalar un paquete adicional de Zabbix desde el mismo repositorio, `apt` puede seleccionar actualizaciones para los paquetes de Zabbix ya instalados durante la resolución de dependencias. Para evitar actualizar un paquete de server o proxy de Zabbix instalado, mantenga temporalmente ese paquete antes de la instalación y libérelo después:

```
apt-mark hold zabbix-server-mysql
apt install zabbix-sender
apt-mark unhold zabbix-server-mysql
```

6 Revisar los parámetros de configuración del componente

Revise las notas de actualización relevantes para comprobar si es necesario realizar algún cambio en los parámetros de configuración.

Para los nuevos parámetros opcionales, consulte la página Novedades.

Si usa Nginx como servidor web y el archivo de configuración `/etc/zabbix/nginx.conf` se reinstaló durante la actualización, revise el archivo y actualice las directivas `listen` y `server_name` para que coincidan con la configuración de su frontend de Zabbix.

7 Iniciar los procesos de Zabbix

Inicie los componentes actualizados de Zabbix:

```
systemctl start zabbix-server
systemctl start zabbix-proxy
systemctl start zabbix-agent
systemctl start zabbix-agent2
```

8 Borrar las cookies y la caché del navegador web

Después de la actualización, es posible que deba borrar las cookies y la caché del navegador web para que la interfaz web de Zabbix funcione correctamente.

Actualización entre versiones menores

Es posible actualizar entre versiones menores de Zabbix 8.0.x (por ejemplo, de 8.0.1 a 8.0.3).

Primero, actualice la información del repositorio:

```
apt update
```

Luego, para actualizar todos los componentes de Zabbix, ejecute:

```
apt install --only-upgrade 'zabbix*'
```

- Para actualizar solo el servidor Zabbix, reemplace `'zabbix*'` por `'zabbix-server*'` en el comando.
- Para actualizar solo el proxy Zabbix, reemplace `'zabbix*'` por `'zabbix-proxy*'` en el comando.
- Para actualizar solo el agente Zabbix, reemplace `'zabbix*'` por `'zabbix-agent*'` en el comando.
- Para actualizar solo el agente Zabbix 2, reemplace `'zabbix*'` por `'zabbix-agent2*'` en el comando.

Red Hat Enterprise Linux

Descripción general

Esta sección proporciona instrucciones para actualizar de Zabbix **7.4.x** a la última versión de Zabbix **8.0.x** utilizando los paquetes oficiales de Zabbix para Red Hat Enterprise Linux o sus derivados: AlmaLinux, CentOS Stream, Oracle Linux y Rocky Linux.

Warning:
Antes de actualizar, revise las notas de actualización correspondientes y asegúrese de que su sistema cumpla con los requisitos para Zabbix 8.0.

Note:
Considere ejecutar dos sesiones SSH en paralelo durante la actualización: una para ejecutar los pasos de actualización y otra para monitorizar los registros del servidor/proxy. Por ejemplo, ejecute `tail -f zabbix_server.log` o `tail -f zabbix_proxy.log` en la segunda sesión para ver las últimas entradas del registro y posibles errores en tiempo real. Esto puede ser crítico para instancias en producción.

Para obtener instrucciones sobre cómo actualizar entre versiones menores de Zabbix 8.0.x (por ejemplo, de 8.0.1 a 8.0.3), consulte Actualizar entre versiones menores.

Procedimiento de actualización

1 Detener los procesos de Zabbix

Detenga el servidor Zabbix para asegurarse de que no se inserten nuevos datos en la base de datos:

```
systemctl stop zabbix-server
```

Si está actualizando el proxy Zabbix, el agente o el agente 2, detenga también estos componentes:

```
systemctl stop zabbix-proxy
systemctl stop zabbix-agent
systemctl stop zabbix-agent2
```

2 Realizar una copia de seguridad de la base de datos de Zabbix

Realice una copia de seguridad de su base de datos de Zabbix existente para protegerse contra fallos en la actualización (por ejemplo, problemas de espacio en disco, cortes de energía o problemas inesperados).

3 Realizar una copia de seguridad de los archivos de configuración de Zabbix, archivos PHP y binarios de Zabbix

Realice una copia de seguridad de los archivos de configuración existentes de Zabbix, archivos PHP y binarios de Zabbix.

Para los archivos de configuración, ejecute:

```
mkdir /opt/zabbix-backup/
cp /etc/zabbix/zabbix_server.conf /opt/zabbix-backup/
cp /etc/httpd/conf.d/zabbix.conf  /opt/zabbix-backup/
```

Para los archivos PHP y los binarios de Zabbix, ejecute:

```
cp -R /usr/share/zabbix/ /opt/zabbix-backup/
cp -R /usr/share/zabbix-* /opt/zabbix-backup/
```

4 Actualizar el paquete de configuración del repositorio

Antes de continuar con la actualización, actualice el paquete de configuración del repositorio actual a la versión más reciente para garantizar la compatibilidad con los paquetes más nuevos e incluir los parches de seguridad o las correcciones de errores recientes.

En **RHEL 10**, ejecute:

```
rpm -Uvh https://repo.zabbix.com/zabbix/8.0/release/rhel/10/noarch/zabbix-release-latest.el10.noarch.rpm
```

Si está actualizando el proxy de Zabbix en RHEL 10 y desea utilizar APM, instale también el repositorio EPEL y habilite el repositorio CRB, ya que ambos proporcionan dependencias para el paquete APM del proxy de Zabbix:

```
dnf -y install epel-release || dnf -y install https://dl.fedoraproject.org/pub/epel/epel-release-latest-$(
crb enable
```

En **RHEL 9**, ejecute:

```
rpm -Uvh https://repo.zabbix.com/zabbix/8.0/release/rhel/9/noarch/zabbix-release-latest.el9.noarch.rpm
```

> **Note:**
> Para versiones anteriores de RHEL o sus derivados, sustituya el enlace anterior por el correcto de Zabbix repository. Tenga en cuenta, sin embargo, que los paquetes para estas versiones pueden no incluir todos los componentes de Zabbix. Para actualizar dichos componentes mediante paquetes, considere actualizar el sistema operativo. Para consultar la lista de componentes incluidos, vea Zabbix packages.

A continuación, limpie la caché del gestor de paquetes `dnf` (incluidos los encabezados, los metadatos y los archivos de paquetes descargados durante instalaciones o actualizaciones anteriores):

```
dnf clean all
```

En la siguiente operación de `dnf`, `dnf` descargará metadatos nuevos de los repositorios, ya que se han eliminado los metadatos antiguos.

Consulte también: Known issues para actualizar el paquete de configuración del repositorio en RHEL.

5 Actualizar los componentes de Zabbix

Para actualizar los componentes de Zabbix, ejecute:

```
dnf install zabbix-server-mysql zabbix-web-mysql zabbix-selinux-policy zabbix-agent
```

- Si utiliza PostgreSQL, sustituya `mysql` por `pgsql` en el comando.
- Si está actualizando el proxy, sustituya `server` por `proxy` en el comando.
- Si tiene RHEL 10 y desea utilizar APM, añada `-apm` al nombre del paquete del proxy (por ejemplo, `zabbix-proxy-mysql-apm`).
- Si está actualizando el agent 2, sustituya `zabbix-agent` por `zabbix-agent2 zabbix-agent2-plugin-*` en el comando.
- Si tiene instalados otros paquetes de Zabbix, considere actualizarlos también incluyéndolos en el comando.

> **Attention:**
> Actualizar Zabbix agent 2 con el comando `dnf install zabbix-agent2` podría provocar un error. Para obtener más información, consulte *Problemas conocidos*.

A continuación, para actualizar el frontend de Zabbix con Apache y reiniciar Apache, ejecute:

```
dnf install zabbix-apache-conf
systemctl restart httpd
```

6 Revisar los parámetros de configuración del componente

Revise las notas de actualización relevantes para comprobar si es necesario realizar algún cambio en los parámetros de configuración.

Para los nuevos parámetros opcionales, consulte la página Novedades.

7 Iniciar los procesos de Zabbix

Inicie los componentes actualizados de Zabbix:

```
systemctl start zabbix-server
systemctl start zabbix-proxy
systemctl start zabbix-agent
systemctl start zabbix-agent2
```

8 Borrar las cookies y la caché del navegador web

Después de la actualización, es posible que deba borrar las cookies y la caché del navegador web para que la interfaz web de Zabbix funcione correctamente.

Actualización entre versiones menores

Es posible actualizar entre versiones menores de Zabbix 8.0.x (por ejemplo, de 8.0.1 a 8.0.3).

Para actualizar todos los componentes de Zabbix, ejecute:

```
dnf upgrade 'zabbix-*'
```

- Para actualizar solo el servidor Zabbix, reemplace `'zabbix-*'` por `'zabbix-server-*'` en el comando.
- Para actualizar solo el proxy Zabbix, reemplace `'zabbix-*'` por `'zabbix-proxy-*'` en el comando.
- Para actualizar solo el agente Zabbix, reemplace `'zabbix-*'` por `'zabbix-agent-*'` en el comando.
- Para actualizar solo el agente Zabbix 2, reemplace `'zabbix-*'` por `'zabbix-agent2-*'` en el comando.

Actualización desde contenedores

Resumen

Esta sección describe los pasos necesarios para una actualización correcta a los contenedores de Zabbix **8.0**.x.

Hay disponibles conjuntos de instrucciones separados para:

- Actualización de la imagen de componentes de Zabbix
- Actualización del archivo Docker Compose

Warning:
¡Antes de la actualización, asegúrese de leer las **notas de actualización** correspondientes!

Attention:
Antes de iniciar la actualización, verifique que los usuarios tengan los permisos necesarios sobre la base de datos para fines de actualización.

 Para actualizaciones desde Zabbix 6.0 o versiones anteriores, será necesario crear triggers deterministas durante la actualización. Si el registro binario está habilitado para MySQL/MariaDB, esto requiere privilegios de superusuario o establecer la variable/parámetro de configuración `log_bin_trust_function_creators = 1`. Consulte Scripts de creación de base de datos para obtener instrucciones sobre cómo establecer la variable.

 Tenga en cuenta que, si se ejecuta desde una consola, la variable solo se establecerá temporalmente y se eliminará cuando se reinicie un Docker. En este caso, mantenga su servicio SQL en ejecución; detenga únicamente el servicio `zabbix-server` ejecutando `docker compose down zabbix-server` y luego `docker compose up -d zabbix-server`.

 Como alternativa, puede establecer esta variable en el archivo de configuración.

Dependiendo del tamaño de la base de datos, la actualización de la base de datos a la versión 8.0 puede tardar mucho tiempo.

Actualización de la imagen del componente Zabbix

Estas instrucciones actualizan las imágenes de Docker que se usan como base para los contenedores de componentes de Zabbix.

Los pasos siguientes usan un proxy de Zabbix como ejemplo.
Reemplace `zabbix-proxy-sqlite3` por el nombre del contenedor de su componente.

1. Compruebe todos los contenedores para identificar sus nombres y las versiones actuales de la imagen:

```
docker ps -a

#### CONTAINER ID   IMAGE                                              (...) NAMES
#### 5a0c2e3d02b4   zabbix/zabbix-proxy-sqlite3:alpine-7.4-latest      (...) zabbix-proxy-sqlite3
```

2. Descargue la nueva versión de la imagen:

```
docker pull zabbix/zabbix-proxy-sqlite3:alpine-8.0-latest
```

3. Detenga el contenedor del componente Zabbix:

```
docker stop zabbix-proxy-sqlite3
```

4. Elimine el contenedor del componente Zabbix:

Attention:
Asegúrese de haber anotado todos los flags de configuración y los montajes de volúmenes usados en el comando `docker run` original antes de continuar, ya que serán necesarios para recrear el contenedor.

```
docker rm zabbix-proxy-sqlite3
```

5. Inicie un nuevo contenedor usando la imagen actualizada:

```
docker run --name zabbix-proxy-sqlite3 \
  -e ZBX_SERVER_HOST=192.0.2.1 \
  -e ZBX_PROXYMODE=0 \
  -e ZBX_HOSTNAME=zabbix-proxy-sqlite3 \
  -v zabbix-proxy-data:/var/lib/zabbix/db_data \
  --init \
  -d \
  zabbix/zabbix-proxy-sqlite3:alpine-8.0-latest
```

Note:
Para obtener más información sobre cómo ejecutar contenedores de Zabbix, consulte Docker (implementación manual).

6. Verifique que la actualización se haya realizado correctamente comprobando los registros del contenedor:

```
docker logs -f zabbix-proxy-sqlite3
```

Actualización del archivo Docker Compose

Estas instrucciones actualizan los archivos de Docker Compose utilizados para desplegar componentes de Zabbix. Siga estas instrucciones si instaló Zabbix usando Docker Compose.

1. Compruebe todos los contenedores para identificar sus nombres y las versiones actuales de las imágenes:

```
docker ps -a

#### CONTAINER ID   IMAGE                                              (...) NAMES
#### d095f486deef   zabbix/zabbix-server-mysql:alpine-7.4-latest       (...) zabbix-docker-zabbix-server-1
#### 70627d833da0   zabbix/zabbix-web-nginx-mysql:alpine-7.4-latest   (...) zabbix-docker-zabbix-web-ngin
#### aaaa664cc185   zabbix/zabbix-server-mysql:alpine-7.4-latest       (...) zabbix-docker-server-db-init-1
#### 9ab3e77492ff   mysql:8.0-oracle                                   (...) zabbix-docker-mysql-server-1
```

2. Vaya al repositorio de Zabbix Docker que clonó anteriormente, descargue las últimas actualizaciones y cambie a la rama 8.0:

Attention:
Si ha modificado `.env`, `compose.yaml` u otros archivos en su repositorio local, un `git pull` puede sobrescribirlos. Haga una copia de seguridad de cualquier cambio local antes de ejecutar este paso.

```
cd zabbix-docker
git pull
git checkout 8.0
```

3. Despliegue los componentes de Zabbix usando el archivo de Docker Compose actualizado:

```
#### Con MySQL como base de datos:
docker compose -f ./compose.yaml up -d

#### Con PostgreSQL como base de datos:
docker compose -f ./compose_pgsql.yaml up -d
```

Note:
Después de que todos los contenedores se vuelvan a crear y se inicien, puede tardar entre 1 y 3 minutos en reanudarse la recopilación de datos. Para obtener más información sobre la ejecución de contenedores de Zabbix, consulte Docker Compose.

4. Verifique que la actualización se haya realizado correctamente comprobando los registros del contenedor (reemplace `zabbix-docker-zabbix-server-1` por el nombre del contenedor de su componente):

```
docker logs -f zabbix-docker-zabbix-server-1
```

Actualización desde fuentes

Descripción general

Esta sección proporciona los pasos necesarios para realizar correctamente una actualización de Zabbix **7.4**.x a Zabbix **8.0**.x utilizando fuentes oficiales de Zabbix.

Warning:
Antes de realizar la actualización, asegúrese de leer las **notas de actualización** correspondientes.

También puede consultar los requisitos de la versión 8.0.

Note:
Puede resultar útil ejecutar dos sesiones SSH en paralelo durante la actualización: en una, ejecute los pasos de actualización y, en la otra, supervise los registros del server/proxy. Por ejemplo, ejecute `tail -f zabbix_server.log` o `tail -f zabbix_proxy.log` en la segunda sesión SSH para mostrar las entradas más recientes del archivo de registro y los posibles errores en tiempo real. Esto puede ser fundamental en instancias de producción.

Proceso de actualización del servidor

1 Detener el server

Detenga el server de Zabbix para asegurarse de que no se inserten datos nuevos en la base de datos.

2 Realizar una copia de seguridad de la base de datos de Zabbix existente

Este es un paso muy importante.
Asegúrese de tener una copia de seguridad de su base de datos.
Será útil si el procedimiento de actualización falla (falta de espacio en disco, apagado, cualquier problema inesperado).

3 Realizar una copia de seguridad de los archivos de configuración, los archivos PHP y los binarios de Zabbix

Realice una copia de seguridad de los binarios de Zabbix, los archivos de configuración y el directorio de archivos PHP.

4 Instalar los nuevos binarios del server

Utilice estas instrucciones para compilar el server de Zabbix a partir del código fuente.

5 Revisar los parámetros de configuración del server

Asegúrese de revisar las Notas de actualización para comprobar si se requieren cambios en los parámetros de configuración.

Para los nuevos parámetros opcionales, consulte la página Novedades.

6 Iniciar los nuevos binarios de Zabbix

Inicie los nuevos binarios.
Compruebe los archivos de registro para verificar que los binarios se hayan iniciado correctamente.

Zabbix server actualizará automáticamente la base de datos.
Al iniciarse, Zabbix server informa de las versiones actuales (obligatoria y opcional) y de las versiones requeridas de la base de datos.

Si la versión obligatoria actual es anterior a la versión requerida, Zabbix server ejecuta automáticamente los parches de actualización de la base de datos necesarios.
El inicio y el nivel de progreso (porcentaje) de la actualización de la base de datos se escriben en el archivo de registro de Zabbix server.
Cuando finaliza la actualización, se escribe el mensaje ”database upgrade fully completed” en el archivo de registro.
Si alguno de los parches de actualización falla, Zabbix server no se iniciará.
Zabbix server tampoco se iniciará si la versión obligatoria actual de la base de datos es posterior a la requerida.
Zabbix server solo se iniciará si la versión obligatoria actual de la base de datos corresponde a la versión obligatoria requerida.

```
8673:20161117:104750.259 current database version (mandatory/optional): 03040000/03040000
8673:20161117:104750.259 required mandatory version: 03040000
```

Antes de iniciar el server:

- Asegúrese de que el usuario de la base de datos tenga permisos suficientes (crear tablas, eliminar tablas, crear índices y eliminar índices).
- Asegúrese de disponer de suficiente espacio libre en disco.

7 Instalar la nueva interfaz web de Zabbix

La versión mínima requerida de PHP es la 8.2.0.
Actualícela si es necesario y siga las instrucciones de instalación.

8 Borrar las cookies y la caché del navegador web

Después de la actualización, es posible que deba borrar las cookies y la caché del navegador web para que la interfaz web de Zabbix funcione correctamente.

Proceso de actualización del proxy

1 Detener proxy

Detener el proxy de Zabbix.

2 Realizar una copia de seguridad de los archivos de configuración y binarios del proxy de Zabbix

Haga una copia de seguridad del binario del proxy de Zabbix y del archivo de configuración.

3 Instalar los nuevos binarios del proxy

Utilice estas instrucciones para compilar el proxy de Zabbix a partir del código fuente.

4 Revisar los parámetros de configuración del proxy

Asegúrese de revisar las Notas de actualización para comprobar si se requieren cambios en los parámetros de configuración.

Para los nuevos parámetros opcionales, consulte la página Novedades.

5 Iniciar el nuevo proxy de Zabbix

Inicie el nuevo proxy de Zabbix.
Compruebe los archivos de registro para verificar si el proxy se ha iniciado correctamente.

El proxy de Zabbix actualizará automáticamente la base de datos.
La actualización de la base de datos se realiza de forma similar a cuando se inicia el server de Zabbix.

Proceso de actualización del agent

> **Attention:**
> La actualización de los agents no es obligatoria. Solo debe actualizar los agents si es necesario para acceder a la nueva funcionalidad.

El procedimiento de actualización descrito en esta sección puede utilizarse para actualizar tanto el Zabbix agent como el Zabbix agent 2.

1 Detener el agente

Detenga el agente Zabbix.

2 Realice una copia de seguridad de los archivos de configuración y los binarios del agente Zabbix

Haga una copia de seguridad del binario del agente Zabbix y del archivo de configuración.

3 Instalar nuevos binarios del agent

Utilice estas instrucciones para compilar el agent de Zabbix a partir del código fuente.

Como alternativa, puede descargar agents de Zabbix precompilados desde la página de descargas de Zabbix.

4 Revisar los parámetros de configuración del agent

Asegúrese de revisar las Notas de actualización para comprobar si se requieren cambios en los parámetros de configuración.

Para conocer los nuevos parámetros opcionales, consulte la página Novedades.

5 Iniciar el nuevo Zabbix agent

Inicie el nuevo Zabbix agent.
Compruebe los archivos de registro para verificar si el agent se ha iniciado correctamente.

Actualización entre versiones menores

Al actualizar entre versiones menores de 8.0.x (por ejemplo, de 8.0.1 a 8.0.3), es necesario ejecutar las mismas acciones para el server/proxy/agent que durante la actualización entre versiones principales. La única diferencia es que, al actualizar entre versiones menores, no se realizan cambios en la base de datos.

Problemas de actualización

Relacionado con el escape Límite de caracteres excedido para los parámetros de función después de la actualización

En Zabbix 7.0 se ha añadido el escape adecuado de las barras invertidas en los parámetros de cadena de las funciones de histórico.

Como se añaden barras invertidas adicionales durante la actualización desde versiones de Zabbix anteriores a la 7.0, esto da lugar a parámetros más largos que pueden provocar que las funciones de los triggers se rompan si la longitud del parámetro supera el tamaño máximo de datos de 255 caracteres.

Para ilustrar qué problemas pueden surgir durante la actualización de la base de datos, cuál es su impacto, cómo reconocerlos y solucionarlos, tomemos como ejemplo una configuración con 3 items y 2 triggers.

Configuración

1. Versión 6.0 de Zabbix.
2. Archivo `/tmp/ONE` que contiene 114 barras invertidas con una 'a' al final:

```
\\\
```

```
2485502:20250228:115442.236 DBpatch_6050165(): cannot save in DB function parameter: resulting size 477 is
functionid:33792 function:'find'
used on host: 'Zabbix server'
  in trigger: 'TRIG_228'.
Current parameter value:
'$,1m:now,"regexp","\\\
```

Ahora ambos triggers funcionan igual que antes: solo Trig_228 está **disparado**.

No hay problemas con los items calculados, pero también podrían actualizarse por coherencia:

Calc_228:

```
find(/Zabbix server/vfs.file.contents[/tmp/ONE],1m:now,"regexp",{$228_BACKSLASHES_A})
```

Calc_232:

```
find(/Zabbix server/vfs.file.contents[/tmp/ONE],1m:now,"regexp",{$232_BACKSLASHES_A})
```

Problemas conocidos

Consulte también: Problemas de compilación.

Actualización

Configuración del modo SQL para una actualización exitosa

La configuración `sql_mode` en MySQL/MariaDB debe tener el modo "STRICT_TRANS_TABLES" establecido. Si está ausente, la actualización de la base de datos de Zabbix fallará (consulte también ZBX-19435).

Actualización con MariaDB 10.2.1 y anteriores

La actualización de Zabbix puede fallar si las tablas de la base de datos se crearon con MariaDB 10.2.1 y anteriores, porque en esas versiones el formato de fila predeterminado es compacto. Esto se puede solucionar cambiando el formato de fila a dinámico (ver también ZBX-17690).

Juego de caracteres y cotejamiento de la base de datos MySQL

Las instalaciones de Zabbix que utilizan una base de datos MySQL creada con el juego de caracteres utf8mb3 y el cotejamiento utf8mb3_bin pueden experimentar problemas. Por ejemplo, un problema conocido de MySQL puede provocar errores `Row size too large (> 8126)` durante las actualizaciones del esquema de la base de datos u otras operaciones DDL.
 Se recomienda encarecidamente convertir las bases de datos que utilizan el juego de caracteres utf8mb3 y el cotejamiento utf8mb3_bin al juego de caracteres utf8mb4 y el cotejamiento utf8mb4_bin, tal como se describe en Reparación del juego de caracteres y el cotejamiento de la base de datos de Zabbix.

Fuga de memoria y de descriptores de archivo con libcurl 8.21.0

Un problema conocido en libcurl 8.21.0 puede provocar fugas de memoria y de descriptores de archivo en las funciones de Zabbix que usan libcurl al gestionar conexiones protegidas por TLS.
 Una solución temporal para este problema es degradar libcurl a la versión 8.20.0 hasta que haya disponible una versión que incluya la corrección.

Plantillas

Compatibilidad de plantillas en entornos de doble pila (IPv4/IPv6)

En entornos de doble pila (sistemas configurados para admitir tanto IPv4 como IPv6), el nombre de host `localhost` normalmente se resuelve en direcciones IPv4 e IPv6. Debido a la priorización común de IPv6 sobre IPv4 por parte de muchos sistemas operativos y resolutores DNS, las plantillas de Zabbix pueden dejar de funcionar correctamente si el servicio que se está monitorizando está configurado para escuchar solo en IPv4.

Los servicios que no están configurados para escuchar en direcciones IPv6 pueden volverse inaccesibles, lo que provoca fallos en la monitorización. Los usuarios pueden configurar el acceso correctamente para IPv4 pero aún así enfrentar problemas de conectividad debido al comportamiento predeterminado de priorizar IPv6.

Una solución para esto es asegurarse de que los servicios (Nginx, Apache, PostgreSQL, etc.) estén configurados para escuchar en direcciones IPv4 e IPv6, y que el servidor/agente Zabbix tenga permitido el acceso a través de IPv6. Además, en las plantillas y configuraciones de Zabbix, utilice explícitamente `localhost` en lugar de `127.0.0.1` para garantizar la compatibilidad con IPv4 e IPv6.

Por ejemplo, al monitorizar PostgreSQL con la plantilla PostgreSQL por Zabbix agent 2, es posible que deba editar el archivo `pg_hba.conf` para permitir conexiones para el usuario `zbx_monitor`. Si el entorno de doble pila prioriza IPv6 (el sistema resuelve localhost como `::1`) y configura `localhost` pero solo agrega una entrada IPv4 (`127.0.0.1/32`), la conexión fallará porque no hay una entrada IPv6 correspondiente.

El siguiente ejemplo de archivo `pg_hba.conf` garantiza que el usuario `zbx_monitor` pueda conectarse a cualquier base de datos desde la máquina local utilizando direcciones IPv4 e IPv6 con diferentes métodos de autenticación:

```
### TYPE     DATABASE     USER            ADDRESS          METHOD
  host     all          zbx_monitor     localhost        trust
  host     all          zbx_monitor     127.0.0.1/32     md5
  host     all          zbx_monitor     ::1/128          scram-sha-256
```

Si es necesario, también puede utilizar la dirección IPv4 (`127.0.0.1`) directamente al configurar la macro de la plantilla PostgreSQL por Zabbix agent 2 para la cadena de conexión.

Instalación accidental de paquetes Zabbix de EPEL

Si el repositorio EPEL está instalado y habilitado, al instalar paquetes de Zabbix se pueden instalar versiones de EPEL en lugar de los paquetes oficiales de Zabbix. Para resolver esto:

1. Elimine cualquier paquete de Zabbix instalado desde EPEL:

```
dnf remove zabbix-server-mysql
```

2. Excluya los paquetes de Zabbix de EPEL añadiendo la siguiente línea al archivo `/etc/yum.repos.d/epel.repo`:

```
[epel]
...
excludepkgs=zabbix*
```

3. Reinstale el paquete oficial de Zabbix server:

```
dnf install zabbix-server-mysql
```

Durante la instalación, los paquetes oficiales de Zabbix incluyen la palabra `release` en su cadena de versión (por ejemplo, `7.0.0-release1.el8`), lo que los distingue de los paquetes de EPEL.

Paquetes de Zabbix para RHEL en entornos Red Hat UBI

Al instalar Zabbix desde paquetes de Red Hat Enterprise Linux en entornos Red Hat Universal Base Image, asegúrese de tener acceso a los repositorios y dependencias requeridos. Los paquetes de Zabbix dependen de las bibliotecas `libOpenIPMI.so` y `libOpenIPMIposix.so`, que no son proporcionadas por ningún paquete en los repositorios predeterminados del gestor de paquetes habilitados en sistemas UBI y esto resultará en fallos de instalación.

Las bibliotecas `libOpenIPMI.so` y `libOpenIPMIposix.so` están disponibles en el paquete `OpenIPMI-libs`, que es proporcionado por el repositorio `redhat-#-for-<arch>-appstream-rpms`. El acceso a este repositorio está gestionado por suscripciones, que, en el caso de entornos UBI, se propagan montando la configuración del repositorio y los directorios de secretos del equipo RHEL en el espacio de nombres del sistema de archivos del contenedor.

Para más información, consulte ZBX-24291.

Clave de firma caducada para paquetes de RHEL

Al actualizar Zabbix en Red Hat Enterprise Linux o sus derivados, puede encontrarse con un problema de clave de firma caducada para los paquetes en el repositorio de Zabbix. Cuando una clave de firma caduca, los intentos de verificar las firmas de los paquetes darán como resultado un error que indica que el certificado o la clave ya no son válidos. Por ejemplo:

```
error: Verifying a signature using certificate D9AA84C2B617479C6E4FCF4D19F2475308EFA7DD (Zabbix LLC (Jul 2
  1. Certificate 19F2475308EFA7DD invalid: certificate is not alive
      because: The primary key is not live
      because: Expired on 2024-07-04T11:41:23Z
  2. Key 19F2475308EFA7DD invalid: key is not alive
      because: The primary key is not live
      because: Expired on 2024-07-04T11:41:23Z
```

Para resolver estos problemas, reinstale manualmente el paquete `zabbix-release` más reciente para su variante específica de RHEL (reemplace el enlace a continuación por el correcto del repositorio de Zabbix).

Por ejemplo, en **RHEL 10**, ejecute:

```
rpm -Uvh https://repo.zabbix.com/zabbix/8.0/release/rhel/10/noarch/zabbix-release-latest.el10.noarch.rpm
```

Luego, actualice la información del repositorio:

```
dnf update
```

Para más información, consulte ZBX-24761.

Conexión TLS de base de datos con MariaDB

La conexión TLS de base de datos no es compatible con la opción `verify_ca` para el parámetro DBTLSConnect si se utiliza MariaDB.

Posibles interbloqueos con MySQL/MariaDB

Al ejecutarse bajo alta carga y con más de un trabajador LLD involucrado, es posible encontrarse con un interbloqueo causado por un error de InnoDB relacionado con la estrategia de bloqueo de filas (consulte el error upstream). El error ha sido corregido en MySQL desde la versión 8.0.29, pero no en MariaDB. Para más detalles, consulte ZBX-21506.

Correlación global de eventos

Es posible que los eventos no se correlacionen correctamente si el intervalo de tiempo entre el primer y el segundo evento es muy pequeño, es decir, medio segundo o menos.

NetBSD 8.0 y versiones más recientes

Varios procesos de Zabbix pueden fallar aleatoriamente al iniciarse en las versiones 8.X y 9.X de NetBSD. Esto se debe a que el tamaño de pila por defecto es demasiado pequeño (4MB), por lo que debe aumentarse ejecutando:

```
ulimit -s 10240
```

Para más información, consulte el informe de problema relacionado: ZBX-18275.

Limitaciones de expresiones regulares en Zabbix agent 2

Zabbix agent 2 no admite lookaheads ni lookbehinds en expresiones regulares debido a las limitaciones de la biblioteca estándar Go regexp.

Comprobaciones IPMI

Las comprobaciones IPMI no funcionarán con el paquete estándar de la biblioteca OpenIPMI en Debian anterior a la versión 9 (stretch) y Ubuntu anterior a la versión 16.04 (xenial). Para solucionar esto, vuelva a compilar la biblioteca OpenIPMI con OpenSSL habilitado como se explica en ZBX-6139.

IPMI — los hosts no confiables pueden bloquear OpenIPMI

Existe un error en la biblioteca OpenIPMI utilizada por Zabbix para consultar datos de IPMI que puede activarse mediante respuestas especialmente diseñadas de un dispositivo no confiable. Un dispositivo IPMI no confiable puede enviar datos diseñados para provocar el bloqueo de la biblioteca OpenIPMI, lo que a su vez puede hacer que finalice el proceso del server de Zabbix que realiza las consultas de IPMI.

Comprobaciones SSH

- Algunas distribuciones de Linux como Debian, Ubuntu no admiten claves privadas cifradas (con frase de contraseña) si la biblioteca libssh2 está instalada desde los paquetes. Consulte ZBX-4850 para más detalles.
- Al usar libssh 0.9.x en algunas distribuciones de Linux con OpenSSH 8, las comprobaciones SSH pueden informar ocasionalmente ”No se pueden leer datos del servidor SSH”. Esto es causado por un problema de libssh (informe más detallado). Se espera que el error haya sido corregido en la versión estable libssh 0.9.5. Consulte también ZBX-17756 para más detalles.
- El uso de la tubería ”|” en el script SSH puede provocar un error de ”No se pueden leer datos del servidor SSH”. En este caso, se recomienda actualizar la versión de la biblioteca libssh. Consulte también ZBX-21337 para más detalles.

Comprobaciones ODBC

- El controlador MySQL unixODBC no debe utilizarse con Zabbix server o Zabbix proxy compilados con la biblioteca conectora de MariaDB, y viceversa; si es posible, también es mejor evitar usar el mismo conector que el controlador debido a un error upstream. Configuración recomendada:

  Conector PostgreSQL, SQLite u Oracle → controlador MariaDB o MySQL unixODBC
Conector MariaDB → controlador MariaDB unixODBC
Conector MySQL → controlador MySQL unixODBC

  Consulte ZBX-7665 para más información y soluciones alternativas disponibles.
- Los datos XML consultados desde Microsoft SQL Server pueden truncarse de varias maneras en sistemas Linux y UNIX.
- Se ha observado que el uso de comprobaciones ODBC para monitorizar bases de datos Oracle con varias versiones de Oracle Instant Client para Linux provoca que Zabbix server se bloquee.
 Véase también: ZBX-18402, ZBX-20803.
- Si utiliza el controlador FreeTDS UnixODBC, debe anteponer una instrucción ’SET NOCOUNT ON’ a una consulta SQL (por ejemplo, `SET NOCOUNT ON DECLARE @strsql NVARCHAR(max) SET @strsql = ....`). De lo contrario, el item monitor de base de datos en Zabbix no podrá recuperar la información y mostrará el error ”La consulta SQL devolvió un resultado vacío”.
 Consulte ZBX-19917 para más información.

Parámetro de método de solicitud incorrecto en elementos

El parámetro de método de solicitud, utilizado solo en comprobaciones HTTP, puede estar configurado incorrectamente en '1', un valor no predeterminado para todos los elementos como resultado de una actualización desde una versión de Zabbix anterior a la 4.0. Para obtener detalles sobre cómo solucionar esta situación, consulte ZBX-19308.

Supervisión web y agente HTTP

El servidor Zabbix presenta fugas de memoria en algunas distribuciones de Linux debido a un error upstream cuando "Verificar par SSL" está habilitado en escenarios web o en el agente HTTP. Consulte ZBX-10486 para obtener más información y soluciones alternativas disponibles.

Comprobaciones simples

Existe un error en las versiones de **fping** anteriores a la v3.10 que maneja incorrectamente los paquetes de eco de respuesta duplicados. Esto puede causar resultados inesperados para los elementos `icmpping`, `icmppingloss`, `icmppingsec`. Se recomienda utilizar la última versión de **fping**. Consulte ZBX-11726 para más detalles.

Errores con la ejecución de fping en contenedores sin privilegios de root

Cuando los contenedores se ejecutan en modo sin privilegios de root o en un entorno con restricciones específicas, puede encontrar errores relacionados con la ejecución de fping al realizar comprobaciones ICMP, como `fping: Operation not permitted` o la pérdida de todos los paquetes a todos los recursos.

Para solucionar este problema, agregue `--cap-add=net_raw` a los comandos "docker run" o "podman run".

Además, la ejecución de fping en entornos sin privilegios de root puede requerir la modificación de sysctl, por ejemplo:

```
sudo sysctl -w "net.ipv4.ping_group_range=0 1995"
```

donde "1995" es el GID de zabbix. Para más detalles, consulte ZBX-22833.

Comprobaciones SNMP

Si se utiliza el sistema operativo OpenBSD, un error de uso después de liberar memoria en la biblioteca Net-SNMP hasta la versión 5.7.3 puede causar un fallo en el servidor Zabbix si el parámetro SourceIP está configurado en el archivo de configuración del servidor Zabbix. Como solución temporal, por favor no configure el parámetro SourceIP. El mismo problema también se aplica a Linux, pero no hace que el servidor Zabbix deje de funcionar. Se aplicó un parche local para el paquete net-snmp en OpenBSD y se lanzará con OpenBSD 6.3.

Picos de datos SNMP

Se han observado picos en los datos SNMP que pueden estar relacionados con ciertos factores físicos como picos de voltaje en la red eléctrica. Consulte ZBX-14318 para más detalles.

Trampas SNMP

El paquete "net-snmp-perl", necesario para las trampas SNMP, fue eliminado en RHEL 8.0-8.2; se volvió a agregar en RHEL 8.3.

Por lo tanto, si está utilizando RHEL 8.0-8.2, la mejor solución es actualizar a RHEL 8.3.

Consulte también ZBX-17192 para obtener más información.

Bloqueo del proceso de alerta en RHEL 7

Se han encontrado casos de bloqueo del proceso de alerta del servidor Zabbix en RHEL 7. Consulte ZBX-10461 para más detalles.

Actualización de Zabbix agent 2 (6.0.5 o anterior)

Al actualizar Zabbix agent 2 (versión 6.0.5 o anterior) desde paquetes, puede producirse un error de conflicto de archivos relacionado con el plugin. Para solucionar el error, haga una copia de seguridad de la configuración del agente 2 (si es necesario), desinstale el agente 2 e instálelo de nuevo.

En sistemas basados en RHEL, ejecute:

```
dnf remove zabbix-agent2
dnf install zabbix-agent2
```

En sistemas basados en Debian, ejecute:

```
apt remove zabbix-agent2
apt install zabbix-agent2
```

Para más información, consulte ZBX-23250.

Cambio de las configuraciones regionales del frontend

Se ha observado que las configuraciones regionales del frontend pueden cambiar sin una lógica aparente; es decir, algunas páginas (o partes de ellas) se muestran en un idioma, mientras que otras páginas (o partes de ellas) se muestran en otro idioma.

Normalmente, el problema puede aparecer cuando hay varios usuarios, algunos de los cuales utilizan una configuración regional y otros utilizan otra.

Una solución alternativa conocida consiste en deshabilitar el procesamiento multihilo en PHP y Apache.

El problema está relacionado con el funcionamiento de la configuración regional en PHP: la información de configuración regional se mantiene por proceso, no por hilo.
Por lo tanto, en un entorno multihilo, cuando hay varios proyectos ejecutándose en el mismo proceso de Apache, es posible que la configuración regional cambie en otro hilo y que esto modifique la forma en que se procesan los datos en el hilo de Zabbix.

Para obtener más información, consulte los informes de problemas relacionados:

- ZBX-10911 (Problema con el cambio de las configuraciones regionales del frontend)
- ZBX-16297 (Problema con el procesamiento de números en los gráficos que utilizan la función `bcdiv` de las funciones BC Math)

Gráficos

Problemas con los gráficos (clásicos)

Si encuentra problemas con los gráficos clásicos, se recomienda actualizar la biblioteca GD (libgd) a la versión 2.3.3-13 o superior, y PHP a la versión 8.0.19, 8.1.33, 8.2.29, 8.3.25, 8.4.12 o superior.

Horario de verano

Los cambios en el horario de verano (DST) provocan irregularidades al mostrar las etiquetas del eje X (duplicación de fechas, fechas que faltan, etc.).

Agregación de suma

Al utilizar la agregación de suma en un gráfico para un período inferior a una hora, los gráficos muestran valores incorrectos (multiplicados) cuando los datos provienen de tendencias.

Superposición de texto

Para algunos idiomas del frontend (por ejemplo, japonés), las fuentes locales pueden causar superposición de texto en la leyenda del gráfico. Para evitarlo, utilice la versión 2.3.0 (o posterior) de la extensión PHP GD.

Supervisión de archivos de registro

Los elementos `log[]` y `logrt[]` vuelven a leer repetidamente el archivo de registro desde el principio si el sistema de archivos está al 100% de su capacidad y el archivo de registro se está ampliando (consulte ZBX-10884 para obtener más información).

Consultas lentas de MySQL

El servidor Zabbix genera consultas `SELECT` lentas en caso de valores inexistentes para los ítems. Se sabe que este problema ocurre en las versiones 5.6/5.7 de MySQL (para una discusión más extensa, consulte ZBX-10652), y, en casos específicos, también puede ocurrir en versiones posteriores de MySQL. Una solución temporal para esto es deshabilitar el optimizador `index_condition_pushdown` o `prefer_ordering_index` en MySQL. Sin embargo, tenga en cuenta que esta solución temporal puede no corregir todos los problemas relacionados con consultas lentas.

Configuración de filtros persistentes desde enlaces

Al abrir un enlace a una página del frontend de Zabbix que contiene configuraciones de filtro, incluido el selector de tiempo, el filtro se guarda automáticamente en la base de datos para el usuario, reemplazando la configuración de filtro y/o selector de tiempo previamente guardada para esa página. Estas configuraciones permanecen activas hasta que el usuario las actualice o restablezca manualmente.

Problema con la dirección IPv6 en traps SNMPv3

Debido a un error en net-snmp, la dirección IPv6 puede no mostrarse correctamente al usar SNMPv3 en traps SNMP. Para más detalles y una posible solución, consulte ZBX-14541.

Dirección IP IPv6 larga recortada en la información de inicio de sesión fallido

Un mensaje de intento de inicio de sesión fallido mostrará solo los primeros 39 caracteres de una dirección IP almacenada, ya que ese es el límite de caracteres en el campo de la base de datos. Eso significa que las direcciones IP IPv6 más largas de 39 caracteres se mostrarán de forma incompleta.

Comprobaciones de Zabbix agent en Windows

Las entradas DNS inexistentes en un parámetro `Server` del archivo de configuración de Zabbix agent (`zabbix_agentd.conf`) pueden aumentar el tiempo de respuesta de Zabbix agent en Windows. Esto ocurre porque el demonio de caché DNS de Windows no almacena en caché las respuestas negativas para direcciones IPv4. Sin embargo, para direcciones IPv6 las respuestas negativas sí se almacenan en caché, por lo que una posible solución alternativa es deshabilitar IPv4 en el host.

Exportación/importación de YAML

Hay algunos problemas conocidos con exportación/importación de YAML:

- Los mensajes de error no son traducibles;
- A veces no se puede importar JSON válido con una extensión de archivo .yaml;
- Las fechas legibles por humanos sin comillas se convierten automáticamente en marcas de tiempo Unix.

Asistente de configuración en SUSE con NGINX y php-fpm

El asistente de configuración del frontend no puede guardar el archivo de configuración en SUSE con NGINX + php-fpm. Esto se debe a una configuración en la unidad /usr/lib/systemd/system/php-fpm.service, que impide que Zabbix escriba en /etc. (introducido en PHP 7.4).

Hay dos opciones de solución disponibles:

- Establecer la opción ProtectSystem en 'true' en lugar de 'full' en la unidad systemd de php-fpm.
- Guardar manualmente el archivo `/etc/zabbix/web/zabbix.conf.php`.

Reenvío de la cabecera Authorization

En algunos casos, Apache o NGINX pueden impedir que la cabecera Authorization en las solicitudes API llegue a Zabbix. Esto puede causar problemas de autenticación al usar la API de Zabbix o servicios de inicio de sesión único (SSO), como SAML con Okta.

Para solucionar esto, actualice la configuración de su servidor web.

Para **Apache**, si lo está utilizando como proxy inverso (configuración no-CGI), agregue la siguiente directiva a `/etc/httpd/conf/httpd.con` (en sistemas basados en RHEL) o `/etc/apache2/apache2.conf` (en Debian/Ubuntu):

```
SetEnvIfNoCase ^Authorization$ "(.+)" HTTP_AUTHORIZATION=$1
```

Si Apache ejecuta directamente scripts para manejar solicitudes (por ejemplo, usando mod_cgi), agregue la siguiente directiva en su lugar:

```
CGIPassAuth On
```

En cambio, **NGINX** maneja la cabecera Authorization automáticamente. Sin embargo, si NGINX está actuando como proxy inverso, puede reenviar explícitamente la cabecera Authorization agregando las siguientes directivas a `/etc/nginx/nginx.conf` (para la ubicación de su interfaz web de Zabbix):

```
...
location / {
...
    proxy_set_header Authorization $http_authorization;
    proxy_pass http://backend_server;
...
}
```

Después de actualizar la configuración, reinicie su servidor web.

Para más detalles, consulte:

- ZBX-22952
- Apache 2.4 + PHP-FPM y cabeceras Authorization
- Directivas SetEnvIfNoCase y CGIPassAuth
- NGINX Reverse Proxy

Chromium para el servicio web de Zabbix en Ubuntu 20

Aunque en la mayoría de los casos, el servicio web de Zabbix puede ejecutarse con Chromium, en Ubuntu 20.04 el uso de Chromium provoca el siguiente error:

```
Cannot fetch data: chrome failed to start:cmd_run.go:994:
WARNING: cannot create user data directory: cannot create
"/var/lib/zabbix/snap/chromium/1564": mkdir /var/lib/zabbix: permission denied
Sorry, home directories outside of /home are not currently supported. See https://forum.snapcraft.io/t/112
```

Este error ocurre porque `/var/lib/zabbix` se utiliza como directorio home del usuario 'zabbix'.

Códigos de error personalizados de MySQL

Cuando Zabbix detecta que la base de datos de backend no es accesible, envía una notificación y continúa intentando conectarse. Para ciertos motores de base de datos, se reconocen códigos de error específicos. En MySQL, estos códigos de error reconocidos incluyen:

- CR_CONN_HOST_ERROR
- CR_SERVER_GONE_ERROR
- CR_CONNECTION_ERROR
- CR_SERVER_LOST
- CR_UNKNOWN_HOST
- ER_SERVER_SHUTDOWN
- ER_ACCESS_DENIED_ERROR
- ER_ILLEGAL_GRANT_FOR_TABLE
- ER_TABLEACCESS_DENIED_ERROR
- ER_UNKNOWN_ERROR

Además, al usar Zabbix con una instalación de MySQL en Azure, puede aparecer el mensaje de error genérico *[9002] Se produjeron algunos errores* en los registros de Zabbix. Este mensaje es enviado por la base de datos al servidor o proxy de Zabbix. Para determinar la causa del error, consulte los registros de Azure.

Expresiones regulares no válidas después de cambiar a PCRE2

En Zabbix 6.0 se ha añadido compatibilidad con PCRE2. Aunque PCRE sigue siendo compatible, los paquetes de instalación de Zabbix para RHEL 7 y versiones posteriores, SLES (todas las versiones), Debian 9 y versiones posteriores, Ubuntu 16.04 y versiones posteriores se han actualizado para usar PCRE2. Si bien ofrece muchas ventajas, el cambio a PCRE2 puede hacer que ciertas expresiones regulares PCRE existentes dejen de ser válidas o se comporten de forma diferente. En particular, esto afecta al patrón *[\w-\.]*. Para que esta expresión regular vuelva a ser válida sin afectar a la semántica, cambie la expresión a *[-\w\.]* . Esto ocurre debido a que PCRE2 trata el signo de guion como un delimitador, creando un rango dentro de una clase de caracteres.

Error del widget Geomap

Es posible que los mapas del widget Geomap no se carguen correctamente si ha actualizado desde una versión anterior de Zabbix con NGINX y no cambió al nuevo archivo de configuración de NGINX durante la actualización.

Para solucionar el problema, puede descartar el archivo de configuración antiguo, usar el archivo de configuración del paquete de la versión actual y reconfigurarlo como se describe en las instrucciones de descarga en la sección *e. Configure PHP for Zabbix frontend*.

Como alternativa, puede editar manualmente un archivo de configuración de NGINX existente (normalmente, `/etc/zabbix/nginx.conf`). Para ello, abra el archivo y localice el siguiente bloque:

```
location ~ /(api\/|conf[^\.]|include|locale|vendor) {
        deny            all;
        return          404;
}
```

Luego, reemplace este bloque por:

```
location ~ /(api\/|conf[^\.]|include|locale) {
        deny            all;
        return          404;
}

location /vendor {
        deny            all;
        return          404;
}
```

Preprocesamiento: las variables globales no son seguras

El JavaScript de preprocesamiento se ejecuta por solicitud, pero las asignaciones a identificadores no declarados (por ejemplo, `secreto = valor`) crean variables globales implícitas que pueden persistir más allá de la ejecución actual. Almacenar datos sensibles (tokens, contraseñas, etc.) en variables globales implícitas aumenta el riesgo de exposición accidental o reutilización por ejecuciones de preprocesamiento posteriores u otras integraciones que se ejecuten en el mismo entorno.

No confíe en las variables globales implícitas. Declare siempre las variables con `var` o `const`, y evite adjuntar secretos a objetos globales (por ejemplo, `globalThis` o `window`). No existe una forma compatible de sobrescribir objetos globales integrados desde el preprocesamiento.

Ejemplo seguro:

```
var apiToken = payload.token;
var count = 1;
return JSON.stringify({ token: apiToken, calls: count });
```

Grupos de procesadores en Windows

La documentación de Microsoft indica que los sistemas con menos de 64 procesadores lógicos siempre tienen un único grupo de procesadores, el Grupo 0. Sin embargo, los usuarios de Zabbix han informado de un error poco frecuente ZBX-20260, en el que existen dos grupos de procesadores en sistemas con 64 o menos procesadores lógicos. Esto resultaba en que los contadores de rendimiento "\Processor(n)" solo estaban disponibles para uno de los dos grupos de procesadores. La causa raíz real de este error no se conoce. Sin embargo, un caso similar fue descrito en stackoverflow.com, y la causa raíz allí estaba en la interoperabilidad entre la BIOS y Windows.

Límites de filtrado con intercalaciones utf8mb4

Los filtros (por ejemplo, en *Recopilación de datos* > *Mantenimiento*) pueden no funcionar correctamente cuando se aplican a entidades que contienen ciertos caracteres Unicode (por ejemplo, 𝓒, 𝔈). Este problema surge debido a la forma en que la intercalación predeterminada utf8mb4_bin para las bases de datos MySQL o MariaDB maneja la ordenación y comparación de caracteres Unicode.

Para abordar esta limitación, los usuarios pueden cambiar la intercalación de las columnas de la base de datos a alternativas como utf8mb4_0900_bin, utf8mb4_0900_ai_ci o utf8mb4_unicode_520_ci. Sin embargo, tenga en cuenta que cambiar la intercalación puede causar un comportamiento inesperado en el manejo de espacios vacíos, así como en la ordenación y el filtrado de otros caracteres.

Para obtener más información sobre cómo cambiar las intercalaciones, consulte la documentación de MySQL o la documentación de MariaDB. Para obtener detalles sobre las diferencias de intercalación, consulte Conjuntos de caracteres Unicode en la documentación de MySQL.

Generación de perfiles del uso excesivo de memoria con tcmalloc

Si sospecha que su instalación de Zabbix está usando demasiada memoria, puede usar la función de perfilado de memoria de tcmalloc para investigar el consumo de memoria de Zabbix server/proxy.

1. Al instalar Zabbix desde el código fuente, configure indicadores adicionales:

```
export CFLAGS="-std=gnu99 -g -O0"
```

El indicador `-std=gnu99` es necesario para compilar Zabbix server, Zabbix proxy o Zabbix agent. El indicador `-g` agrega información adicional de depuración, mientras que `-O0` deshabilita las optimizaciones, que pueden interferir con el perfilado de tcmalloc.

2. Establezca las siguientes variables de entorno antes de iniciar Zabbix server. Estas variables indican a tcmalloc cómo rastrear e informar el uso de memoria:

```
LD_PRELOAD="/usr/lib/aarch64-linux-gnu/libtcmalloc.so" \
HEAPPROFILE=./heap_profile \
HEAP_PROFILE_ALLOCATION_INTERVAL=0 \
HEAP_PROFILE_INUSE_INTERVAL=4294967296 \
HEAPPROFILESIGNAL=5 \
MALLOCSTATS=1 \
./sbin/zabbix_server -f -c /etc/zabbix/zabbix_server.conf
```

3. Genere un volcado del perfil enviando la señal 5 al proceso de destino. Reemplace 1234 con el ID real del proceso (PID):

```
kill -5 1234
```

4. Imprima el perfil generado:

```
pprof-symbolize -text ./sbin/zabbix_server ./heap_profile.0001.heap

Using local file ./sbin/zabbix_server.
Using local file ./heap_profile.0001.heap.
Total: 1078.1 MB
  1076.8  99.9%  99.9%   1076.8  99.9% zbx_malloc2
     1.0   0.1% 100.0%      1.0   0.1% __GI___strdup
     0.2   0.0% 100.0%      0.2   0.0% CRYPTO_zalloc@@OPENSSL_3.0.0
     0.1   0.0% 100.0%      0.1   0.0% OPENSSL_LH_insert@@OPENSSL_3.0.0
     0.0   0.0% 100.0%      0.0   0.0% zbx_realloc2
     0.0   0.0% 100.0%      0.1   0.0% PKCS7_decrypt@@OPENSSL_3.0.0
     0.0   0.0% 100.0%      0.0   0.0% find_best_tree_node
     0.0   0.0% 100.0%      0.0   0.0% CRYPTO_strndup@@OPENSSL_3.0.0
     ...
```

```
     0.0   0.0% 100.0%      0.0   0.0% preprocessing_flush_value
     0.0   0.0% 100.0%   1074.0  99.6% preprocessor_add_request
```

En este ejemplo, zbx_malloc2 es responsable de casi todas las asignaciones de memoria.

Véase también:

- ZBX-25050 y ZBX-25584 para los informes de problemas relacionados.
- GCC Option Summary sobre opciones de compilación (`-std=gnu99`, `-g`, `-O0`, etc.).
- La documentación de Gperftools Heap Profiler sobre variables de entorno para el perfilado de tcmalloc.

Replicación de grupos de MySQL 8.0 en modo multi-primary

Al utilizar la replicación de grupos de MySQL 8.0 en modo multi-primary, puede encontrar un error durante la confirmación de transacciones similar al siguiente:

```
1531697:20250128:064734.697 query [txnlev:1] [update alerts set status=1,retries=0,error='' where alertid=
1531697:20250128:064734.713 query [txnlev:1] [commit;]
1531697:20250128:064734.753 [Z3005] query failed: [3101] Plugin instructed the server to rollback the curr
```

Este error parece deberse a problemas con las operaciones de rollback que involucran restricciones de clave foránea.

Vea también:

- ZBX-26060 para el informe del problema relacionado.
- MySQL Bug #96758 "Rollbacks with Foreign Keys on single node" para el problema original.

Inicio de sesión y systemd

Recomendamos crear un usuario *zabbix* como usuario del sistema, es decir, sin capacidad para iniciar sesión. Algunos usuarios ignoran esta recomendación y utilizan la misma cuenta para iniciar sesión (por ejemplo, mediante SSH) en el host donde se ejecuta Zabbix. Esto podría provocar que el daemon de Zabbix se bloquee al cerrar sesión. En este caso, verá algo similar a lo siguiente en el registro del server de Zabbix:

```
zabbix_server [27730]: [file:'selfmon.c',line:375] lock failed: [22] Invalid argument
zabbix_server [27716]: [file:'dbconfig.c',line:5266] lock failed: [22] Invalid argument
zabbix_server [27706]: [file:'log.c',line:238] lock failed: [22] Invalid argument
```

y en el registro del agent de Zabbix:

```
zabbix_agentd [27796]: [file:'log.c',line:238] lock failed: [22] Invalid argument
```

Esto ocurre debido a la configuración predeterminada de systemd `RemoveIPC=yes`, configurada en `/etc/systemd/logind.conf`. Cuando cierra sesión en el sistema, se eliminan los semáforos creados previamente por Zabbix, lo que provoca el bloqueo.

Una cita de la documentación de systemd:

```
RemoveIPC=

Controls whether System V and POSIX IPC objects belonging to the user shall be removed when the
user fully logs out. Takes a boolean argument. If enabled, the user may not consume IPC resources
after the last of the user's sessions terminated. This covers System V semaphores, shared memory
and message queues, as well as POSIX shared memory and message queues. Note that IPC objects of the
root user and other system users are excluded from the effect of this setting. Defaults to "yes".
```

Hay 2 soluciones para este problema:

1. (recomendada) Deje de utilizar la cuenta *zabbix* para cualquier otra cosa que no sean los procesos de Zabbix y cree una cuenta dedicada para las demás actividades.
2. (no recomendada) Establezca `RemoveIPC=no` en `/etc/systemd/logind.conf` y reinicie el sistema. Tenga en cuenta que `RemoveIPC` es un parámetro de todo el sistema; cambiarlo afectará a todo el sistema.

Requisitos

Hardware

Memoria

Zabbix requiere tanto memoria física como en disco. La cantidad de memoria en disco necesaria depende obviamente del número de equipos y parámetros que se estén monitorizando. Si planea mantener un historial largo de los parámetros monitorizados,

debería pensar en al menos un par de gigabytes para tener suficiente espacio para almacenar el historial en la base de datos. Cada proceso demonio de Zabbix requiere varias conexiones a un servidor de base de datos. La cantidad de memoria asignada para la conexión depende de la configuración del motor de base de datos.

> **Note:**
> Cuanta más memoria física tenga, más rápido funcionará la base de datos (y por lo tanto Zabbix).

CPU

Zabbix y especialmente la base de datos de Zabbix pueden requerir recursos significativos de CPU dependiendo del número de parámetros monitorizados y del motor de base de datos elegido.

Otro hardware

Se requiere un puerto de comunicación serie y un módem GSM serie para utilizar la compatibilidad con notificaciones SMS en Zabbix. Un convertidor USB a serie también funcionará.

Ejemplos de configuración de hardware

La tabla proporciona ejemplos de configuración de hardware, suponiendo una plataforma **Linux/BSD/Unix**.

Estos son ejemplos de tamaño y configuración de hardware para comenzar. Cada instalación de Zabbix es única. Asegúrese de comparar el rendimiento de su sistema Zabbix en un entorno de pruebas o desarrollo, para comprender completamente sus requisitos antes de implementar la instalación de Zabbix en su entorno de producción.

Tamaño de la instalación	Métricas supervisadas[1]	Núcleos de CPU/vCPU	Memoria (GiB)	Base de datos	Amazon EC2[2]
Pequeña	1 000	2	8	MySQL Server, Percona Server, MariaDB Server, PostgreSQL	m6i.large/m6g.large
Mediana	10 000	4	16	MySQL Server, Percona Server, MariaDB Server, PostgreSQL	m6i.xlarge/m6g.xlarge
Grande	100 000	16	64	MySQL Server, Percona Server, MariaDB Server, PostgreSQL	m6i.4xlarge/m6g.4xlarge
Muy grande	1 000 000	32	96	MySQL Server, Percona Server, MariaDB Server, PostgreSQL	m6i.8xlarge/m6g.8xlarge

[1] Para calcular el tamaño, 1 métrica supervisada se representa aproximadamente mediante la fórmula 1 métrica = 1 item + 1 trigger + 1 gráfico. Esta es solo una regla de estimación, no una fórmula para calcular las métricas supervisadas.
 [2] Ejemplo con instancias EC2 de uso general de Amazon, utilizando la arquitectura ARM64 o x86_64. Durante la evaluación y las pruebas de la instalación de Zabbix, antes de instalarla en el entorno de producción, se debe seleccionar un tipo de instancia adecuado, como una instancia optimizada para Compute/Memory/Storage.

> **Note:**
> La configuración real depende en gran medida del número de items activos y de las frecuencias de actualización (consulte la sección tamaño de la base de datos de esta página para obtener más información). Se recomienda encarecidamente ejecutar la base de datos en un server independiente para las instalaciones grandes.

Plataformas compatibles

Debido a los requisitos de seguridad y la naturaleza crítica del servidor de monitorización, UNIX es el único sistema operativo que puede ofrecer de manera constante el rendimiento, la tolerancia a fallos y la resiliencia necesarios. Zabbix funciona en versiones líderes del mercado.

Los componentes de Zabbix están disponibles y probados para las siguientes plataformas:

Plataforma	Servidor	Agente	Agente 2	Comentarios
Linux	x	x	x	
Windows	-	x	x	El agente Zabbix es compatible con todas las versiones de escritorio y servidor desde Windows XP (64 bits)/Server 2003. El agente Zabbix 2 es compatible con todas las versiones de escritorio y servidor desde Windows 10 (32 bits)/Server 2016, ya que solo se compila con una versión de Go compatible para evitar vulnerabilidades de seguridad críticas. Desde Go 1.21, las versiones mínimas requeridas de Windows se han elevado, haciendo que Windows 10/Server 2016 sea la versión mínima para el agente Zabbix 2.
macOS	x	x	-	
IBM AIX	x	x	-	El agente Zabbix no funciona en plataformas AIX por debajo de las versiones 6.1 TL07 / 7.1 TL01.
FreeBSD	x	x	-	
OpenBSD	x	x	-	
Solaris	x	x	-	
NetBSD	x	x	-	
HP-UX	x	x	-	

Note:
El servidor/agente Zabbix puede funcionar en otros sistemas operativos tipo Unix.

Attention:
Zabbix desactiva los volcados de memoria si se compila con cifrado y no se inicia si el sistema no permite desactivar los volcados de memoria.

Software requerido

Zabbix está construido sobre servidores web modernos, motores de bases de datos líderes y el lenguaje de scripting PHP.

Software externo de terceros del entorno

Si se indica como obligatorio, el software o la biblioteca requeridos son estrictamente necesarios.
Los opcionales son necesarios para admitir alguna función específica.

Software	Estado obligatorio	Versiones compatibles	Comentarios
MySQL/Percona	One of	8.4.0-9.7.X	Es necesario si se utiliza MySQL (o Percona) como base de datos backend de Zabbix. Se requiere el motor InnoDB. Recomendamos utilizar la biblioteca C API (libmysqlclient) para compilar el server/proxy. Recomendamos encarecidamente utilizar únicamente versiones LTS en entornos de producción. No se recomiendan las versiones etiquetadas como ”rolling release”, ”early access” o similares.
MariaDB		10.11.00-12.3.X	Se requiere el motor InnoDB. Recomendamos utilizar la biblioteca MariaDB Connector/C para compilar el server/proxy. Consulte también: Posibles interbloqueos con MariaDB y Acceso a elementos de la interfaz de usuario con MariaDB 10.5.1-10.5.9. Recomendamos encarecidamente utilizar únicamente versiones LTS en entornos de producción. No se recomiendan las versiones etiquetadas como ”rolling release”, ”early access” o similares.

Software	Estado obligatorio	Versiones compatibles	Comentarios
PostgreSQL		15.0-18.X	Es necesario si se utiliza PostgreSQL como base de datos backend de Zabbix. Según el tamaño de la instalación, puede ser necesario aumentar la propiedad de configuración *work_mem* de PostgreSQL (el valor predeterminado es 4 MB), de modo que la cantidad de memoria utilizada por la base de datos para una operación concreta sea suficiente y la ejecución de la consulta no tarde demasiado.
TimescaleDB para PostgreSQL		2.20.X-2.29.X	Es necesario si se utiliza TimescaleDB como extensión de una base de datos PostgreSQL. Asegúrese de instalar TimescaleDB Community Edition, que admite la compresión. Consulte la documentación de TimescaleDB para obtener información detallada sobre la compatibilidad entre las versiones de PostgreSQL y TimescaleDB.
Elasticsearch		7.X	Es necesario si se utiliza Elasticsearch como base de datos backend de Zabbix. Elasticsearch solo se puede utilizar para almacenar datos de historial y solo es compatible con Zabbix servers. La compatibilidad con Elasticsearch es actualmente experimental. Consulte también el software requerido para server/proxy.
ClickHouse		26.3.0.0-26.9.X.X	Es necesario si se utiliza ClickHouse como base de datos backend de Zabbix para almacenar datos de historial (compatible únicamente con Zabbix servers) o como fuente de datos de telemetría. Para utilizar ClickHouse como fuente de datos de telemetría, se requiere ClickHouse 26.8.17.4 o posterior. La hora del sistema del Zabbix server y del server de ClickHouse también debe estar sincronizada; de lo contrario, las consultas de telemetría pueden omitir datos o devolver los mismos datos dos veces.
SQLite	Opcional	3.3.5-3.53.X	SQLite solo es compatible con Zabbix proxies. Es necesario si se utiliza SQLite como base de datos del Zabbix proxy.
smartmontools		7.1 o posterior	Es necesario para Zabbix agent 2.
who			Es necesario para el plugin de recuento de usuarios.
dpkg			Es necesario para el plugin system.sw.packages.
pkgtool			Es necesario para el plugin system.sw.packages.
rpm			Es necesario para el plugin system.sw.packages.
pacman			Es necesario para el plugin system.sw.packages.
q applets			`qlist` y `qsize`, como parte de q applets, son necesarios para el plugin system.sw.packages en Gentoo Linux.

Note:
Aunque Zabbix puede funcionar con bases de datos disponibles en los sistemas operativos, para obtener la mejor experiencia recomendamos utilizar bases de datos instaladas desde los repositorios oficiales del desarrollador de la base de datos.

Frontend

El ancho mínimo de pantalla compatible con el frontend de Zabbix es de 1200px.

Si se indica como obligatorio, el software o la biblioteca requeridos son estrictamente necesarios. Los opcionales son necesarios para admitir alguna función específica.

Software	Estado obligatorio	Versiones compatibles	Comentarios
PHP	Sí	8.2.0 - 8.5.X	

Software	Estado obligatorio	Versiones compatibles	Comentarios
Apache	Uno de	2.4 o posterior	
Nginx		1.20 o posterior	
MySQL	Uno de	Consulte Software externo circundante de terceros	
PostgreSQL			
Extensiones de PHP			
mysqli	Sí		Necesario si MySQL se utiliza como base de datos backend de Zabbix.
pgsql			Necesario si PostgreSQL se utiliza como base de datos backend de Zabbix.
bcmath			php-bcmath (*--enable-bcmath*)
mbstring			php-mbstring (*--enable-mbstring*)
sockets			php-net-socket (*--enable-sockets*); necesario para admitir scripts de usuario.
gd		2.0.28 o posterior	php-gd (si el distribuidor lo proporciona como un paquete independiente); la extensión GD de PHP debe admitir imágenes PNG (*--with-png-dir*), imágenes JPEG (*--with-jpeg-dir*) y FreeType 2 (*--with-freetype-dir*). Puede ser necesaria la versión 2.3.0 o posterior para evitar un posible solapamiento de texto en los gráficos en algunos idiomas del frontend.
libxml		2.6.15 o posterior	php-xml (si el distribuidor lo proporciona como un paquete independiente)
xmlwriter			php-xmlwriter (si el distribuidor lo proporciona como un paquete independiente)
xmlreader			php-xmlreader (si el distribuidor lo proporciona como un paquete independiente)
ctype			php-ctype (*--enable-ctype*)
session			php-session (si el distribuidor lo proporciona como un paquete independiente)
ldap	No		php-ldap; necesario para la autenticación LDAP.
openssl			php-openssl; necesario para la autenticación SAML.
gettext			php-gettext (*--with-gettext*); necesario para las traducciones.
cURL		7.19.4 o posterior	php-curl; necesario para Duo Universal Prompt MFA, la autenticación SMTP y para probar el origen de datos de telemetría.

Bibliotecas de frontend de terceros suministradas con Zabbix:

Biblioteca	Estado obligatorio	Versión suministrada	Comentarios
Biblioteca JavaScript jQuery	Sí	3.6.0	Biblioteca JavaScript que simplifica el proceso de desarrollo entre navegadores.
jQuery UI		1.12.1	Conjunto de interacciones de interfaz de usuario, efectos, widgets y temas creado sobre jQuery.
SAML PHP Toolkit		4.3.1	Kit de herramientas de PHP que añade compatibilidad con la autenticación SAML 2.0 para poder iniciar sesión en Zabbix.
Duo Universal PHP library		1.1.2	Biblioteca de PHP que añade compatibilidad con la autenticación multifactor Duo Universal Prompt para Zabbix.
Symfony Yaml Component		5.1.0	Añade compatibilidad para exportar e importar elementos de configuración de Zabbix en formato YAML.

Note:
Zabbix también puede funcionar con versiones anteriores de Apache, MySQL y PostgreSQL.

Attention:
Para fuentes distintas de la fuente predeterminada DejaVu, podría ser necesaria la función de PHP imagerotate. Si falta, estas fuentes podrían representarse incorrectamente al mostrar un gráfico. Esta función solo está disponible si PHP se compila con GD incluido, lo que no ocurre en Debian ni en otras distribuciones.

Bibliotecas de terceros utilizadas para escribir y depurar código del frontend de Zabbix:

Biblioteca	Estado obligatorio	Versión mínima	Descripción
Composer	No	2.4.1	Administrador de paquetes a nivel de aplicación para PHP que proporciona un formato estándar para gestionar las dependencias del software PHP y las bibliotecas necesarias.
PHPUnit		8.5.29	Framework de pruebas unitarias de PHP para probar el frontend de Zabbix.
SASS		3.4.22	Lenguaje de scripting de preprocesamiento que se interpreta y compila en hojas de estilo en cascada (CSS).

Navegador web en el lado del cliente

Las cookies y JavaScript deben estar habilitados.

Se admiten las últimas versiones estables de Google Chrome, Mozilla Firefox, Microsoft Edge, Apple Safari y Opera.

Warning:
Se implementa la política de mismo origen para IFrames, lo que significa que Zabbix no puede colocarse en marcos en un dominio diferente.

 Aun así, las páginas colocadas en un marco de Zabbix tendrán acceso al frontend de Zabbix (a través de JavaScript) si la página que se coloca en el marco y el frontend de Zabbix están en el mismo dominio. Una página como `http://secure-zabbix.com/cms/page.html`, si se coloca en paneles en `http://secure-zabbix.com/zabbix/`, tendrá acceso completo por JS a Zabbix.

Server/proxy

Si se indica como obligatorio, el software o la biblioteca requeridos son estrictamente necesarios. Los elementos opcionales son necesarios para admitir alguna función específica.

Requisito	Estado obligatorio	Descripción
libpcre2	Sí	La biblioteca PCRE2 es necesaria para admitir expresiones regulares compatibles con Perl (PCRE). Se admite PCRE2 v10.x.
libevent		Necesario para la comunicación entre procesos. Versión 2.0.10 o posterior.
libevent-pthreads		Necesario para la comunicación entre procesos.
libpthread		Necesario para admitir mutex y bloqueos de lectura y escritura (puede formar parte de libc).
libresolv		Necesario para la resolución DNS (puede formar parte de libc).
libiconv		Necesario para la codificación y conversión de formatos de texto (puede formar parte de libc). Obligatorio para Zabbix server en Linux.
libz		Necesario para admitir la compresión.
libm		Biblioteca matemática. Necesaria únicamente para Zabbix server.
libmysqlclient	Uno de	Necesario si se utiliza MySQL.
libmariadb		Necesario si se utiliza MariaDB.
libpq5		Necesario si se utiliza PostgreSQL; la versión de *libpq5* debe coincidir con la versión de la base de datos PostgreSQL utilizada o ser posterior.
libsqlite3		Necesario si se utiliza SQLite. Necesario únicamente para Zabbix proxy.
libOpenIPMI	No	Necesario para admitir IPMI. Necesario únicamente para Zabbix server.
libssh2 o *libssh*		Necesario para las comprobaciones SSH. Versión 1.8.0 o posterior (libssh2); 0.9.0 o posterior (libssh).

Requisito	Estado obligatorio	Descripción
libcurl		Necesario para las siguientes funciones: - Elementos de monitorización web, monitorización de VMware y HTTP agent (para todos: versión 7.19.1 o posterior); - Elementos web.page.* de Zabbix agent (versión 7.19.1 o posterior; consulte también los requisitos de agent); - Autenticación SMTP (básica: versión 7.20.0 o posterior; OAuth: versión 7.33 o posterior; consulte también los requisitos de frontend); - Elasticsearch (versión 7.28.0 o posterior); - Elementos de consulta de telemetría (versión 7.28.0 o posterior). Se recomienda utilizar la versión 7.28.0 o posterior para todas las funciones. Para utilizar las funciones actualizadas de cURL con los elementos `web.page.*`, reinicie Zabbix server/proxy. Para la autenticación SMTP, utilice el paquete `libcurl-full` en tiempo de ejecución.
libxml2		Necesario para la monitorización de VMware y el preprocesamiento XML XPath.
net-snmp		Necesario para admitir SNMP. Versión 5.3.0 o posterior. La compatibilidad con protocolos de cifrado robustos (AES192/AES192C, AES256/AES256C) está disponible a partir de la biblioteca net-snmp 5.8; en sistemas basados en RHEL 8 o posterior, se recomienda utilizar net-snmp 5.8.15 o posterior.
libunixodbc		Necesario para la monitorización de bases de datos.
libgnutls o *libopenssl*		Necesario al utilizar cifrado. Versiones mínimas: *libgnutls* - 3.1.18, *libopenssl* - 1.0.1
libldap		Necesario para admitir LDAP.
fping		Necesario para los elementos de ping ICMP.
c-ares		Necesario para la resolución DNS asíncrona si Zabbix está configurado con la opción `--with-ares`. De lo contrario, se utilizará *libevent*. Versión mínima: 1.16.0. La función de almacenamiento en caché de consultas DNS y conmutación por error (la caché de consultas está habilitada de forma predeterminada) requiere c-ares 1.26.0 o posterior. En los sistemas donde c-ares lo proporciona el paquete del sistema operativo, asegúrese de que la biblioteca de tiempo de ejecución instalada exponga las funciones del resolvedor necesarias para el almacenamiento en caché y la conmutación por error de DNS.
g++		Compilador de C++. Necesario para compilar Zabbix proxy con la opción `--with-apm`.
gRPC		Necesario para admitir APM en Zabbix proxy. Bibliotecas necesarias: *grpc++*, *grpc*, *gpr*. La utilidad `grpc_cpp_plugin` también es necesaria para compilar Zabbix proxy. La versión necesaria está disponible en los repositorios de paquetes estándar de estos sistemas operativos: - Ubuntu 23.04 o posterior; - Debian 12 (Bookworm) o posterior; - Alpine Linux 3.15 o posterior; - RHEL, Rocky Linux, CentOS Stream 9 o posterior (solo desde el repositorio EPEL); - Fedora X o posterior. En versiones anteriores de los sistemas operativos, *gRPC* debe compilarse desde el código fuente.
protobuf		Necesario para admitir APM en Zabbix proxy. El compilador `protoc` también es necesario para compilar Zabbix proxy.
Abseil		Necesario para admitir APM en Zabbix proxy. Bibliotecas necesarias: *absl_synchronization*, *absl_base*. Normalmente se instala junto con *gRPC*.

Agent

Requirement	Mandatory status	Description
libpcre2	Yes	Se requiere la biblioteca PCRE2 para la compatibilidad con Perl Compatible Regular Expression (PCRE). Se admite PCRE2 v10.x. Requerida para la monitorización de logs. También es necesaria en Windows.
libpthread		Requerida para la compatibilidad con mutex y bloqueos de lectura/escritura (puede formar parte de libc). No es necesaria en Windows.
libresolv		Requerida para la resolución DNS (puede formar parte de libc). No es necesaria en Windows.
libiconv		Requerida para la codificación de texto y la conversión de formato a UTF-8 en items de log, contenido de archivos, expresiones regulares de archivos y items regmatch (puede formar parte de libc). No es necesaria en Windows.
libgnutls or *libopenssl*	No	Requerida si se usa encryption. Versiones mínimas: *libgnutls* - 3.1.18, *libopenssl* - 1.0.1 En Microsoft Windows se requiere OpenSSL 1.1.1 o posterior.
libldap		Requerida si se usa LDAP. No es compatible con Windows.
libcurl		Requerida para la compatibilidad ampliada con los items web.page.* del agent de Zabbix. Sin libcurl, está disponible la funcionalidad básica (por ejemplo, web.page.get[http://example.com]). Con libcurl, el agent admite funciones adicionales, como URL HTTP con credenciales (por ejemplo, http://user:password@example.com) y URL HTTPS. Se requiere la versión 7.19.1 o superior (se recomienda 7.28.0 o superior). Para usar las funciones actualizadas de cURL, reinicie el agent de Zabbix.
libmodbus		Solo es necesaria si se usa la monitorización de Modbus. Versión 3.0 o superior.

Agent 2

Requirement	Mandatory status	Description
Go	Yes	Se requiere para compilar Zabbix agent 2 y sus plugins desde el código fuente. Se admite Go 1.25 o posterior. Consulte go.dev para obtener instrucciones de instalación. Las bibliotecas de Go utilizadas por Zabbix agent 2 y sus plugins se enumeran en el repositorio Git de Zabbix (las bibliotecas marcadas como `indirect` en el repositorio son dependencias de otras bibliotecas requeridas): - Zabbix agent 2 - Plugin support - PostgreSQL - MongoDB - MSSQL - Ember+ - NVIDIA GPU - Example plugin
libpcre2	Yes	Se requiere la biblioteca PCRE2 para la compatibilidad con Expresiones regulares compatibles con Perl (PCRE). Se admite PCRE2 v10.x. Es necesaria para la supervisión de registros. También es necesaria en Windows.
libopenssl	No	Se requiere al usar cifrado. En plataformas UNIX se requiere OpenSSL 1.0.1 o posterior. La biblioteca OpenSSL debe tener habilitado el soporte PSK. LibreSSL no es compatible. En sistemas Microsoft Windows se requiere OpenSSL 1.1.1 o posterior.

Servicio web

La última versión estable de Google Chrome es compatible para generar informes programados utilizando el servicio web de Zabbix.

La versión de Go requerida para compilar el servicio web coincide con la utilizada para el agente Zabbix 2.

Java gateway

Si obtuvo Zabbix del repositorio de código fuente o de un archivo, las dependencias necesarias ya están incluidas en el árbol de código fuente.

Si obtuvo Zabbix del paquete de su distribución, las dependencias necesarias ya las proporciona el sistema de empaquetado.

En ambos casos, el software está listo para usarse y no es necesario realizar descargas adicionales.

Sin embargo, si desea proporcionar sus propias versiones de estas dependencias (por ejemplo, si está preparando un paquete para alguna distribución de Linux), a continuación se muestra la lista de versiones de bibliotecas con las que se sabe que Java gateway funciona. Es posible que Zabbix también funcione con otras versiones de estas bibliotecas.

La siguiente tabla muestra los archivos JAR incluidos actualmente con Java gateway en el código original:

Biblioteca	Estado obligatorio	Versión incluida	Comentarios
android-json	Sí	4.3r1	JSON (JavaScript Object Notation) es un formato ligero de intercambio de datos. Esta es la implementación de Android compatible con `org.json`, extraída del SDK de Android.
logback-classic		1.5.25	
logback-core		1.5.25	
slf4j-api		2.0.16	

Java gateway puede compilarse utilizando Oracle Java o el OpenJDK de código abierto (versión 8 o posterior). Los paquetes proporcionados por Zabbix se compilan utilizando OpenJDK. La siguiente tabla muestra los paquetes de OpenJDK utilizados para compilar los paquetes de Zabbix por distribución:

Distribución	Paquete de OpenJDK
AlmaLinux 10	`java-21-openjdk-devel` (amd64, arm64: 21.0.12.0.8-1.2)
AlmaLinux 9	`java-17-openjdk-devel` (amd64, arm64: 17.0.15.0.6-2)
AlmaLinux 8	`java-1.8.0-openjdk-devel` (amd64: 1.8.0.332.b09-2; arm64: 1.8.0.382.b05-2)
Amazon Linux 2023	`java-22-amazon-corretto-devel` (amd64, arm64: 22.0.2+9-1)
CentOS Stream 10	`java-21-openjdk-devel` (amd64, arm64: 21.0.11.0.10-2)
CentOS Stream 9	`java-17-openjdk-devel` (amd64, arm64: 17.0.15.0.6-2)
CentOS Stream 8	`java-1.8.0-openjdk-devel` (amd64, arm64: 1.8.0.362.b08-3)
CentOS 7	`java-1.8.0-openjdk-devel` (amd64: 1.8.0.282.b08-1)
Debian 13	`default-jdk-headless` (amd64, arm64: 2:1.21-76)
Debian 12	`default-jdk-headless` (amd64, arm64: 2:1.17-74)
Debian 11	`default-jdk-headless` (amd64: 2:1.11-72)
OpenSUSE Leap 16	`java-21-openjdk-devel` (amd64, arm64: 21.0.11.0-160000.2.1)
OpenSUSE Leap 15	`java-17-openjdk-devel` (amd64: 17.0.5.0-150400.3.9.3; arm64: 17.0.8.0-150400.3.27.1)
Oracle Linux 10	`java-21-openjdk-devel` (amd64, arm64: 21.0.11.0.10-2.0.1)
Oracle Linux 9	`java-17-openjdk-devel` (amd64, arm64: 17.0.15.0.6-2.0.1)
Oracle Linux 8	`java-1.8.0-openjdk-devel` (amd64: 1.8.0.372.b07-4.0.1); `java-11-openjdk-devel` (arm64: 11.0.20.0.8-3.0.1)
Oracle Linux 7	`java-1.8.0-openjdk-devel` (amd64: 1.8.0.282.b08-1)
Raspberry Pi OS 13	`default-jdk-headless` (arm64, armhf: 2:1.21-76)
Raspberry Pi OS 12	`default-jdk-headless` (arm64, armhf: 2:1.17-74)
Raspberry Pi OS 11	`default-jdk-headless` (arm64: 2:1.11-72; armhf: 2:1.11-72+b4)
RHEL 10	`java-21-openjdk-devel` (amd64, arm64: 21.0.12.0.8-1.2)
RHEL 9	`java-17-openjdk-devel` (amd64, arm64: 17.0.15.0.6-2)
RHEL 8	`java-1.8.0-openjdk-devel` (amd64: 1.8.0.372.b07-4; arm64: 1.8.0.382.b05-2)
RHEL 7	`java-1.8.0-openjdk-devel` (amd64: 1.8.0.282.b08-1)
Rocky Linux 10	`java-21-openjdk-devel` (amd64, arm64: 21.0.12.0.8-1.2)
Rocky Linux 9	`java-17-openjdk-devel` (amd64, arm64: 17.0.15.0.6-2)
Rocky Linux 8	`java-1.8.0-openjdk-devel` (amd64, arm64: 1.8.0.372.b07-4; arm64: 1.8.0.382.b05-2)
SLES 16	`java-21-openjdk-devel` (amd64, arm64: 21.0.11.0-160000.1.1)
SLES 15	`java-17-openjdk-devel` (amd64: 17.0.5.0-150400.3.9.3; arm64: 17.0.8.0-150400.3.27.1)
Ubuntu 26.04	`default-jdk-headless` (amd64, arm64: 2:1.25-77)
Ubuntu 24.04	`default-jdk-headless` (amd64, arm64: 2:1.21-75+exp1)

Distribución	Paquete de OpenJDK
Ubuntu 22.04	`default-jdk-headless` (amd64, arm64: 2:1.11-72build2)
Ubuntu 20.04	`default-jdk-headless` (amd64, arm64: 2:1.11-72)

Adaptador de bridge

El adaptador de bridge de Zabbix no tiene requisitos especiales.

Puertos predeterminados

La siguiente lista contiene los puertos que los componentes de Zabbix pueden utilizar en una configuración predeterminada:

Componente de Zabbix	Protocolo y puerto	Conexión iniciada por
Zabbix agent	TCP 10050	Zabbix server o Zabbix proxy (solo comprobaciones pasivas)
Zabbix agent 2	TCP 10050	Zabbix server o Zabbix proxy (solo comprobaciones pasivas)
Zabbix server	TCP 10051	Zabbix agents activos, Zabbix proxy activo, Zabbix sender, Zabbix frontend
Zabbix proxy	TCP 10051	Zabbix agents activos, Zabbix sender, Zabbix server (solo proxy pasivo)
Zabbix Java gateway	TCP 10052	Zabbix server o proxy
Zabbix web service	TCP 10053	Zabbix server
Zabbix MCP server	TCP 8443	Clientes MCP (agentes de IA)
Zabbix frontend	TCP 80	Navegadores web, Zabbix web service (informes programados), clientes de API
	TCP 443	
Traps SNMP (snmptrapd)	UDP 162	Dispositivos SNMP
Adaptador puente de Zabbix	TCP 10054	Zabbix server

Warning:
Abra únicamente los puertos requeridos por los componentes que realmente ejecute y restrinja cada regla del firewall a los hosts específicos que necesiten conectarse (por ejemplo, permita el puerto 10050 únicamente desde el Zabbix server o proxy, no desde cualquier dirección). No exponga directamente los puertos de Zabbix a internet.
 Antes de abrir los puertos, refuerce la instalación: habilite el cifrado entre los componentes, restrinja los hosts permitidos en los archivos de configuración de los componentes y revise las prácticas recomendadas de seguridad.

Tamaño de la base de datos

Los datos de configuración de Zabbix requieren una cantidad fija de espacio en disco y no crecen mucho.

El tamaño de la base de datos de Zabbix depende principalmente de estas variables, que definen la cantidad de datos históricos almacenados:

- Número de valores procesados por segundo

Este es el número promedio de nuevos valores que Zabbix server recibe cada segundo. Por ejemplo, si tenemos 3000 items para supervisión con una frecuencia de actualización de 60 segundos, el número de valores por segundo se calcula como 3000/60 = **50**.

Esto significa que se agregan 50 nuevos valores a la base de datos de Zabbix cada segundo.

- Configuración de housekeeper para history

Zabbix conserva los valores durante un período fijo de tiempo, normalmente varias semanas o meses. Cada nuevo valor requiere una cierta cantidad de espacio en disco para los datos y el índice.

Por lo tanto, si quisiéramos conservar 30 días de history y recibiéramos 50 valores por segundo, el número total de valores sería aproximadamente (**30***24*3600)* **50** = 129.600.000, o alrededor de 130M de valores.

Según el motor de base de datos utilizado, el tipo de valores recibidos (flotantes, enteros, cadenas, archivos de log, etc.), el espacio en disco para almacenar un solo valor puede variar de 40 bytes a cientos de bytes. Normalmente es de alrededor de 90 bytes por valor para items numéricos[2]. En nuestro caso, esto significa que 130M de valores requerirán 130M * 90 bytes = **10.9GB** de espacio en disco.

Note:
El tamaño de los valores de items de texto/log es imposible de predecir con exactitud, pero puede esperar alrededor de 500 bytes por valor.

- Configuración de housekeeper para trends

Zabbix conserva un conjunto de valores max/min/avg/count de 1 hora para cada item en la tabla **trends**. Los datos se utilizan para tendencias y gráficos de largo período. El período de una hora no se puede personalizar.

La base de datos de Zabbix, según el tipo de base de datos, requiere alrededor de 90 bytes por cada total. Supongamos que queremos conservar datos de trends durante 5 años. Los valores para 3000 items requerirán 3000*24*365* **90** = **2.2GB** por año, o **11GB** para 5 años.

- Configuración de housekeeper para events

Cada event de Zabbix requiere aproximadamente 250 bytes de espacio en disco[1]. Es difícil estimar el número de events que Zabbix genera diariamente. En el peor de los casos, podemos asumir que Zabbix genera un event por segundo.

Para cada event recuperado, se crea un registro event_recovery. Normalmente la mayoría de los events se recuperarán, por lo que podemos asumir un registro event_recovery por event. Eso significa 80 bytes adicionales por event.

Opcionalmente, los events pueden tener tags, y cada registro de tag requiere aproximadamente 100 bytes de espacio en disco[1]. El número de tags por event (#tags) depende de la configuración. Por lo tanto, cada uno necesitará #tags * 100 bytes adicionales de espacio en disco.

Esto significa que, si queremos conservar 3 años de events, se requerirá 3*365*24*3600* (250+80+#tags*100) = **~30GB**+#tags*100B de espacio en disco[2].

Note:
[1] Más cuando hay nombres de events, tags y valores no ASCII.
 [2] Las aproximaciones de tamaño se basan en MySQL y pueden ser diferentes para otras bases de datos.

La tabla contiene fórmulas que se pueden usar para calcular el espacio en disco requerido para el sistema Zabbix:

Parameter	Formula for required disk space (in bytes)
Zabbix configuration	Fixed size. Normally 10MB or less.
History	days*(items/refresh rate)*24*3600*bytes items : number of items days : number of days to keep history refresh rate : average refresh rate of items bytes : number of bytes required to keep single value, depends on database engine, normally ~90 bytes.
Trends	days*(items/3600)*24*3600*bytes items : number of items days : number of days to keep history bytes : number of bytes required to keep single trend, depends on the database engine, normally ~90 bytes.
Events	days*events*24*3600*bytes events : number of event per second. One (1) event per second in worst-case scenario. days : number of days to keep history bytes : number of bytes required to keep single event, depends on the database engine, normally ~330 + average number of tags per event * 100 bytes.

Por lo tanto, el espacio total requerido en disco se puede calcular como:

Configuration + History + Trends + Events

El espacio en disco NO se utilizará inmediatamente después de la instalación de Zabbix. El tamaño de la base de datos crecerá y luego dejará de crecer en algún momento, lo que depende de la configuración de housekeeper.

Sincronización de tiempo

Es muy importante tener una hora del sistema precisa en el servidor donde se ejecuta Zabbix. ntpd es el demonio más popular que sincroniza la hora del host con la hora de otras máquinas. Se recomienda encarecidamente mantener la hora del sistema sincronizada en todos los sistemas donde se ejecutan los componentes de Zabbix.

Novedades y notas de actualización: 8.0

Zabbix 8.0.0 continúa a partir de Zabbix 7.4.0 e incorpora nuevas funciones y mejoras.

También en esta página:

- Novedades en los templates, incluidos los nuevos templates y los cambios en los templates existentes.
- Notas de actualización, incluidos los cambios incompatibles.

Consulte también: Cambios en la API.

Notas de actualización

Estas notas son para actualizar desde Zabbix 7.4.x a Zabbix 8.0.0.

Todas las notas se agrupan en:

- **Cambios incompatibles** - cambios que pueden romper instalaciones existentes y otra información crítica relacionada con el proceso de actualización
- **Otros** - toda la información restante que describe los cambios en la funcionalidad de Zabbix

Véase también:

- Procedimiento de actualización para toda la información relevante sobre la actualización desde versiones anteriores a Zabbix 8.0.0;
- Actualización del clúster HA para instrucciones sobre la actualización de servers en un clúster de **alta disponibilidad** (HA).

Cambios incompatibles

Otro

Novedades y notas de actualización: 8.0.x

Esta página presenta todas las nuevas funcionalidades incluidas en las versiones menores de la versión principal de Zabbix. También se incluyen los cambios en los template y las notas de actualización.

Consulte también Novedades de la versión principal.

Novedades en Zabbix 8.0.1 Esta versión aún no ha sido publicada.

Consulte las notas de actualización de esta versión.

Notas de actualización

Appliance de Zabbix

Descripción general El appliance de Zabbix permite implementar instantáneamente el server y el frontend de Zabbix, en lugar de configurarlos manualmente o reutilizar un server existente para Zabbix.

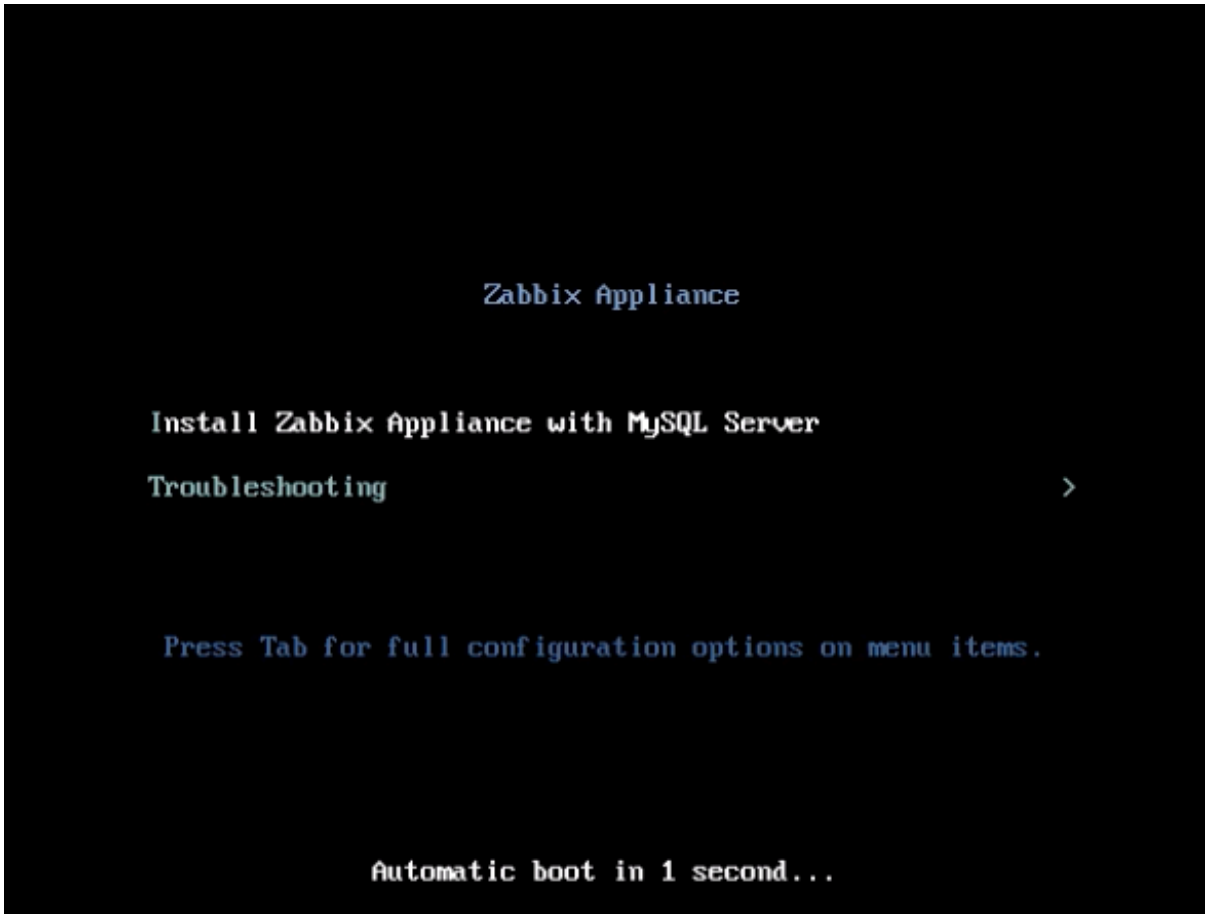
El appliance para Zabbix 8.0 se basa en derivados de Red Hat Enterprise Linux 10, como AlmaLinux o Rocky Linux, e incluye un server de Zabbix preconfigurado que se ejecuta en MySQL y el frontend que se ejecuta en un server web NGINX.

Attention:
Este appliance está diseñado para evaluar Zabbix. No se recomienda utilizarlo en entornos de producción importantes.

Las imágenes del appliance están disponibles para descargar en los siguientes formatos:

- CD/DVD de instalación (.iso)
- VMware (.vmx): consulte las notas
- Formato de virtualización abierta (.ovf)
- Microsoft Hyper-V (.vhd/.vhdx): consulte las notas
- KVM, Parallels, QEMU, memoria USB, VirtualBox, Xen (.raw): consulte las notas
- KVM, QEMU (.qcow2)

Menú de arranque del CD/DVD de instalación de Zabbix:



Inicio rápido Requisitos previos

1. Asegúrese de que el equipo anfitrión tenga recursos suficientes para cumplir con los requisitos del sistema de la máquina virtual:

- *RAM*: 4 GB
- *Espacio en disco*: al menos 8 GB deben asignarse a la máquina virtual
- *CPU*: mínimo 2 núcleos

2. Si aún no está instalado, instale el software de virtualización para arrancar la imagen del appliance (por ejemplo, VirtualBox).
3. Descargue el appliance en el formato compatible con su software de virtualización.
4. Verifique la configuración de red para asegurarse de que DHCP esté habilitado en el equipo anfitrión.

Instalación

1. Inicie la máquina virtual del appliance desde la imagen descargada. Para obtener instrucciones, consulte la documentación de su software de virtualización, por ejemplo, la documentación de VirtualBox.
2. Configure los ajustes de red de la máquina virtual para permitir el acceso desde un navegador en el host. Esto se puede lograr habilitando el *modo puente*.
3. Inicie sesión en la máquina virtual usando las credenciales predeterminadas del sistema.

4. Para recuperar la dirección IP, ejecute el siguiente comando en la máquina virtual:

```
ip addr show
```

5. Abra un navegador en el host y diríjalo a la dirección IP que el appliance ha recibido mediante DHCP.
6. Inicie sesión en Zabbix usando las credenciales predeterminadas y comience a supervisar.

Configuración Esta sección describe las configuraciones predeterminadas requeridas con frecuencia junto con las opciones de personalización disponibles.

Credenciales

Sistema

- Nombre de usuario: `root`
- Contraseña: `zabbix`

Zabbix frontend

- Nombre de usuario: `Admin`
- Contraseña: `zabbix`

Después de iniciar sesión, puede cambiar la contraseña predeterminada en la configuración del perfil de usuario o crear nuevos usuarios y eliminar el usuario predeterminado.

Base de datos

Las contraseñas de todos los usuarios de la base de datos se generan aleatoriamente durante el proceso de instalación. Los siguientes usuarios están definidos para la base de datos:

Root:

- Nombre de usuario: `root`
- Contraseña: la contraseña se almacena en el archivo `/root/.my.cnf` No es necesario introducir una contraseña con la cuenta root.

Zabbix server:

- Nombre de usuario: `zabbix_srv`
- Contraseña: la contraseña se almacena en el archivo de configuración del server

Zabbix frontend:

- Nombre de usuario: `zabbix_web`
- Contraseña: la contraseña se almacena en el archivo de configuración del frontend

Para cambiar la contraseña de un usuario de la base de datos, modifíquela en MySQL y en el archivo de configuración correspondiente.

Acceso al frontend

Se puede acceder al frontend de Zabbix en *http://<virtual machine's IP>*.

De forma predeterminada, se permite el acceso desde cualquier lugar. Para limitar el acceso, modifique `/etc/nginx/conf.d/zabbix.conf` Después de guardar el archivo editado, reinicie NGINX iniciando sesión mediante SSH como usuario root y ejecutando:

```
systemctl restart nginx
```

Dirección IP estática

De forma predeterminada, el appliance usa DHCP para obtener la dirección IP. Para establecer una dirección IP estática:

1. Inicie sesión como usuario root.
2. Ejecute los siguientes comandos, reemplazando los valores por su dirección IP personalizada:

```
nmcli con mod eth0 ipv4.addresses 192.168.1.10/24    # Dirección IP/CIDR del appliance
nmcli con mod eth0 ipv4.gateway 192.168.1.1     # Dirección IP de la puerta de enlace
nmcli con mod eth0 ipv4.dns 8.8.8.8    # Dirección IP del servidor DNS
nmcli con mod eth0 ipv4.method manual
nmcli con up eth0
```

Como alternativa, puede abrir el archivo `/etc/NetworkManager/system-connections/eth0.nmconnection` y agregar las siguientes líneas:

```
[ipv4]
address1=192.168.1.10/24
dns=8.8.8.8;
gateway=192.168.1.1
method=manual
```

Después de guardar el archivo editado, ejecute los comandos:

```
nmcli con reload
nmcli con up eth0
```

Para verificar que los cambios de configuración de red se hayan aplicado, ejecute el siguiente comando:

```
nmcli dev show eth0
```

Configuración del firewall

Para gestionar la configuración del firewall, el appliance utiliza iptables con reglas predefinidas:

- Abrir el puerto SSH (22 TCP)
- Abrir los puertos del agente Zabbix (10050 TCP) y Zabbix trapper (10051 TCP)
- Abrir los puertos HTTP (80 TCP) y HTTPS (443 TCP)
- Abrir el puerto SNMP trap (162 UDP)
- Abrir conexiones salientes al puerto NTP (123 UDP)
- Limitar los paquetes ICMP a 5 paquetes por segundo
- Rechazar todas las demás conexiones entrantes

Para abrir puertos adicionales, modifique el archivo `/etc/sysconfig/iptables` y recargue las reglas del firewall:

```
systemctl reload iptables
```

Repositorios

El appliance de Zabbix utiliza el paquete *zabbix-release* del repositorio de Zabbix. Los repositorios se configuran en el directorio `/etc/yum.repos.d/*`.

Zona horaria

Por defecto, el appliance utiliza UTC para el reloj del sistema. Para cambiar la zona horaria, copie el archivo apropiado de `/usr/share/zoneinfo` a `/etc/localtime`, por ejemplo:

```
cp /usr/share/zoneinfo/Europe/Riga /etc/localtime
```

> **Note:**
> La zona horaria de la interfaz web de Zabbix se establece por separado y puede cambiarse en la configuración de la interfaz. La zona horaria predeterminada para la interfaz web de Zabbix es Europe/Riga.

Ubicaciones de los archivos

- Los archivos de configuración del Zabbix server y del agent se encuentran en `/etc/zabbix`.
- Los archivos de registro del Zabbix server y del agent se encuentran en `/var/log/zabbix`.
- El archivo de configuración del frontend de Zabbix se encuentra en `/etc/zabbix/web`.
- El frontend de Zabbix se encuentra en `/usr/share/zabbix`.
- El directorio de inicio del usuario *zabbix* es `/var/lib/zabbix`.

Servicios del sistema

Los servicios de systemd están disponibles. Para ver la lista de servicios de Zabbix, ejecute el siguiente comando en la máquina virtual:

```
systemctl list-units zabbix*
```

Notas específicas del formato de imagen VMware

Las imágenes en formato *vmdk* se pueden utilizar directamente en los productos VMware Player, Server y Workstation.
Para utilizarlas en ESX, ESXi y vSphere, deben convertirse mediante VMware vCenter Converter (se requiere autenticación para la descarga).
Si utiliza VMware vCenter Converter, puede encontrarse con problemas con el adaptador de red híbrido.
En ese caso, puede intentar especificar el adaptador E1000 durante el proceso de conversión.
Como alternativa, una vez completada la conversión, puede eliminar el adaptador existente y añadir un adaptador E1000.

Imagen HDD/flash (raw)

Para arrancar la imagen, ejecute:

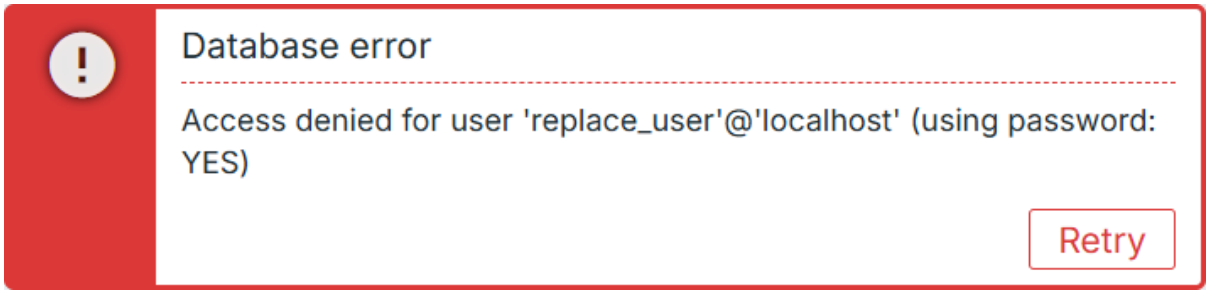
```
dd if=./zabbix_appliance_8.0.0.raw of=/dev/sdc bs=4k conv=fdatasync
```

Reemplace `/dev/sdc` con la ruta de su dispositivo de disco Flash/HDD.

Hyper-V

Si el appliance no se inicia en Hyper-V, intente presionar **Ctrl+Alt+F2** para cambiar a una sesión TTY.

Solución de problemas Si encuentra el mensaje de error *Access denied for user 'replace_user'@'localhost' (using password: YES)* al intentar iniciar sesión en el frontend, puede indicar que la instalación aún está en curso.



Si el error persiste después de esperar unos minutos, o si observa cualquier otro comportamiento inesperado, probablemente significa que el proceso de instalación no se ha completado correctamente. En este caso, recomendamos eliminar el appliance actual y volver a implementarlo siguiendo las mismas instrucciones de instalación. Este paso normalmente resuelve el problema.

Tenga en cuenta que no se recomienda intentar reparar manualmente una instalación dañada, ya que puede provocar complicaciones adicionales.

Procesos

Descripción general Zabbix consta de varios procesos que trabajan juntos para supervisar sistemas y enviar notificaciones. Cada proceso tiene una función específica en la infraestructura de supervisión.

Procesos principales

- Server: el proceso central que realiza el sondeo y la recepción de datos, calcula los trigger y envía notificaciones a los usuarios.
- Agent: recopila datos de los host supervisados.
- Agent 2: un agent de nueva generación con arquitectura de plugins y capacidades mejoradas.
- Proxy: recopila datos de supervisión de los dispositivos supervisados en nombre del server de Zabbix y envía la información a un server de Zabbix centralizado en una configuración distribuida.
- Java gateway: permite supervisar aplicaciones Java.
- MCP server: permite que los agentes de IA trabajen con Zabbix mediante el Model Context Protocol.

Para acceder fácilmente a Zabbix desde cualquier lugar y desde cualquier plataforma, se proporciona una interfaz basada en la web. La interfaz forma parte del server de Zabbix y, normalmente (aunque no necesariamente), se ejecuta en la misma máquina física que el server.

Toda la información de configuración, así como los datos recopilados por Zabbix, se almacena en una base de datos.

Encabezado El encabezado está presente en todos los mensajes de solicitud y respuesta entre los componentes de Zabbix. Es necesario para determinar la longitud del mensaje, si está comprimido o no, y si es un paquete grande o no.

El protocolo de comunicaciones de Zabbix tiene un límite de tamaño de paquete de 1 GB por conexión. El límite de 1 GB se aplica tanto a la longitud de los datos del paquete recibido como a la longitud de los datos sin comprimir.

Al enviar la configuración a Zabbix proxy, el límite de tamaño de paquete se incrementa a 4 GB para permitir la sincronización de configuraciones grandes. Cuando la longitud de los datos antes de la compresión supera los 4 GB, Zabbix server automáticamente empieza a usar el formato de paquete grande (marca 0x04), lo que aumenta el límite de tamaño de paquete a 16 GB.

Tenga en cuenta que, aunque el formato de paquete grande puede usarse para enviar cualquier dato, actualmente solo el sincronizador de configuración de Zabbix proxy puede manejar paquetes de más de 1 GB.

Estructura del encabezado El encabezado consta de cuatro campos. Todos los números del encabezado están formateados en little-endian.

Field	Size	Size (large packet)	Description
<PROTOCOL>	4	4	"ZBXD" o 5A 42 58 44
<FLAGS>	1	1	Banderas del protocolo: 0x01 - protocolo de comunicaciones de Zabbix 0x02 - compresión 0x04 - paquete grande
<DATALEN>	4	8	Longitud de los datos.
<RESERVED>	4	8	Cuando se usa compresión (bandera 0x02) - la longitud de los datos sin comprimir Cuando no se usa compresión - 00 00 00 00

Ejemplos de encabezado Aquí hay algunos fragmentos de código que muestran cómo agregar el encabezado del protocolo Zabbix a los datos que desea enviar para obtener el paquete que debe enviar a Zabbix para que se interprete correctamente. Estos fragmentos de código asumen que los datos no son mayores que 1 GB, por lo que no se utiliza el formato de paquete grande.

Python

```
packet = b"ZBXD\1" + struct.pack("<II", len(data), 0) + data
```

o

```
def zbx_create_header(plain_data_size, compressed_data_size=None):
    protocol = b"ZBXD"
    flags = 0x01
    if compressed_data_size is None:
        datalen = plain_data_size
        reserved = 0
    else:
        flags |= 0x02
        datalen = compressed_data_size
        reserved = plain_data_size
    return protocol + struct.pack("<BII", flags, datalen, reserved)

packet = zbx_create_header(len(data)) + data
```

Perl

```
my $packet = "ZBXD\1" . pack("(II)<", length($data), 0) . $data;
```

o

```
sub zbx_create_header($;$)
{
    my $plain_data_size = shift;
    my $compressed_data_size = shift;

    my $protocol = "ZBXD";
    my $flags = 0x01;
    my $datalen;
    my $reserved;

    if (!defined($compressed_data_size))
    {
        $datalen = $plain_data_size;
        $reserved = 0;
    }
    else
    {
        $flags |= 0x02;
        $datalen = $compressed_data_size;
```

```
        $reserved = $plain_data_size;
    }

    return $protocol . chr($flags) . pack("(II)<", $datalen, $reserved);
}

my $packet = zbx_create_header(length($data)) . $data;
```

PHP

```
$packet = "ZBXD\1" . pack("VV", strlen($data), 0) . $data;
```

o

```
function zbx_create_header($plain_data_size, $compressed_data_size = null)
{
    $protocol = "ZBXD";
    $flags = 0x01;
    if (is_null($compressed_data_size))
    {
        $datalen = $plain_data_size;
        $reserved = 0;
    }
    else
    {
        $flags |= 0x02;
        $datalen = $compressed_data_size;
        $reserved = $plain_data_size;
    }
    return $protocol . chr($flags) . pack("VV", $datalen, $reserved);
}

$packet = zbx_create_header(strlen($data)) . $data;
```

Bash

```
datalen=$(printf "%08x" ${#data})
datalen="\\x${datalen:6:2}\\x${datalen:4:2}\\x${datalen:2:2}\\x${datalen:0:2}"
printf "ZBXD\1${datalen}\0\0\0\0%s" "$data"
```

server

Overview Zabbix server es el proceso central del software Zabbix.

El server realiza el sondeo y la captura de datos, calcula triggers y envía notificaciones a los usuarios. Es el componente central al que los agents de Zabbix y los proxies informan sobre la disponibilidad e integridad de los sistemas. El server puede comprobar de forma remota servicios en red (como servidores web y servidores de correo) mediante comprobaciones simples de servicios.

El server es el repositorio central en el que se almacenan todos los datos de configuración, estadísticos y operativos, y es la entidad de Zabbix que alertará activamente a los administradores cuando surjan problemas en cualquiera de los sistemas monitorizados.

El funcionamiento de un Zabbix server básico se divide en tres componentes distintos; estos son: Zabbix server, web frontend y almacenamiento de base de datos.

Toda la información de configuración de Zabbix se almacena en la base de datos, con la que interactúan tanto el server como el web frontend. Por ejemplo, cuando crea un nuevo item usando el web frontend (o la API), se añade a la tabla items en la base de datos. Después, aproximadamente una vez por minuto, Zabbix server consultará la tabla items para obtener una lista de los items que están activos, que luego se almacena en una caché dentro de Zabbix server. Por eso, los cambios realizados en Zabbix frontend pueden tardar hasta dos minutos en aparecer en la sección de datos más recientes.

Véase también:

- High availability setup for Zabbix servers
- Configuration parameters: Zabbix server

Ejecución del server Si está instalado como paquete

Zabbix server se ejecuta como un proceso daemon. El server se puede iniciar ejecutando:

```
systemctl start zabbix-server
```

Esto funcionará en la mayoría de los sistemas GNU/Linux. En otros sistemas, es posible que deba ejecutar:

```
/etc/init.d/zabbix-server start
```

De forma similar, para detener, reiniciar o ver el estado, use los siguientes comandos:

```
systemctl stop zabbix-server
systemctl restart zabbix-server
systemctl status zabbix-server
```

Iniciar manualmente

Si lo anterior no funciona, debe iniciarlo manualmente. Busque la ruta al binario `zabbix_server` y ejecute:

```
zabbix_server
```

Puede usar los siguientes parámetros de línea de comandos con Zabbix server:

Parameter	Description
`-c --config <config-file>`	Ruta al archivo de configuración.
`-f --foreground`	Ejecutar Zabbix server en primer plano.
`-R --runtime-control <option>`	Realizar funciones administrativas.
`-T --test-config`	Validar el archivo de configuración y salir.
`-h --help`	Mostrar información de ayuda y salir.
`-V --version`	Mostrar el número de versión.

Ejemplos de ejecución de Zabbix server con parámetros de línea de comandos:

```
zabbix_server -c /usr/local/etc/zabbix_server.conf
zabbix_server --help
zabbix_server -V
```

Control del tiempo de ejecución

Opciones de control del tiempo de ejecución:

Opción	Descripción	Destino
`config_cache_reload`	Recargar la caché de configuración. Se ignora si la caché se está cargando actualmente.	
`history_cache_clear=<target>`	Borrar la caché del historial para el item especificado por su ID. Afecta a todos los valores del item, excepto al primero y al último.	**target** - ID del item
`diaginfo[=<section>]`	Recopilar información de diagnóstico en el archivo de registro del server.	`historycache` - estadísticas de la caché del historial; `valuecache` - estadísticas de la caché de valores; `preprocessing` - estadísticas del gestor de preprocesamiento; `alerting` - estadísticas del gestor de alertas; `lld` - estadísticas del gestor de LLD; `locks` - lista de mutexes (está vacía en sistemas *BSD*); `connector` - estadísticas de los conectores con la cola más grande.
`ha_status`	Registrar el estado del clúster de alta disponibilidad (HA).	
`ha_remove_node=<target>`	Eliminar el nodo de alta disponibilidad (HA) especificado por su nombre o ID. Tenga en cuenta que los nodos activos/en espera no se pueden eliminar.	**target** - nombre o ID del nodo (se puede obtener ejecutando `ha_status`).

Opción	Descripción	Destino
`ha_set_failover_delay`	Establecer el retraso de conmutación por error de alta disponibilidad (HA). Se admiten sufijos de tiempo, por ejemplo, 10s, 1m.	
`proxy_config_cache_reload [=<target>]`	Recargar la caché de configuración del proxy.	**target** - lista de nombres de proxy separados por comas. Si no se especifica ningún destino, se recarga la configuración de todos los proxies.
`secrets_reload`	Recargar los secretos desde Vault.	
`service_cache_reload`	Recargar la caché del gestor de servicios.	
`snmp_cache_reload`	Recargar la caché de SNMP: borrar las propiedades del motor SNMP (tiempo del motor, arranques del motor, ID del motor y credenciales) para todos los hosts. Utilícelo para forzar un borrado global de la caché al solucionar problemas de SNMP.	
`housekeeper_execute`	Iniciar el procedimiento de mantenimiento. Se ignora si el procedimiento de mantenimiento está actualmente en curso.	
`trigger_housekeeper_execute`	Iniciar el procedimiento de mantenimiento de los triggers. Se ignora si el procedimiento de mantenimiento de los triggers está actualmente en curso.	
`log_level_increase [=<target>]`	Aumentar el nivel de registro; afecta a todos los procesos si no se especifica el destino. No se admite en sistemas *BSD*.	**process type** - todos los procesos del tipo especificado (por ejemplo, `poller`). Consulte todos los tipos de procesos del server. **process type,N** - tipo y número del proceso (por ejemplo, `poller,3`). **pid** - identificador del proceso (1 a 65535). Para valores mayores, especifique el destino como 'process type,N'.
`log_level_decrease [=<target>]`	Disminuir el nivel de registro; afecta a todos los procesos si no se especifica el destino. No se admite en sistemas *BSD*.	
`prof_enable [=<target>]`	Habilitar la creación de perfiles. Afecta a todos los procesos si no se especifica el destino. La creación de perfiles habilitada proporciona detalles de todos los rwlocks/mutexes por nombre de función.	**process type** - todos los procesos del tipo especificado (por ejemplo, `history syncer`) Tipos de procesos admitidos como destinos de creación de perfiles: `alerter`, `alert manager`, `availability manager`, `configuration syncer`, `discovery manager`, `escalator`, `history poller`, `history syncer`, `housekeeper`, `http poller`, `icmp pinger`, `ipmi manager`, `ipmi poller`, `java poller`, `lld manager`, `lld worker`, `odbc poller`, `poller`, `preprocessing manager`, `proxy poller`, `self-monitoring`, `service manager`, `snmp trapper`, `task manager`, `timer`, `trapper`, `unreachable poller`, `vmware collector`. **process type,N** - tipo y número del proceso (por ejemplo, `history syncer,1`). **pid** - identificador del proceso (1 a 65535). Para valores mayores, especifique el destino como 'process type,N'. **scope** - `rwlock`, `mutex`, `processing` se pueden utilizar con el tipo y número del proceso (por ejemplo, `history syncer,1,processing`) o con todos los procesos del tipo (por ejemplo, `history syncer,rwlock`).

Opción	Descripción	Destino
`prof_disable[=<destino>]`	Deshabilitar la creación de perfiles. Afecta a todos los procesos si no se especifica el destino.	**process type** - todos los procesos del tipo especificado (por ejemplo, `history syncer`). Tipos de procesos admitidos como destinos de creación de perfiles: consulte `prof_enable`. **process type,N** - tipo y número del proceso (por ejemplo, `history syncer,1`). **pid** - identificador del proceso (1 a 65535). Para valores mayores, especifique el destino como 'process type,N'.

Cuando se especifica un tipo de proceso como destino para el control en tiempo de ejecución, el comando se aplica solo a los procesos en ejecución de ese tipo. Si actualmente no se está ejecutando ningún proceso del tipo especificado, el comando devuelve un error:

```
zabbix_server -R log_level_increase='discovery manager'
Cannot redirect signal: "discovery manager" process does not exist
```

Ejemplo de uso del control en tiempo de ejecución para recargar la caché de configuración del server:

```
zabbix_server -c /usr/local/etc/zabbix_server.conf -R config_cache_reload
```

Ejemplos de uso del control en tiempo de ejecución para recargar la configuración del proxy:

```
### Recargar la configuración de todos los proxies:
zabbix_server -R proxy_config_cache_reload

### Recargar la configuración de Proxy1 y Proxy2:
zabbix_server -R proxy_config_cache_reload=Proxy1,Proxy2
```

Ejemplo de uso del control en tiempo de ejecución para vaciar la caché de historial de un item:

```
zabbix_server -c /usr/local/etc/zabbix_server.conf -R history_cache_clear=42243
```

Ejemplos de uso del control en tiempo de ejecución para recopilar información de diagnóstico:

```
### Recopilar toda la información de diagnóstico disponible en el archivo de log del server:
zabbix_server -R diaginfo

### Recopilar estadísticas de la caché de historial en el archivo de log del server:
zabbix_server -R diaginfo=historycache
```

Ejemplo de uso del control en tiempo de ejecución para recargar la caché SNMP:

```
zabbix_server -R snmp_cache_reload
```

Attention:
Cuando una interfaz SNMPv3 se actualiza mediante la interfaz web de Zabbix, Zabbix recargará automáticamente las nuevas credenciales SNMPv3 para esa interfaz en la mayoría de los casos; use `-R snmp_cache_reload` solo si la sondeo sigue fallando después de cambiar las credenciales (por ejemplo, debido a inconsistencias de engineBoots/engineID o a dispositivos no compatibles con RFC), o cuando necesite forzar un vaciado global de la caché SNMP para tareas de resolución de problemas.

Ejemplo de uso del control en tiempo de ejecución para ejecutar housekeeper:

```
zabbix_server -c /usr/local/etc/zabbix_server.conf -R housekeeper_execute
```

Ejemplos de uso del control en tiempo de ejecución para cambiar el nivel de log:

```
### Aumentar el nivel de log de todos los procesos:
zabbix_server -c /usr/local/etc/zabbix_server.conf -R log_level_increase

### Aumentar el nivel de log del segundo proceso poller:
zabbix_server -c /usr/local/etc/zabbix_server.conf -R log_level_increase=poller,2

### Aumentar el nivel de log del proceso con PID 1234:
zabbix_server -c /usr/local/etc/zabbix_server.conf -R log_level_increase=1234
```

```
### Disminuir el nivel de log de todos los procesos http poller:
zabbix_server -c /usr/local/etc/zabbix_server.conf -R log_level_decrease="http poller"
```

Ejemplo de establecer el retraso de failover de HA al mínimo de 10 segundos:

```
zabbix_server -R ha_set_failover_delay=10s
```

Usuario del proceso

Zabbix server está diseñado para ejecutarse como un usuario que no sea root. Se ejecutará con el usuario no root con el que se inicie. Por lo tanto, puede ejecutar server con cualquier usuario no root sin ningún problema.

Si intenta ejecutarlo como 'root', cambiará a un usuario 'zabbix' codificado, que debe estar presente en su sistema. Solo puede ejecutar server como 'root' si modifica el parámetro 'AllowRoot' en el archivo de configuración de server en consecuencia.

Si Zabbix server y agent se ejecutan en la misma máquina, se recomienda usar un usuario diferente para ejecutar server que para ejecutar agent. De lo contrario, si ambos se ejecutan con el mismo usuario, agent puede acceder al archivo de configuración de server y cualquier usuario con nivel Admin en Zabbix puede obtener fácilmente, por ejemplo, la contraseña de la base de datos.

Archivo de configuración

La ruta predeterminada al archivo de configuración del server depende del sistema operativo y del método de instalación de Zabbix:

- Para la instalación desde las fuentes: `/usr/local/etc/zabbix_server.conf`, o según lo establecido por las variables de compilación `--sysconfdir` o `--prefix`.
- Para la instalación desde paquetes y el appliance: `/etc/zabbix/zabbix_server.conf`.

Consulte las opciones del archivo de configuración para obtener más información sobre la configuración de `zabbix_server`.

Scripts de inicio

Los scripts se utilizan para iniciar/detener automáticamente los procesos de Zabbix durante el arranque/apagado del sistema. Los scripts se encuentran en el directorio misc/init.d.

Tipos de procesos y threads del server

- `agent poller`: proceso poller asíncrono para comprobaciones pasivas con un thread de trabajo;
- `alert manager`: gestor de la cola de alertas;
- `alert syncer`: escritor de alertas en la base de datos;
- `alerter`: proceso para enviar notificaciones;
- `availability manager`: proceso para actualizar la disponibilidad de los host;
- `browser poller`: poller para comprobaciones de item del navegador;
- `configuration syncer`: proceso para gestionar la caché en memoria de los datos de configuración;
- `configuration syncer worker`: proceso para resolver y sincronizar los valores de las macros de usuario en los nombres de los item;
- `connector manager`: proceso gestor de los conectores;
- `connector worker`: proceso para gestionar las solicitudes del gestor de conectores;
- `discovery manager`: proceso gestor para el descubrimiento de dispositivos;
- `discovery worker`: proceso para gestionar las tareas de descubrimiento del gestor de descubrimiento;
- `escalator`: proceso para la escalación de acciones;
- `ha manager`: proceso para gestionar la alta disponibilidad;
- `history poller`: proceso para gestionar las comprobaciones calculadas que requieren una conexión a la base de datos;
- `history syncer`: escritor del historial en la base de datos;
- `housekeeper`: proceso para eliminar datos obsoletos (historial y tendencias de los item, sesiones de usuario, eventos, etc.), así como los datos restantes de objetos eliminados;
- `http agent poller`: proceso poller asíncrono para comprobaciones HTTP con un thread de trabajo;
- `http poller`: poller de monitorización web;
- `icmp pinger`: poller para comprobaciones icmpping;
- `internal poller`: poller para comprobaciones internas;
- `ipmi manager`: gestor del poller IPMI;
- `ipmi poller`: poller para comprobaciones IPMI;
- `java poller`: poller para comprobaciones de Java;
- `lld manager`: proceso gestor de tareas de descubrimiento de bajo nivel;
- `lld worker`: proceso de trabajo de tareas de descubrimiento de bajo nivel;
- `odbc poller`: poller para comprobaciones ODBC;
- `poller`: poller normal para comprobaciones pasivas;
- `preprocessing manager`: gestor de tareas de preprocesamiento con threads de trabajo de preprocesamiento;

- `preprocessing worker`: thread para el preprocesamiento de datos;
- `proxy poller`: poller para proxy pasivos;
- `proxy group manager`: gestor del balanceo de carga y la alta disponibilidad de los proxy;
- `report manager`: gestor de tareas programadas de generación de informes;
- `report writer`: proceso para generar informes programados;
- `self-monitoring`: proceso para recopilar estadísticas internas del server;
- `service manager`: proceso para gestionar servicios mediante la recepción de información sobre problemas, etiquetas de problemas y recuperación de problemas desde history syncer, task manager y alert manager;
- `snmp poller`: proceso poller asíncrono para comprobaciones SNMP con un thread de trabajo (solo item `walk[OID]` y `get[OID]`);
- `snmp trapper`: trapper para traps SNMP;
- `supervisor`: coordina el inicio y el apagado de los procesos y threads del server, y supervisa su disponibilidad;
- `task manager`: proceso para la ejecución remota de tareas solicitadas por otros componentes (por ejemplo, cerrar un problema, reconocer un problema, comprobar ahora el valor de un item y la funcionalidad de comandos remotos);
- `telemetry query poller`: poller para comprobaciones de item de consulta de telemetría;
- `timer`: temporizador para procesar mantenimientos;
- `trapper`: trapper para comprobaciones activas, traps y comunicación con el proxy;
- `trigger housekeeper`: proceso para eliminar problemas y eventos generados por trigger que posteriormente se hayan eliminado;
- `unreachable poller`: poller para dispositivos inaccesibles;
- `vmware collector`: recopilador de datos de VMware responsable de recopilar datos de los servicios de VMware.

El archivo de registro del server se puede utilizar para observar estos tipos de procesos.

El archivo de registro del server se crea con permisos de lectura y escritura únicamente para el propietario del archivo. Además, el grupo propietario puede leer el archivo. Todos los demás permisos están denegados.

Se pueden supervisar varios tipos de procesos del server de Zabbix mediante el item interno `zabbix[process,<type>,<mode>,<state>]`.

Estadísticas de transacciones del history syncer

El título del proceso history syncer muestra estadísticas detalladas sobre las transacciones de history syncer:

```
205182 ?        S      0:00  zabbix_server: history syncer #2 [processed 0 values, 0+0 triggers in 0.00002
205183 ?        S      0:00  zabbix_server: history syncer #3 [processed 18 values, 7+0 triggers in 0.0026
205184 ?        S      0:00  zabbix_server: history syncer #4 [processed 0 values, 0+0 triggers in 0.00002
```

En "A+B triggers":

- A - triggers procesados debido a valores de history;
- B - triggers procesados debido a temporizadores.

Los tiempos, en `processed...in N (<timings>) sec`, son:

- Tiempo dedicado a escribir los valores de item en la base de datos;
- Tiempo dedicado a actualizar los datos del item (estado, errores, inventario del host, etc.);
- Tiempo dedicado a vaciar los trends en la base de datos;
- Tiempo dedicado a calcular triggers;
- Tiempo dedicado a procesar eventos y acciones.

Procedimiento de housekeeping

El proceso `housekeeper` elimina periódicamente los datos obsoletos (historial y tendencias de item, sesiones de usuario, eventos, etc.), así como los datos que quedaron de objetos eliminados. Se ejecuta en ciclos, con una frecuencia y un límite de eliminación por ciclo determinados por `HousekeepingFrequency` y `MaxHousekeeperDelete`. Los datos que no se eliminan en un ciclo se trasladan al siguiente. El housekeeping automático puede habilitarse y configurarse en *Administration* > *Housekeeping*.

Cuando el proceso `housekeeper` elimina datos que quedaron de objetos eliminados, se basa en tareas de la tabla `housekeeper`, que se rellena cada vez que se elimina un objeto. Por ejemplo, cuando elimina un host, Zabbix elimina sus items, pero no su historial, tendencias ni problemas. En su lugar, los triggers de la base de datos rellenan la tabla `housekeeper` con tareas compuestas por estos campos:

- `housekeeperid` - ID de la tarea
- `object` - tipo de objeto (0 - item; 1 - trigger; 2 - servicio; 3 - host descubierto; 4 - servicio descubierto)
- `objectid` - ID del objeto (ayuda a housekeeper a encontrar los datos relacionados con el objeto)

Por ejemplo, al eliminar un host con dos items y un trigger, la tabla `housekeeper` se rellena de la siguiente manera:

```
+---------------+--------+----------+
| housekeeperid | object | objectid |
+---------------+--------+----------+
```

```
|              1 |       1 |    28724 |
|              2 |       0 |    59396 |
|              3 |       0 |    59397 |
+----------------+---------+----------+
```

Los triggers de la base de datos rellenan la tabla `housekeeper` con tareas sin comprobar si existen datos relacionados con el objeto; esa comprobación la realiza el proceso `housekeeper` cuando elimina los datos:

- Para items eliminados (incluidas las reglas LLD) - elimina sus datos de todas las tablas de historial y tendencias (`history`, `history_str`, `history_log`, `history_uint`, `history_text`, `history_bin`, `history_json`, `trends`, `trends_uint`). También elimina sus eventos internos de la tabla `problem`.
- Para triggers eliminados - elimina sus eventos de las tablas relacionadas con eventos (`problem`, `event_symptom`, `events`; la propia base de datos también elimina los eventos de la tabla `event_recovery`). También notifica al proceso `service manager` sobre los eventos eliminados.

Note:
Un proceso independiente `trigger housekeeper` se encarga de una tarea más limitada: eliminar problemas y eventos que no tienen un trigger de origen conocido. Su frecuencia de ejecución está controlada por `ProblemHousekeepingFrequency`.
Hasta que se inicie el procedimiento de housekeeping de triggers, los problemas creados por triggers que desde entonces se hayan eliminado podrían seguir generando problemas de servicio y asignándolos a servicios. Si su entorno incluye muchas reglas de cálculo de estado de servicio basadas en triggers descubiertos/no descubiertos con frecuencia, considere aumentar la frecuencia del procedimiento de housekeeping ajustando el parámetro de configuración del server `ProblemHousekeepingFrequency`.

- Para servicios eliminados - elimina sus problemas de la tabla `problem`; sus eventos de la tabla `events` se eliminan solo después del período de almacenamiento de datos configurado.
- Para hosts/servicios descubiertos eliminados (network discovery) - elimina sus eventos de la tabla `events`.

El proceso `housekeeper` elimina solo aquellos eventos que no están asociados con problemas. Por ejemplo, un evento obsoleto de problema/recuperación no se eliminará si está asociado a un problema abierto. Cuando `housekeeper` elimina entidades obsoletas, primero elimina los problemas y luego los eventos.

Note:
Las tablas que usan el modo `partition` (tablas particionadas de TimescaleDB) se omiten; solo se procesan las tablas que usan el modo `regular`.

Plataformas compatibles Debido a los requisitos de seguridad y a la naturaleza crítica de la operación del server, UNIX es el único sistema operativo que puede ofrecer de forma consistente el rendimiento, la tolerancia a fallos y la resiliencia necesarios. Zabbix funciona en las versiones líderes del mercado.

El server de Zabbix se prueba en las siguientes plataformas:

- Linux
- Solaris
- AIX
- HP-UX
- Mac OS X
- FreeBSD
- OpenBSD
- NetBSD
- SCO Open Server

Note:
Es posible que Zabbix también funcione en otros sistemas operativos similares a Unix.

Configuración regional Tenga en cuenta que el server requiere una configuración regional UTF-8 para que algunos elementos textuales puedan interpretarse correctamente. La mayoría de los sistemas modernos tipo Unix tienen una configuración regional UTF-8 como predeterminada; sin embargo, hay algunos sistemas en los que puede ser necesario establecerla específicamente.

Configuración de alta disponibilidad para servers Zabbix

Descripción general

La alta disponibilidad (HA) suele ser necesaria en infraestructuras críticas que no pueden permitirse prácticamente ningún tiempo de inactividad. Por ello, para cualquier servicio que pueda fallar debe existir una opción de failover que asuma el control si el servicio actual falla.

Zabbix ofrece una solución de alta disponibilidad **nativa** que es fácil de configurar y no requiere experiencia previa en HA. La HA nativa de Zabbix puede ser útil como una capa adicional de protección frente a fallos de software o hardware del server de Zabbix, o para reducir el tiempo de inactividad debido al mantenimiento.

En el modo de alta disponibilidad de Zabbix, varios servers de Zabbix se ejecutan como nodos en un clúster. Mientras un server de Zabbix del clúster está activo, los demás permanecen en espera, listos para asumir el control si es necesario.



Cambiar a Zabbix HA no es vinculante. Puede volver al funcionamiento independiente en cualquier momento.

Véase también: Detalles de implementación

Habilitar alta disponibilidad

Iniciar Zabbix server como nodo del clúster

Se requieren dos parámetros en la configuración del server para iniciar un Zabbix server como nodo del clúster:

- El parámetro `HANodeName` debe especificarse para cada Zabbix server que será un nodo del clúster de alta disponibilidad (HA).

Este es un identificador único del nodo (por ejemplo, `zabbix-node-01`) con el que se hará referencia al server en las configuraciones del agent y del proxy.
Si no especifica `HANodeName`, el server se iniciará en modo independiente.

- El parámetro `NodeAddress` debe especificarse para cada nodo.

El parámetro `NodeAddress` (`address:port`) será utilizado por el frontend de Zabbix para conectarse al nodo activo del server.
`NodeAddress` debe coincidir con la dirección IP o el nombre FQDN del Zabbix server correspondiente.

Reinicie todos los Zabbix servers después de realizar cambios en los archivos de configuración.
Ahora se iniciarán como nodos del clúster.
El nuevo estado de los servers puede consultarse en *Informes* > *Información del sistema* y también ejecutando:

```
zabbix_server -R ha_status
```

Este comando de tiempo de ejecución registrará el estado actual del clúster de HA en el registro del Zabbix server (y en la salida estándar):

```
Failover delay: 60 seconds
Cluster status:
   #  ID                       Name                  Address                 Status      Last Access
   1. ckzxxqg7u0001lsropenyzh3m zabbix-node-01        64.227.66.193:10051     standby     0s
   2. ckzxyqo1k00013frpq539e1jp zabbix-node-02        64.227.74.25:10051      active      3s
```

Preparación del frontend

Asegúrese de que `address:port` del Zabbix server no esté definido en el archivo de configuración del frontend de Zabbix (`zabbix.conf.php`).

```
// Uncomment and set to desired values to override Zabbix hostname/IP and port.
// $ZBX_SERVER                 = '';
// $ZBX_SERVER_PORT            = '';
```

El frontend de Zabbix detectará automáticamente el nodo activo leyendo la configuración de la tabla de nodos de la base de datos de Zabbix. La dirección del nodo activo se utilizará como dirección del Zabbix server.

Configuración de proxy

Los nodos del clúster HA (servers) deben figurar en la configuración de un proxy Zabbix pasivo o activo.

Para un proxy pasivo, los nombres de los nodos deben figurar en el parámetro Server del proxy, separados por una **coma**.

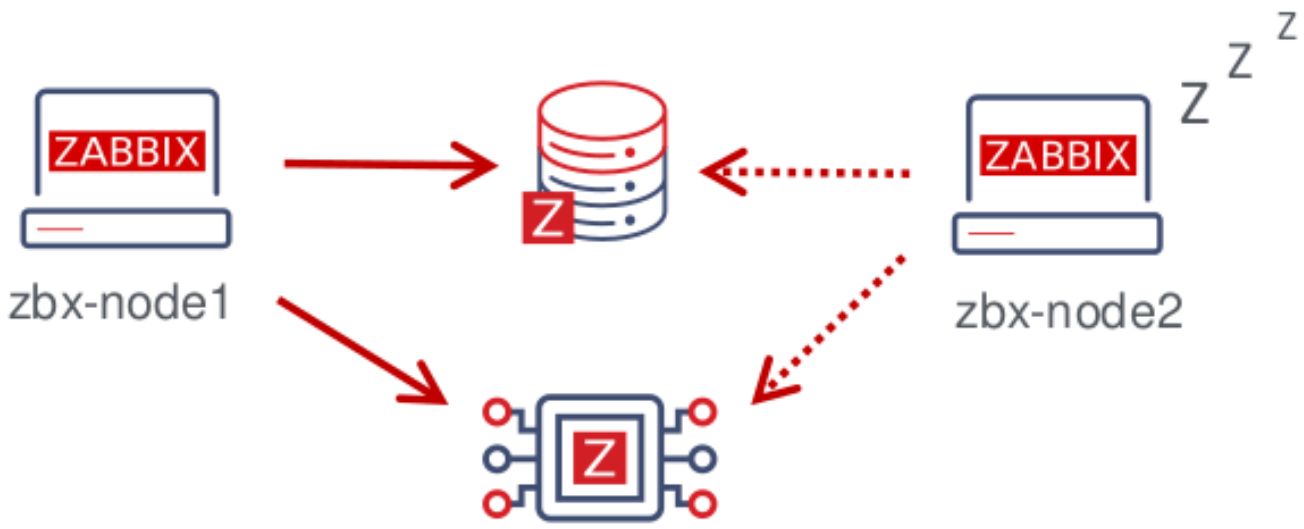
```
Server=zabbix-node-01,zabbix-node-02
```

Para un proxy activo, los nombres de los nodos deben figurar en el parámetro Server del proxy, separados por un **punto y coma**.

```
Server=zabbix-node-01;zabbix-node-02
```

Configuración del agent

Los nodos del clúster HA (servers) deben incluirse en la configuración de Zabbix agent o Zabbix agent 2.



Para habilitar las comprobaciones pasivas, los nombres de los nodos deben incluirse en el parámetro Server parameter, separados por una **coma**.

```
Server=zabbix-node-01,zabbix-node-02
```

Para habilitar las comprobaciones activas, los nombres de los nodos deben incluirse en el parámetro ServerActive parameter. Tenga en cuenta que, para las comprobaciones activas, los nodos deben separarse de cualquier otro servidor mediante una coma, mientras que los propios nodos deben separarse con un **punto y coma**, por ejemplo:

```
ServerActive=zabbix-node-01;zabbix-node-02
```

Conmutación por error al nodo en espera

Zabbix realizará la conmutación por error a otro nodo automáticamente si el nodo activo se detiene. Debe haber al menos un nodo en estado de espera para que se produzca la conmutación por error.

¿Qué tan rápida será la conmutación por error? Todos los nodos actualizan su última hora de acceso (y el estado, si cambia) cada 5 segundos. Por lo tanto:

- Si el nodo activo se apaga y logra informar su estado como ”stopped”, otro nodo asumirá el control en un plazo de **5 segundos**.
- Si el nodo activo se apaga o deja de estar disponible sin poder actualizar su estado, los nodos en espera esperarán el **failover delay** + 5 segundos para asumir el control.

El failover delay es configurable, con un rango admitido entre 10 segundos y 15 minutos (un minuto por defecto). Para cambiar el failover delay, puede ejecutar:

```
zabbix_server -R ha_set_failover_delay=5m
```

Gestión del clúster de HA

El estado actual del clúster de HA se puede gestionar mediante las opciones dedicadas de control en tiempo de ejecución:

- `ha_status`: registra el estado del clúster de HA en el registro del Zabbix server (y en la salida estándar);
- `ha_remove_node=target`: elimina un nodo de HA identificado por su `<target>`: el nombre o el ID del nodo (el nombre/ID se puede obtener de la salida de `ha_status`), por ejemplo:

```
zabbix_server -R ha_remove_node=zabbix-node-02
```

Tenga en cuenta que los nodos activos/en espera no se pueden eliminar.

- `ha_set_failover_delay=delay`: establece el retraso de conmutación por error de HA (entre 10 segundos y 15 minutos; se admiten sufijos de tiempo (por ejemplo, `30s`, `1m`)).

El estado del nodo se puede supervisar:

- En *Informes* > *Información del sistema*.
- En el widget del panel *Información del sistema*.
- Mediante la opción de control en tiempo de ejecución `ha_status` del server (consulte lo anterior).

El item interno `zabbix[cluster,discovery,nodes]` se puede utilizar para descubrir nodos, ya que devuelve un JSON con la información de los nodos de alta disponibilidad.

Deshabilitar un clúster HA

Para deshabilitar un clúster de alta disponibilidad:

1. Haga copias de seguridad de los archivos de configuración.
2. Detenga los nodos en espera.
3. Elimine el parámetro HANodeName del server primario activo.
4. Reinicie el server primario (se iniciará en modo independiente).

Actualización del clúster HA

Para realizar una actualización de versión mayor de los nodos HA:

1. Detenga todos los nodos.
2. Cree una copia de seguridad completa de la base de datos.
3. Si la base de datos usa replicación, asegúrese de que todos los nodos estén sincronizados y no tengan problemas. No actualice si la replicación está rota.
4. Seleccione un único nodo que realizará la actualización de la base de datos, cambie su configuración a modo independiente comentando `HANodeName` y actualícelo.
5. Asegúrese de que la actualización de la base de datos se haya completado por completo (*System information* debe mostrar que Zabbix server se está ejecutando).
6. Reinicie el nodo en modo HA.
7. Actualice e inicie el resto de los nodos (no es necesario cambiarlos a modo independiente, ya que en este punto la base de datos ya está actualizada).

En una actualización de versión menor, basta con actualizar el primer nodo, asegurarse de que se haya actualizado y esté en ejecución, y luego iniciar la actualización del siguiente nodo.

Detalles de implementación

El clúster de alta disponibilidad (HA) es una solución opcional y es compatible con Zabbix server. La solución nativa de HA está diseñada para ser sencilla de usar, funciona entre sitios y no tiene requisitos específicos para las bases de datos que Zabbix reconoce. Los usuarios pueden usar la solución nativa de HA de Zabbix o una solución de HA de terceros, según lo que mejor se adapte a los requisitos de alta disponibilidad de su entorno.

La solución consta de varias instancias o nodos de `zabbix_server`. Cada nodo:

- Se configura por separado.
- Usa la misma base de datos.
- Puede tener varios modos: active, standby, unavailable, stopped.

Solo un nodo puede estar activo (working) a la vez. Un nodo standby ejecuta solo un proceso: el gestor de HA. Un nodo standby no realiza recopilación de datos, procesamiento ni otras actividades habituales del server; no escucha en puertos; tiene un mínimo de conexiones a la base de datos.

Tanto los nodos active como los standby actualizan su hora de último acceso cada 5 segundos. Cada nodo standby supervisa la hora de último acceso del nodo active. Si la hora de último acceso del nodo active supera los segundos de 'failover delay', el nodo standby pasa a ser el nodo active y asigna el estado 'unavailable' al nodo que estaba activo anteriormente.

El nodo active supervisa su propia conectividad con la base de datos: si se pierde durante más de `failover delay`−5 segundos, debe detener todo el procesamiento y cambiar al modo standby. El nodo active también supervisa el estado de los nodos standby: si la hora de último acceso de un nodo standby supera los segundos de 'failover delay', al nodo standby se le asigna el estado `unavailable`.

Los nodos están diseñados para ser compatibles entre versiones menores de Zabbix.

Parámetros de configuración: Zabbix server

Descripción general

Los parámetros de configuración permiten personalizar el comportamiento del server de Zabbix.

Especifique los valores de los parámetros en el archivo de configuración del server de Zabbix (`zabbix_server.conf`). Todos los parámetros son **opcionales**, a menos que se indique explícitamente que el parámetro es obligatorio.

Esta página está estructurada de la siguiente manera:

- Descripción general de los parámetros (para consultar todos los detalles de un parámetro, haga clic en su nombre)
- Detalles de los parámetros

Parámetro	Descripción
AlertScriptsPath	Especifique la ubicación de los scripts de alerta.
AllowRoot	Permita que el server se ejecute como `root`.
AllowSoftwareUpdateCheck	Permita recibir información sobre actualizaciones de software desde `zabbix.com`.
AllowUnsupportedDBVersions	Permita que el server funcione con versiones de base de datos no compatibles.
BridgeAdapterURL	Establezca la URL HTTP(S) del adaptador de puente de Zabbix.
CacheSize	Establezca el tamaño de la caché de configuración (en bytes).
CacheUpdateFrequency	Establezca la frecuencia de actualización de la caché de configuración (en segundos).
DBHost	Especifique el host de la base de datos (o el directorio del socket).
DBName	Especifique el nombre de la base de datos. **Obligatorio**.
DBPassword	Especifique la contraseña de la base de datos.
DBPort	Especifique el puerto del host de la base de datos.
DBSchema	Especifique el nombre del esquema de la base de datos. Se utiliza para PostgreSQL.
DBSocket	Especifique la ruta al archivo de socket de MySQL.
DBUser	Especifique el usuario de la base de datos.
DBTLSConnect	Especifique un valor para exigir una conexión TLS con la base de datos.
DBTLSCAFile	Especifique la ruta completa del archivo que contiene los certificados de las CA de nivel superior para la verificación del certificado de la base de datos.
DBTLSCertFile	Especifique la ruta completa del archivo que contiene el certificado del server de Zabbix para autenticarse en la base de datos.
DBTLSKeyFile	Especifique la ruta completa del archivo que contiene la clave privada para autenticarse en la base de datos.
DBTLSCipher	Especifique la lista de cifrados de encriptación que el server de Zabbix permite para protocolos TLS hasta TLS v1.2. Compatible únicamente con MySQL.
DBTLSCipher13	Especifique la lista de suites de cifrado que el server de Zabbix permite para el protocolo TLS v1.3. Compatible únicamente con MySQL a partir de la versión 8.0.16.
DebugLevel	Establezca el nivel de depuración.
EnableGlobalScripts	Habilite los scripts globales en el server de Zabbix.
EnableMobileDevices	Habilite la compatibilidad con dispositivos móviles en el server de Zabbix.
ExportDir	Especifique un directorio para habilitar la exportación en tiempo real de eventos, historial y tendencias.
ExportFileSize	Establezca el tamaño máximo del archivo de exportación (en bytes).
ExportType	Especifique los tipos de entidades para la exportación en tiempo real (eventos, historial, tendencias).
ExternalScripts	Especifique la ubicación de los scripts externos.
Fping6Location	Especifique la ubicación de fping6.
FpingLocation	Especifique la ubicación de fping.
FrontendAllowedIP	Especifique las direcciones IP o los rangos CIDR desde los que se permite conectarse al frontend.
HANodeName	Especifique el nombre del nodo en un clúster de alta disponibilidad.
HistoryCacheSize	Establezca el tamaño de la caché del historial.
HistoryIndexCacheSize	Establezca el tamaño de la caché de índices del historial.
HistoryProvider	Establezca el proveedor de almacenamiento del historial (Elasticsearch, ClickHouse o SQL).

Parámetro	Descripción
HistoryStorageDateIndex	Este parámetro está **obsoleto** (utilice HistoryProvider en su lugar). Habilite el almacenamiento de los valores del historial en diferentes índices según la fecha.
HistoryStorageURL	Este parámetro está **obsoleto** (utilice HistoryProvider en su lugar). Especifique la URL del almacenamiento del historial.
HistoryStorageTypes	Este parámetro está **obsoleto** (utilice HistoryProvider en su lugar). Especifique los tipos de valores que se enviarán al almacenamiento del historial.
HousekeepingFrequency	Establezca la frecuencia del procedimiento de mantenimiento (en horas).
Include	Especifique un directorio o archivos individuales que se incluirán en el archivo de configuración.
JavaGateway	Especifique la dirección IP (o el nombre de host) del Java gateway de Zabbix.
JavaGatewayPort	Especifique el puerto del Java gateway de Zabbix.
ListenBacklog	Establezca el número máximo de conexiones pendientes en la cola TCP.
ListenIP	Especifique la dirección IP de escucha del trapper.
ListenPort	Especifique el puerto de escucha del trapper.
LoadModule	Especifique el módulo que se cargará al iniciar el server.
LoadModulePath	Especifique la ruta completa a la ubicación de los módulos del server.
LogFile	Especifique el archivo de registro del server.
LogFileSize	Establezca el tamaño máximo del archivo de registro.
LogSlowQueries	Establezca la duración máxima que puede tener una consulta de base de datos antes de registrarse (en milisegundos).
LogType	Especifique el tipo de salida del registro.
MaxConcurrentChecksPerPoller	Establezca el número máximo de comprobaciones asíncronas que cada poller de HTTP agent, poller de agent o poller SNMP puede ejecutar simultáneamente.
MaxHousekeeperDelete	Establezca el número máximo de filas que se eliminarán por tarea en un ciclo de mantenimiento.
NodeAddress	Especifique la IP o el nombre de host, con un puerto opcional, para anular la forma en que el frontend se conecta al server.
PidFile	Especifique el archivo PID.
ProblemHousekeepingFrequency	Establezca la frecuencia de eliminación de problemas correspondientes a triggers eliminados.
ProxyConfigFrequency	Establezca la frecuencia con la que el server envía datos de configuración al proxy de Zabbix.
ProxyDataFrequency	Establezca la frecuencia con la que el server solicita datos del historial al proxy de Zabbix.
ServiceManagerSyncFrequency	Establezca la frecuencia de sincronización de la configuración de un administrador de servicios.
SMSDevices	Especifique los archivos de módem que pueden utilizar el server de Zabbix.
SNMPTrapperFile	Especifique el archivo temporal utilizado para pasar datos del demonio de traps SNMP al server.
SocketDir	Especifique el directorio donde se almacenan los sockets IPC utilizados por los servicios internos de Zabbix.
SourceIP	Especifique la dirección IP de origen.
SSHKeyLocation	Especifique la ubicación de las claves públicas y privadas para las comprobaciones y acciones SSH.
SSLCertLocation	Especifique la ubicación de los archivos de certificados de cliente SSL para la autenticación del cliente.
SSLKeyLocation	Especifique la ubicación de los archivos de claves privadas SSL para la autenticación del cliente.
SSLCALocation	Especifique la ubicación de los archivos de la autoridad certificadora (CA) para la verificación del certificado del server SSL.
StartAgentPollers	Establezca el número de instancias prefork de pollers asíncronos de Zabbix agent.
StartAlerters	Establezca el número de instancias prefork de alerters.
StartBrowserPollers	Establezca el número de instancias prefork de pollers de items de navegador.
StartConnectors	Establezca el número de instancias prefork de workers de conectores.
StartDBSyncers	Establezca el número de instancias prefork de sincronizadores del historial.
StartDiscoverers	Establezca el número de instancias prefork de workers de descubrimiento.
StartEscalators	Establezca el número de instancias prefork de escaladores.
StartHistoryPollers	Establezca el número de instancias prefork de pollers del historial.
StartHTTPAgentPollers	Establezca el número de instancias prefork de pollers asíncronos de HTTP agent.
StartHTTPPollers	Establezca el número de instancias prefork de pollers HTTP.
StartIPMIPollers	Establezca el número de instancias prefork de pollers IPMI.
StartJavaPollers	Establezca el número de instancias prefork de pollers Java.
StartLLDProcessors	Establezca el número de instancias prefork de workers de descubrimiento de bajo nivel (LLD).
StartODBCPollers	Establezca el número de instancias prefork de pollers ODBC.
StartPingers	Establezca el número de instancias prefork de pingers ICMP.
StartPollersUnreachable	Establezca el número de instancias prefork de pollers para hosts inaccesibles (incluidos IPMI y Java).
StartPollers	Establezca el número de instancias prefork de pollers.
StartPreprocessors	Establezca el número de instancias preiniciadas de workers de preprocesamiento.

Parámetro	Descripción
StartProxyPollers	Establezca el número de instancias prefork de pollers para proxies pasivos.
StartReportWriters	Establezca el número de instancias prefork de redactores de informes.
StartSNMPPollers	Establezca el número de instancias prefork de pollers SNMP asíncronos.
StartSNMPTrapper	Establézcalo en "1" para iniciar un proceso trapper SNMP.
StartTelemetryQueryPollers	Establezca el número de instancias prefork de pollers asíncronos de consultas de telemetría.
StartTimers	Establezca el número de instancias prefork de temporizadores.
StartTrappers	Establezca el número de instancias prefork de trappers.
StartVMwareCollectors	Establezca el número de instancias prefork de recopiladores de VMware.
StatsAllowedIP	Especifique las direcciones IP o los nombres DNS de las instancias externas de Zabbix que pueden solicitar estadísticas.
TelemetryProvider	Establezca la fuente de datos de telemetría.
Timeout	Especifique cuánto tiempo esperar (en segundos) para establecer la conexión e intercambiar datos con el proxy, agent, servicio web de Zabbix y para las comprobaciones SNMP (excepto los items SNMP `walk[OID]` y `get[OID]`).
TLSCAFile	Especifique la ruta completa del archivo que contiene los certificados de las CA de nivel superior para la verificación de certificados de pares, utilizado para las comunicaciones cifradas entre componentes de Zabbix.
TLSCertFile	Especifique la ruta completa del archivo que contiene el certificado o la cadena de certificados del server, utilizado para las comunicaciones cifradas entre componentes de Zabbix.
TLSCipherAll	Especifique la cadena de prioridad de GnuTLS o la cadena de cifrado de OpenSSL (TLS 1.2). Anule los criterios predeterminados de selección de suites de cifrado para el cifrado basado en certificados y PSK.
TLSCipherAll13	Especifique la cadena de cifrado para OpenSSL 1.1.1 o posterior en TLS 1.3. Anule los criterios predeterminados de selección de suites de cifrado para el cifrado basado en certificados y PSK.
TLSCipherCert	Especifique la cadena de prioridad de GnuTLS o la cadena de cifrado de OpenSSL (TLS 1.2). Anule los criterios predeterminados de selección de suites de cifrado para el cifrado basado en certificados.
TLSCipherCert13	Especifique la cadena de cifrado para OpenSSL 1.1.1 o posterior en TLS 1.3. Anule los criterios predeterminados de selección de suites de cifrado para el cifrado basado en certificados.
TLSCipherPSK	Especifique la cadena de prioridad de GnuTLS o la cadena de cifrado de OpenSSL (TLS 1.2). Anule los criterios predeterminados de selección de suites de cifrado para el cifrado basado en PSK.
TLSCipherPSK13	Especifique la cadena de cifrado para OpenSSL 1.1.1 o posterior en TLS 1.3. Anule los criterios predeterminados de selección de suites de cifrado para el cifrado basado en PSK.
TLSCRLFile	Especifique la ruta completa del archivo que contiene los certificados revocados. Este parámetro se utiliza para las comunicaciones cifradas entre componentes de Zabbix.
TLSFrontendAccept	Especifique el nivel de cifrado para las conexiones entrantes desde el frontend.
TLSFrontendCertIssuer	Especifique el emisor de certificado permitido para el frontend.
TLSFrontendCertSubject	Especifique el asunto de certificado permitido para el frontend.
TLSKeyFile	Especifique la ruta completa del archivo que contiene la clave privada del server, utilizada para las comunicaciones cifradas entre componentes de Zabbix.
TLSListen	Exija conexiones exclusivamente TLS en el socket del trapper.
TmpDir	Especifique el directorio temporal.
TrapperTimeout	Especifique el tiempo de espera del trapper, en segundos, para: - recuperar datos históricos del proxy de Zabbix; - enviar datos de configuración al proxy de Zabbix; - ejecutar scripts globales o comandos remotos en el server de Zabbix.
TrendCacheSize	Establezca el tamaño de la caché de tendencias.
TrendFunctionCacheSize	Establezca el tamaño de la caché de funciones de tendencias.
UnavailableDelay	Establezca la frecuencia de comprobación de la disponibilidad del host durante el período de indisponibilidad.
UnreachableDelay	Establezca la frecuencia de comprobación de la disponibilidad del host durante el período de inaccesibilidad.
UnreachablePeriod	Establezca después de cuántos segundos de inaccesibilidad se debe considerar que el host no está disponible.
User	Abandone los privilegios y utilice un usuario específico existente en el sistema.
ValueCacheSize	Establezca el tamaño de la caché de valores del historial.
Vault	Especifique el proveedor de vault.
VaultAppRoleID	Especifique el ID de AppRole del vault de HashiCorp.
VaultAppSecretID	Especifique el ID secreto de AppRole del vault de HashiCorp.
VaultDBPath	Especifique la ubicación para recuperar las credenciales de la base de datos mediante claves.
VaultPrefix	Especifique un prefijo personalizado para la ruta o consulta del vault.

Parámetro	Descripción
VaultTLSCertFile	Especifique el nombre del archivo de certificado SSL utilizado para la autenticación del cliente.
VaultTLSKeyFile	Especifique el nombre del archivo de clave privada SSL utilizado para la autenticación del cliente.
VaultToken	Especifique el token de autenticación del vault de HashiCorp.
VaultURL	Especifique la URL del server de vault.
VMwareCacheSize	Establezca el tamaño de la memoria compartida para almacenar datos de VMware.
VMwareFrequency	Establezca la frecuencia de recopilación de datos de un único servicio VMware, en segundos.
VMwarePerfFrequency	Establezca la frecuencia de recuperación de estadísticas de contadores de rendimiento de un único servicio VMware, en segundos.
VMwareTimeout	Establezca el número máximo de segundos que un `vmware collector` esperará una respuesta del servicio VMware.
WebDriverURL	Especifique la URL de la interfaz WebDriver.
WebServiceURL	Especifique la URL del servicio web con el formato `http[s]://host:port/report`.

Tenga en cuenta lo siguiente:

- Los valores predeterminados reflejan los valores predeterminados del demonio, no los valores de los archivos de configuración distribuidos.
- Los valores admiten variables de entorno.
- Zabbix admite archivos de configuración únicamente con codificación UTF-8 y sin BOM.
- Los comentarios que comienzan con # solo se admiten al principio de la línea.

Warning:
Demasiados procesos pueden agotar el límite de descriptores de archivo por proceso del administrador de preprocesamiento.

Estos procesos corresponden a la recopilación de datos mediante pollers, pollers de hosts inaccesibles, pollers ODBC, pollers HTTP, pollers Java, pingers, trappers y proxypollers, junto con el administrador IPMI, el trapper SNMP, los workers de preprocesamiento y los workers de descubrimiento. Agotar el límite de descriptores de archivo hará que el server de Zabbix se **detenga**, normalmente poco después del inicio, aunque en ocasiones puede tardar más.

Para evitar estos problemas, revise el archivo de configuración del server de Zabbix para optimizar el número de comprobaciones y procesos simultáneos. Además, si es necesario, asegúrese de que el límite de descriptores de archivo sea suficientemente alto comprobando y ajustando los límites del sistema.

Detalles del parámetro

AlertScriptsPath

Especifique la ubicación del alert script. Depende de la variable de instalación en tiempo de compilación `datadir`.

Predeterminado: `/usr/local/share/zabbix/alertscripts`

AllowRoot

Permite que el server se ejecute como `root`. Si está deshabilitado y el server se inicia como `root`, el server intentará cambiar al usuario `zabbix`. No tiene efecto si se inicia con un usuario normal.

Predeterminado: 0
 Valores: 0 - no permitir; 1 - permitir

AllowSoftwareUpdateCheck

Permitir que la interfaz de usuario de Zabbix reciba información sobre actualizaciones de software desde `zabbix.com`.

Predeterminado: 1
 Valores: 0 - no permitir; 1 - permitir

AllowUnsupportedDBVersions

Permitir que el server funcione con versiones de base de datos no compatibles.

Predeterminado: 0
 Valores: 0 - no permitir; 1 - permitir

BridgeAdapterURL

Establece la URL HTTP(S) del adaptador de puente de Zabbix.

Ejemplo:

```
BridgeAdapterURL=https://bridge-adapter.example.com:10054/rpc
```

CacheSize

Establezca el tamaño de la caché de configuración (en bytes). El tamaño de la memoria compartida para almacenar datos de host, item y trigger.

Predeterminado: 32M
 Rango: 128K-64G

CacheUpdateFrequency

Establezca la frecuencia de actualización de la caché de configuración (en segundos). Consulte también las opciones de runtime control.

Predeterminado: 10
 Rango: 1-3600

DBHost

Especifique el host de la base de datos (o el directorio del socket).

Para MySQL:

- `localhost` o una cadena vacía usa el socket UNIX de dominio predeterminado.

Para PostgreSQL:

- `localhost` se resuelve mediante DNS (normalmente a 127.0.0.1).
- Una cadena vacía usa el socket UNIX de dominio predeterminado.
- Una ruta (por ejemplo, `/var/run/pgbouncer`) da como resultado el uso del socket UNIX en esa ruta.
- Una lista separada por comas puede contener varios valores `host:port`. Cada host se prueba en orden hasta que se establece una conexión de lectura y escritura.

Ejemplo: `DBHost=localhost:5431,127.0.0.1:20051,zabbix.domain,[::1]:30051,[12fc::1]`

Predeterminado: `localhost`

DBName

Especifique el nombre de la base de datos.

Obligatorio: **Sí**

DBPassword

Especifique la contraseña de la base de datos. Comente esta línea si no se utiliza ninguna contraseña.

DBPort

Especifique el puerto del host de la base de datos de Zabbix (consulte `DBHost`).
 `DBSocket` y `DBPort` son mutuamente excluyentes en la configuración del server. Especifique solo uno, o deje ambos sin definir.

Valor predeterminado para MySQL: 3306

Valor predeterminado para PostgreSQL: 5432

Rango: 1024-65535

DBSchema

Especifique el nombre del esquema de la base de datos. Se utiliza para PostgreSQL.

DBSocket

Especifique la ruta al archivo de socket de MySQL.
 `DBSocket` y `DBPort` se excluyen mutuamente en la configuración del server. Especifique solo uno, o deje ambos sin definir.

DBUser

Especifique el usuario de la base de datos.

DBTLSConnect

Especifique un valor para forzar una conexión TLS a la base de datos: - *required* - conectarse usando TLS; - *verify_ca* - conectarse usando TLS y verificar el certificado; - *verify_full* - conectarse usando TLS, verificar el certificado y verificar que la identidad de la base de datos especificada por `DBHost` coincida con su certificado.

Con MySQL, a partir de la versión 5.7.11, y PostgreSQL, se admiten los siguientes valores: `required`, `verify_ca`, `verify_full`.
 Con MariaDB, a partir de la versión 10.2.6, se admiten los valores `required` y `verify_full`.
De forma predeterminada, no se establece ninguna opción y el comportamiento depende de la configuración de la base de datos.

DBTLSCAFile

Especifique la ruta completa del archivo que contiene los certificados de la CA de nivel superior para la verificación del certificado de la base de datos.

Obligatorio: no (sí, si DBTLSConnect está configurado en `verify_ca` o `verify_full`)

DBTLSCertFile

Especifique la ruta completa del archivo que contiene el certificado del server de Zabbix para la autenticación en la base de datos.

DBTLSKeyFile

Especifique la ruta completa del archivo que contiene la clave privada para autenticarse en la base de datos.

DBTLSCipher

Especifique la lista de cifrados de cifrado que el server de Zabbix permite para los protocolos TLS hasta TLS v1.2. Solo es compatible con MySQL.

DBTLSCipher13

Especifique la lista de conjuntos de cifrado que el server de Zabbix permite para el protocolo TLS v1.3. Solo es compatible con MySQL, a partir de la versión 8.0.16.

DebugLevel

Establece el nivel de depuración: - *0* - información básica sobre el inicio y la detención de los procesos de Zabbix; - *1* - información crítica; - *2* - información de error; - *3* - advertencias; - *4* - para depuración (genera mucha información); - *5* - depuración extendida (genera aún más información).
 Consulte también las opciones de control en tiempo de ejecución.

Predeterminado: 3
 Rango: 0-5

EnableGlobalScripts

Habilita los scripts globales en el server de Zabbix.
 Nota: La ejecución de scripts globales está habilitada de forma predeterminada. Sin embargo, para nuevas instalaciones a partir de Zabbix 7.0, EnableGlobalScripts se establece explícitamente en 0 (deshabilitado).

Predeterminado: 1
 Valores: 0 - deshabilitar; 1 - habilitar

EnableMobileDevices

Habilita la compatibilidad con dispositivos móviles en el Zabbix server.

Cuando está habilitado, el Zabbix server procesará las solicitudes para agregar o eliminar dispositivos móviles y enviará notificaciones push a dichos dispositivos.

Valor predeterminado: 1
 Valores: 0 - deshabilitar; 1 - habilitar

ExportDir

Especifique un directorio para habilitar la exportación en tiempo real de eventos, historial y tendencias.

ExportFileSize

Establece el tamaño máximo del archivo de exportación (en bytes). Se usa para la rotación si `ExportDir` está configurado.

Predeterminado: 1G
 Rango: 1M-1G

ExportType

Especifique los tipos de entidad para la exportación en tiempo real (events, history, trends). Una lista delimitada por comas. De forma predeterminada, se exportan todos los tipos. Válido solo si `ExportDir` está configurado.
 Nota que si se especifica `ExportType`, pero `ExportDir` no, entonces esto es un error de configuración y el server no se iniciará.

Ejemplo para exportación de history y trends:

```
ExportType=history,trends
```

Ejemplo para exportación solo de events:

```
ExportType=events
```

ExternalScripts

Especifique la ubicación del script externo. Depende de la variable de instalación en tiempo de compilación `datadir`.

Predeterminado: `/usr/local/share/zabbix/externalscripts`

Fping6Location

Especifique la ubicación de fping6. Asegúrese de que el binario fping6 tenga como propietario a root y que el indicador SUID esté establecido. Déjelo vacío (`Fping6Location=`) si su utilidad fping es capaz de procesar direcciones IPv6.

Predeterminado: `/usr/sbin/fping6`

FpingLocation

Especifique la ubicación de fping. Asegúrese de que el binario de fping tenga como propietario a root y que el indicador SUID esté establecido.

Predeterminado: `/usr/sbin/fping`

FrontendAllowedIP

Especifique direcciones IP o rangos CIDR que tienen permiso para conectarse desde el frontend. Una lista separada por comas. De forma predeterminada, se aceptan todas las conexiones para las solicitudes del frontend. Si la compatibilidad con IPv6 está habilitada, entonces `127.0.0.1`, `::127.0.0.1`, `::ffff:127.0.0.1` se tratan por igual y `::/0` permitirá cualquier dirección IPv4 o IPv6. `0.0.0.0/0` se puede usar para permitir cualquier dirección IPv4.

Ejemplo:

```
FrontendAllowedIP=127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.example.com
```

HANodeName

Especifique el nombre del nodo en un clúster de alta disponibilidad. Cuando está vacío, el server funciona en modo independiente y se crea un nodo con nombre vacío.

HistoryCacheSize

Establece el tamaño de la caché de historial (en bytes). El tamaño de la memoria compartida para almacenar datos de historial.

Predeterminado: `16M`
 Rango: 128K-16G

HistoryIndexCacheSize

Establece el tamaño de la caché de índice del historial (en bytes). El tamaño de la memoria compartida para indexar los datos del historial almacenados en la caché del historial. El tamaño de la caché de índice necesita aproximadamente 100 bytes para almacenar en caché un item.

Predeterminado: `4M`
 Rango: 128K-16G

HistoryProvider

Establece el proveedor de almacenamiento de history usando el formato `<provider>[;<options>]`, donde:

- `<provider>` - uno de los proveedores de almacenamiento de history compatibles: `sql` (predeterminado), `elasticsearch` o `clickhouse`.
- `<options>` - pares `key=value` delimitados por comas. Los valores pueden ir entre comillas.

Opciones para todos los proveedores de almacenamiento de history:

- `value_types` - establece los tipos de valor que se almacenarán (`uint`, `dbl`, `str`, `log`, `text`, `json`). Cada uno puede usarse una sola vez y solo para un único proveedor.
- `precache` - establece 1 (predeterminado) para precargar en bloque los datos del proveedor de almacenamiento de history en la caché de valores. Esto puede mejorar significativamente el calentamiento de la caché de valores después de un reinicio.

Opciones para Elasticsearch y ClickHouse:

- `url` - establece la URL HTTP o HTTPS usada para conectarse al proveedor de almacenamiento de history.
- `log_slow_queries` - establece la duración máxima (en milisegundos) que una consulta al proveedor de almacenamiento de history puede tardar antes de registrarse. Esta opción se habilita a partir de `DebugLevel=3`.
- `source_ip` - establece la dirección IP de origen usada para las conexiones salientes al proveedor de almacenamiento de history.

Opciones para Elasticsearch:

- `date_index` - establece 1 para habilitar el almacenamiento de valores de history en distintos índices según la fecha.

Opciones para ClickHouse:

- `db` - establece el nombre de la base de datos.
- `username` - establece el nombre de usuario usado para iniciar sesión en la base de datos.
- `password` - establece la contraseña usada para iniciar sesión en la base de datos.

- `ssl_cert_file`, `ssl_key_file`, `ssl_key_password`, `ssl_verify_peer`, `ssl_verify_host`, `ssl_ca_location`, `ssl_cert_location`, `ssl_key_location` - establece estas opciones para configurar una conexión segura (TLS/SSL) a la base de datos.

Ejemplo para ClickHouse:

```
HistoryProvider=clickhouse;value_types="uint,dbl,str,log,text,json",url=http://localhost:8123,db=zabbix,us
```

Ejemplo para Elasticsearch:

```
HistoryProvider=elasticsearch;value_types="uint,dbl,str,log,text,json",url=http://localhost:9200,date_inde
```

Ejemplo para ClickHouse y Elasticsearch:

```
HistoryProvider=clickhouse;value_types="uint,dbl,str",url=http://localhost:8123,db=zabbix,username=zabbix,
HistoryProvider=elasticsearch;value_types="log,text,json",url=http://localhost:9200,date_index=0
```

HistoryStorageDateIndex

Este parámetro está obsoleto (use HistoryProvider en su lugar).
 Habilita el almacenamiento de valores de historial en diferentes índices según la fecha. Este parámetro se usa para la configuración de Elasticsearch.

Predeterminado: 0
 Valores: 0 - deshabilitar; 1 - habilitar

HistoryStorageURL

Este parámetro está obsoleto (use HistoryProvider en su lugar).
 Especifique la URL de almacenamiento del historial. Este parámetro se utiliza para la configuración de Elasticsearch.

HistoryStorageTypes

Este parámetro está obsoleto (use HistoryProvider en su lugar).
 Especifique los tipos de valores que se enviarán al almacenamiento de historial. Una lista delimitada por comas. Este parámetro se utiliza para la configuración de Elasticsearch.

Predeterminado: `uint,dbl,str,log,text,json`

HousekeepingFrequency

Establece la frecuencia del procedimiento de housekeeping (en horas). Housekeeping elimina la información obsoleta de la base de datos.
Nota: Para evitar que housekeeper se sobrecargue (por ejemplo, cuando los períodos de history y trend se reducen considerablemente), en un ciclo de housekeeping no se eliminan más de 4 veces las horas de `HousekeepingFrequency` de información obsoleta, para cada item. Por lo tanto, si `HousekeepingFrequency` es 1, en cada ciclo se eliminarán como máximo 4 horas de información obsoleta (empezando por la entrada más antigua).
Nota: Para reducir la carga al iniciar server, housekeeping se pospone durante 30 minutos después del arranque de server. Por lo tanto, si `HousekeepingFrequency` es 1, el primer procedimiento de housekeeping después del arranque de server se ejecutará tras 30 minutos y, a partir de entonces, se repetirá con un retraso de una hora.
Es posible desactivar el housekeeping automático estableciendo `HousekeepingFrequency` en 0. En este caso, el procedimiento de housekeeping solo puede iniciarse mediante la opción de control en tiempo de ejecución `housekeeper_execute`, y el período de información obsoleta eliminado en un ciclo de housekeeping es 4 veces el período transcurrido desde el último ciclo de housekeeping, pero no menos de 4 horas ni más de 4 días.
Consulte también las opciones de runtime control y los detalles sobre el housekeeping procedure.

Predeterminado: 1
 Rango: 0-24

Incluir

Especifique un directorio o archivos individuales que se incluirán en el archivo de configuración.
 Durante la instalación de Zabbix, este parámetro se establece en `/usr/local/etc`, a menos que lo cambie durante el tiempo de compilación.

Reglas generales para el parámetro `Include`:

- La ruta puede ser relativa a la ubicación del archivo `zabbix_server.conf`.
- Puede especificar varios parámetros `Include` en un solo archivo de configuración.
- Zabbix procesa el parámetro `Include` en cuanto lo lee, así que considere colocarlo al final del archivo de configuración.

Reglas para los archivos incluidos:

- Cada archivo debe ser legible.
- Cada archivo debe tener una sintaxis correcta. De lo contrario, el componente no se iniciará.

Reglas para incluir directorios o patrones de archivos:

- No asuma ningún orden de inclusión (por ejemplo, los archivos no se incluyen en orden alfabético).
- No defina el mismo parámetro en más de un archivo incluido (por ejemplo, para sobrescribir una configuración general con una específica), ya que no puede predecir qué valor se utilizará.

- Tenga cuidado con las copias de seguridad que algunos editores de texto crean automáticamente (por ejemplo, editar `include/my_specific.conf` puede crear `include/my_specific.conf.BAK`), ya que ambos archivos se incluirán.

Para incluir solo los archivos relevantes de un directorio, use el comodín (`*`) para la coincidencia de patrones.

Ejemplo:

```
Include=/absolute/path/to/config/files/*.conf
```

JavaGateway

Especifique la dirección IP o el nombre de host de Zabbix Java gateway. Solo es necesario si se han iniciado los Java pollers.

JavaGatewayPort

Especifique el puerto de Zabbix Java gateway.

Predeterminado: 10052
 Rango: 1024-32767

ListenBacklog

Establece el número máximo de conexiones pendientes en la cola TCP.
El valor predeterminado es una constante codificada, que depende del sistema.
El valor máximo admitido también depende del sistema; los valores demasiado altos pueden truncarse silenciosamente al ”máximo especificado por la implementación”.

Predeterminado: SOMAXCONN
 Rango: 0 - INT_MAX

ListenIP

Especifique la dirección IP de escucha del trapper. Una lista delimitada por comas.
 El trapper escuchará en todas las interfaces de red si no se especifica este parámetro.

Predeterminado: `0.0.0.0`

ListenPort

Especifique el puerto de escucha de trapper.

Predeterminado: 10051
 Rango: 1024-32767

LoadModule

Especifique el módulo que se cargará al iniciar el server. Los módulos se utilizan para ampliar la funcionalidad del server. El módulo debe encontrarse en el directorio especificado por `LoadModulePath` o la ruta debe preceder al nombre del módulo. Si la ruta precedente es absoluta (comienza con /), `LoadModulePath` se ignora.
 Formatos:

- `LoadModule=<module.so>`
- `LoadModule=<path/module.so>`
- `LoadModule=</abs_path/module.so>`

Se pueden especificar varios parámetros `LoadModule`.

LoadModulePath

Especifique la ruta completa a la ubicación de los módulos del server. El valor predeterminado depende de las opciones de compilación.

LogFile

Especifique el archivo de registro del server.

Obligatorio: Sí, si `LogType` está establecido en `file`; en caso contrario, no

LogFileSize

Establezca el tamaño máximo del archivo de registro (en MB).
0 - deshabilita la rotación automática del archivo de registro.
Nota: Si se alcanza el límite de tamaño del archivo de registro y la rotación del archivo falla, por cualquier motivo, el archivo de registro existente se trunca y se inicia uno nuevo.

Predeterminado: 1
 Rango: 0-1024
 Obligatorio: Sí, si `LogType` está establecido en `file`; de lo contrario, no

LogSlowQueries

Establece la duración máxima que una consulta de base de datos puede tardar antes de registrarse (en milisegundos).
0 - no registrar consultas lentas.
Esta opción se habilita a partir de DebugLevel=3.

Predeterminado: 0
 Rango: 0-3600000

LogType

Especifique el tipo de salida del log: - *file* - escribe el log en el archivo especificado por el parámetro LogFile; - *system* - escribe el log en syslog; - *console* - escribe el log en la salida estándar.

Predeterminado: file

MaxConcurrentChecksPerPoller

Establece el número máximo de comprobaciones asíncronas que puede ejecutar a la vez cada HTTP agent poller, agent poller o SNMP poller. Consulte StartHTTPAgentPollers, StartAgentPollers y StartSNMPPollers.

Predeterminado: 1000
 Rango: 1-1000

MaxHousekeeperDelete

Establece el número máximo de filas que se eliminarán por tarea en un ciclo de housekeeping de las tablas history, history_str, history_log, history_uint, history_text, history_bin, history_json, trends, trends_uint y problem.
Las tareas de housekeeping existen en la tabla housekeeper con el formato: [housekeeperid], [object], [objectid].
 Este parámetro se aplica solo a la eliminación de datos que quedaron de items ya eliminados.
 Si se establece en 0, no se usa ningún límite.
 En este caso, debe saber lo que está haciendo para no sobrecargar la base de datos.
 Cuando se eliminan muchos items, aumenta la carga de la base de datos, porque el housekeeper tendrá que eliminar todos los datos históricos que tenían esos items.
 Por ejemplo, si solo tenemos que eliminar 1 prototipo de item de la template, pero esta template está vinculada a 50 hosts y para cada host el prototipo se expande a 100 items reales, en total hay que eliminar 5000 items (1*50*100).
 Si se establece 500 para MaxHousekeeperDelete (MaxHousekeeperDelete=500), el proceso housekeeper tendrá que eliminar hasta 2500000 valores (5000*500) de los items eliminados de las tablas history y trends en un ciclo.

Consulte también los detalles sobre el procedimiento de housekeeping.

Predeterminado: 5000
 Rango: 0-1000000

NodeAddress

Especifique una IP o nombre de host con un puerto opcional para anular la forma en que el frontend se conecta al server.
Formato: <address>[:<port>]

Si no se establece IP o nombre de host, se utilizará el valor de ListenIP. Si ListenIP no está establecido, se utilizará el valor localhost.
Si no se establece el puerto, se utilizará el valor de ListenPort. Si ListenPort no está establecido, se utilizará el valor 10051.

Esta opción puede ser anulada por la dirección especificada en la configuración del frontend.

Véase también: parámetro HANodeName; Habilitación de alta disponibilidad.

Predeterminado: localhost:10051

PidFile

Especifique el archivo PID.

Predeterminado: /tmp/zabbix_server.pid

ProblemHousekeepingFrequency

Establece la frecuencia de eliminación de problemas para triggers eliminados (en segundos).
 Consulte también los detalles sobre el procedimiento de housekeeping.

Predeterminado: 60
 Rango: 1-3600

ProxyConfigFrequency

Establece la frecuencia de envío de datos de configuración al proxy de Zabbix por parte del server. Se usa solo para proxies en modo pasivo.

Predeterminado: 10
 Rango: 1-604800

ProxyDataFrequency

Establece la frecuencia con la que el server solicita datos históricos al proxy de Zabbix. Se usa solo para proxies en modo pasivo.

Predeterminado: 1
 Rango: 1-3600

ServiceManagerSyncFrequency

Establece la frecuencia de sincronización de la configuración de un service manager (en segundos).

Predeterminado: 60
 Rango: 1-3600

SMSDevices

Especifique los archivos de módem que tienen permiso para usar el server de Zabbix. Una lista delimitada por comas.
 El envío de SMS no es posible si este parámetro no está configurado.

Ejemplo:

```
SMSDevices=/dev/ttyUSB0,/dev/ttyUSB1
```

SNMPTrapperFile

Especifique el archivo temporal utilizado para pasar datos desde el daemon de SNMP trap al server.
Debe ser el mismo que en `zabbix_trap_receiver.pl` o en el archivo de configuración de SNMPTT.

Predeterminado: `/tmp/zabbix_traps.tmp`

SocketDir

Especifique el directorio para almacenar los sockets IPC utilizados por los servicios internos de Zabbix.

Predeterminado: `/tmp`

SourceIP

Especifique la dirección IP de origen para:

- conexiones salientes a Zabbix proxy y Zabbix agent
- conexiones sin agent (VMware, SSH, JMX, SNMP, Telnet y comprobaciones simples)
- conexiones de HTTP agent
- solicitudes HTTP de JavaScript de item de script
- solicitudes HTTP de JavaScript de preprocesamiento
- envío de correos electrónicos de notificación (conexiones al server SMTP)
- notificaciones de webhook (conexiones HTTP de JavaScript)
- conexiones al Vault

SSHKeyLocation

Especifique la ubicación de las claves pública y privada para las comprobaciones y acciones de SSH.

SSLCertLocation

Especifique la ubicación de los archivos de certificado de cliente SSL para la autenticación del cliente.
 Este parámetro se utiliza en la monitorización web, para la comunicación con Vault, para el proveedor de almacenamiento del historial ClickHouse y para el origen de datos de telemetría ClickHouse.

SSLKeyLocation

Especifique la ubicación de los archivos de clave privada SSL para la autenticación del cliente.
 Este parámetro se utiliza en la monitorización web, para la comunicación con Vault, para el proveedor de almacenamiento del historial ClickHouse y para el origen de datos de telemetría ClickHouse.

SSLCALocation

Especifique la ubicación de los archivos de la autoridad certificadora (CA) para la verificación de certificados del servidor SSL. Si no se establece, se utilizará el directorio de todo el sistema.
 Tenga en cuenta que el valor de este parámetro se establecerá como la opción de libcurl `CURLOPT_CAPATH`. En las versiones de libcurl anteriores a la 7.42.0, esto solo tiene efecto si libcurl se compiló para utilizar OpenSSL. Para obtener más información, consulte la página web de cURL.
 Este parámetro se utiliza en la supervisión web, la autenticación SMTP, los items de HTTP agent, para la comunicación con Vault, para el proveedor de almacenamiento del historial ClickHouse y para el origen de datos de telemetría ClickHouse.

StartAgentPollers

Establece el número de instancias pre-forked de Zabbix agent pollers. Consulte MaxConcurrentChecksPerPoller.

Predeterminado: 1
 Rango: 0-1000

StartAlerters

Establece el número de instancias pre-forked de alerters.

Predeterminado: 3
 Rango: 1-100

StartBrowserPollers

Establece el número de instancias pre-forked de browser item pollers.

Predeterminado: 1
 Rango: 0-1000

StartConnectors

Establece el número de instancias preforked de connector workers. El proceso connector manager se inicia automáticamente cuando se inicia un connector worker.

Predeterminado: 0
 Rango: 0-1000

StartDBSyncers

Establece el número de instancias pre-forked de history syncers.
Nota: Tenga cuidado al cambiar este valor; aumentarlo puede hacer más daño que beneficio. Aproximadamente, el valor predeterminado debería ser suficiente para manejar hasta 4000 NVPS.

Predeterminado: 4
 Rango: 1-100

StartDiscoverers

Establece el número de instancias pre-forked de discovery workers[1].

Predeterminado: 5
 Rango: 0-1000

StartEscalators

Establece el número de instancias pre-forked de escalators.

Predeterminado: 1
 Rango: 1-100

StartHistoryPollers

Establece el número de instancias pre-forked de history pollers.
Solo es necesario para calculated checks.

Predeterminado: 5
 Rango: 0-1000

StartHTTPAgentPollers

Establece el número de instancias pre-forked de HTTP agent pollers. Consulte MaxConcurrentChecksPerPoller.

Predeterminado: 1
 Rango: 0-1000

StartHTTPPollers

Establece el número de instancias pre-forked de HTTP pollers[1].

Predeterminado: 1
 Rango: 0-1000

StartIPMIPollers

Establece el número de instancias pre-forked de IPMI pollers.

Predeterminado: 0
 Rango: 0-1000

StartJavaPollers

Establece el número de instancias preforkeadas de Java pollers[1].

Predeterminado: 0
 Rango: 0-1000

StartLLDProcessors

Establece el número de instancias preforked de low-level discovery (LLD) workers[1].
 El proceso del gestor de LLD se inicia automáticamente cuando se inicia un worker de LLD.

Predeterminado: 2
 Rango: 1-100

StartODBCPollers

Establece el número de instancias pre-forked de ODBC pollers[1].

Predeterminado: 1
 Rango: 0-1000

StartPingers

Establece el número de instancias pre-forked de ICMP pingers[1].

Predeterminado: 1
 Rango: 0-1000

StartPollersUnreachable

Establece el número de instancias pre-forked de pollers para hosts inaccesibles (incluidos IPMI y Java)[1].
 Debe estar en ejecución al menos un poller para hosts inaccesibles si se inician pollers normales, IPMI o Java.

Predeterminado: 1
 Rango: 0-1000

StartPollers

Establece el número de instancias pre-forked de pollers[1].

Predeterminado: 5
 Rango: 0-1000

StartPreprocessors

Establezca el número de instancias de preinicio de los hilos de worker[1] de preprocesamiento en un valor no inferior al número de núcleos de CPU disponibles. Se deben configurar más workers si el preprocesamiento no está limitado por la CPU y realiza muchas solicitudes de red.

Predeterminado: 16
 Rango: 1-1000

StartProxyPollers

Establece el número de instancias preforkeadas de pollers para proxies pasivos[1].

Predeterminado: 1
 Rango: 0-250

StartReportWriters

Establece el número de instancias pre-forked de report writers.
Si se establece en 0, la generación programada de informes se deshabilita.
El proceso del gestor de informes se inicia automáticamente cuando se inicia un report writer.

Predeterminado: 0
 Rango: 0-100

StartSNMPPollers

Establece el número de instancias pre-forked de SNMP pollers. Consulte `MaxConcurrentChecksPerPoller`.

Predeterminado: 1
 Rango: 0-1000

StartSNMPTrapper

Establézcalo en 1 para iniciar un proceso de SNMP trapper.

Predeterminado: 0
 Rango: 0-1

StartTelemetryQueryPollers

Establece el número de instancias precreadas de pollers de consultas de Telemetry asíncronas. Consulta MaxConcurrentChecksPerPoller.

Valor predeterminado: 1
 Rango: 0-1000

StartTimers

Establece el número de instancias preforked de timers.
Los timers procesan los periodos de mantenimiento.

Predeterminado: 1
 Rango: 1-1000

StartTrappers

Establece el número de instancias pre-forked de trappers[1].
 Los trappers aceptan conexiones entrantes de Zabbix sender, active agents y active proxies.

Predeterminado: 5
 Rango: 0-1000

StartVMwareCollectors

Establece el número de instancias de VMware collector pre-forked.

Predeterminado: 0
 Rango: 0-250

StatsAllowedIP

Especifique direcciones IP (opcionalmente en notación CIDR) o nombres DNS de instancias externas de Zabbix que tienen □□□□□□ para solicitar estadísticas. Una lista separada por comas.
 Si este parámetro no está establecido, no se aceptarán solicitudes de estadísticas.
 Si la compatibilidad con IPv6 está habilitada, entonces `127.0.0.1`, `::127.0.0.1`, `::ffff:127.0.0.1` se tratan por igual y `::/0` permitirá cualquier dirección IPv4 o IPv6. `0.0.0.0/0` puede usarse para permitir cualquier dirección IPv4.

Ejemplo:

```
StatsAllowedIP=127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.example.com
```

TelemetryProvider

Establezca la fuente de datos de telemetría con el formato `<provider>[;<options>]`, donde:

- `<provider>` - una de las fuentes de datos de telemetría compatibles: `clickhouse`
- `<options>` - pares `key=value` delimitados por comas. Los valores pueden estar entre comillas.

Opciones para ClickHouse:

- `url` - establece la URL HTTP o HTTPS utilizada para conectarse a la fuente de datos de telemetría.

- db - establece el nombre de la base de datos.
- username - establece el nombre de usuario utilizado para iniciar sesión en la base de datos.
- password - establece la contraseña utilizada para iniciar sesión en la base de datos.
- ssl_cert_file, ssl_key_file, ssl_key_password, ssl_verify_peer, ssl_verify_host, ssl_ca_location, ssl_cert_location, ssl_key_location - establece estas opciones para configurar una conexión segura (TLS/SSL) a la base de datos.
- source_ip - establece la dirección IP de origen utilizada para las conexiones salientes a la fuente de datos de telemetría.
- vault_path - establece la ruta a un secreto de Vault que contiene los valores de username y password. No lo utilice junto con las opciones username y password.

Ejemplo para ClickHouse:

```
TelemetryProvider=clickhouse;url=http://localhost:8123,db=zabbix,username=zabbix,password=zabbix
```

Ejemplo para ClickHouse con Vault:

```
TelemetryProvider=clickhouse;url=http://localhost:8123,db=zabbix,vault_path=clickhouse
```

Tiempo de espera

Especifique cuánto tiempo esperar (en segundos) para establecer la conexión e intercambiar datos con Zabbix proxy, agent, web service y comprobaciones SNMP heredadas (número OID único o cadena).

Este parámetro define la duración de varias operaciones de comunicación:

- ejecución de comandos remotos en Zabbix agent
- ejecución de comandos SSH/Telnet
- solicitudes al web service de Zabbix
- tiempo de espera de comunicación para la solicitud de prueba de mediatype y el método history.push
- reprogramación de items cuando una interfaz IPMI deja de estar disponible
- envío de respuesta a Zabbix proxy cuando falla el intercambio de datos debido a problemas de permisos o cifrado
- plazo para sockets asíncronos IPC y opciones de control en tiempo de ejecución
- conexiones JMX
- obtención de estadísticas desde un Zabbix proxy o server remoto
- envío de respuestas a Zabbix frontend
- solicitudes DNS del sondeador asíncrono
- respuesta para el latido de comprobación activa
- recuperación de datos (valores) de Zabbix agent desde agentes activos
- recuperación de datos desde Zabbix sender
- respuesta cuando trapper no puede procesar una solicitud
- envío de la lista de comprobaciones activas a Zabbix agent

Este tiempo de espera **no** se usará para aquellas comprobaciones que tengan configurados ajustes de tiempo de espera flexible en el frontend (a nivel global, de proxy o por item). Por ejemplo, los items SNMP walk[OID] y get[OID] usan el tiempo de espera configurado en el frontend; las comprobaciones SNMP heredadas siguen usando el valor de tiempo de espera del server.

Predeterminado: 3
 Rango: 1-30

TLSCAFile

Especifique la ruta completa del archivo que contiene los certificados de la CA de nivel superior para la verificación del certificado del par, utilizados para las comunicaciones cifradas entre los componentes de Zabbix.

TLSCertFile

Especifique la ruta completa del archivo que contiene el certificado del server o la cadena de certificados, utilizado para las comunicaciones cifradas entre los componentes de Zabbix.

TLSCipherAll

Especifique la cadena de prioridad de GnuTLS o la cadena de cifrado de OpenSSL (TLS 1.2). Anule los criterios de selección de la suite de cifrado predeterminada para el cifrado basado en certificados y PSK.

Ejemplo:

```
TLS_AES_256_GCM_SHA384:TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
```

TLSCipherAll13

Especifique la cadena de cifrado para OpenSSL 1.1.1 o posterior en TLS 1.3. Anule los criterios predeterminados de selección de ciphersuite para el cifrado basado en certificados y PSK.

Ejemplo para GnuTLS:

```
NONE:+VERS-TLS1.2:+ECDHE-RSA:+RSA:+ECDHE-PSK:+PSK:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL
```

Ejemplo para OpenSSL:

```
EECDH+aRSA+AES128:RSA+aRSA+AES128:kECDHEPSK+AES128:kPSK+AES128
```

TLSCipherCert

Especifique la cadena de prioridad de GnuTLS o la cadena de cifrado de OpenSSL (TLS 1.2). Anule los criterios de selección predeterminados del conjunto de cifrado para el cifrado basado en certificados.

Ejemplo para GnuTLS:

```
NONE:+VERS-TLS1.2:+ECDHE-RSA:+RSA:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL:+COMP-NULL:+SIG
```

Ejemplo para OpenSSL:

```
EECDH+aRSA+AES128:RSA+aRSA+AES128
```

TLSCipherCert13

Especifique la cadena de cifrado para OpenSSL 1.1.1 o una versión posterior en TLS 1.3. Anule los criterios de selección predeterminados de la suite de cifrado para el cifrado basado en certificados.

TLSCipherPSK

Especifique la cadena de prioridad de GnuTLS o la cadena de cifrado de OpenSSL (TLS 1.2). Anule los criterios de selección de la suite de cifrado predeterminada para el cifrado basado en PSK.

Ejemplo para GnuTLS:

```
NONE:+VERS-TLS1.2:+ECDHE-PSK:+PSK:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL:+COMP-NULL:+SIG
```

Ejemplo para OpenSSL:

```
kECDHEPSK+AES128:kPSK+AES128
```

TLSCipherPSK13

Especifique la cadena de cifrado para OpenSSL 1.1.1 o una versión posterior en TLS 1.3. Anule los criterios de selección predeterminados del conjunto de cifrado para el cifrado basado en PSK.

Ejemplo:

```
TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
```

TLSCRLFile

Especifique la ruta completa del archivo que contiene los certificados revocados. Este parámetro se utiliza para las comunicaciones cifradas entre los componentes de Zabbix.

TLSFrontendAccept

Especifique el nivel de cifrado para las conexiones entrantes desde el frontend.

Se pueden especificar varios valores, separados por comas:

- unencrypted - aceptar conexiones sin cifrado.
- cert - aceptar conexiones protegidas con TLS y un certificado.

Predeterminado: `unencrypted`

TLSFrontendCertIssuer

Especifique el emisor permitido del certificado del frontend.

TLSFrontendCertSubject

Especifique el subject del certificado permitido para el frontend.

TLSKeyFile

Especifique la ruta completa del archivo que contiene la clave privada del server, utilizada para las comunicaciones cifradas entre los componentes de Zabbix.

TLSListen

Requiere conexiones solo TLS en el socket trapper.

Valores admitidos:

- required - aceptar solo conexiones TLS

TmpDir

Especifique el directorio temporal.

Predeterminado: `/tmp`

TrapperTimeout

Especifique el tiempo de espera de trapper en segundos para:

- la recuperación de datos históricos desde Zabbix proxy
- el envío de datos de configuración a Zabbix proxy
- la ejecución de scripts globales o la ejecución de comandos remotos en Zabbix server

Predeterminado: 300
 Rango: 1-300

TrendCacheSize

Establece el tamaño de la caché de tendencias, en bytes.
El tamaño de la memoria compartida para almacenar tendencias.

Predeterminado: `4M`
 Rango: 128K-16G

TrendFunctionCacheSize

Establece el tamaño de la caché de trend-function, en bytes.
El tamaño de la memoria compartida para almacenar en caché los datos calculados de trend-function.

Predeterminado: `4M`
 Rango: 128K-2G

UnavailableDelay

Establece la frecuencia de comprobación de la disponibilidad del host durante el período de indisponibilidad en segundos.

Predeterminado: 60
 Rango: 1-3600

UnreachableDelay

Establece la frecuencia de comprobación de la disponibilidad del host durante el período de inaccesibilidad en segundos.

Predeterminado: 15
 Rango: 1-3600

UnreachablePeriod

Establece después de cuántos segundos de inaccesibilidad tratar al host como no disponible.

Predeterminado: 45
 Rango: 1-3600

Usuario

Reduce los privilegios a un usuario específico y existente en el sistema.
Solo tiene efecto si se ejecuta como 'root' y AllowRoot está deshabilitado.

Predeterminado: `zabbix`

ValueCacheSize

Establece el tamaño de la caché de valores del historial (en bytes).
 El tamaño de la memoria compartida para almacenar en caché las solicitudes de datos del historial de los item.
 Establecerlo en 0 deshabilita la caché de valores (no recomendado).
 Cuando la caché de valores se queda sin memoria compartida, se escribe un mensaje de advertencia en el registro del server cada 5 minutos.

Valor predeterminado: `8M`
 Rango: 0, 128K-64G

Vault

Especifique el proveedor de vault: - *HashiCorp* - HashiCorp KV Secrets Engine versión 2; - *CyberArk* - CyberArk Central Credential Provider
Debe coincidir con el proveedor de vault configurado en el frontend.

Predeterminado: `HashiCorp`

VaultAppRoleID

Especifique el AppRole ID de HashiCorp vault.
 Si se establece, habilita la autenticación AppRole. No se puede usar con VaultToken.
 Solo es compatible con HashiCorp vault.

VaultAppSecretID

Especifique el secret ID de AppRole de HashiCorp Vault.
 Compatible solo con HashiCorp Vault.

VaultDBPath

Especifique la ubicación para recuperar las credenciales de la base de datos mediante claves. Es una ruta o consulta de Vault, según el Vault.

Las claves utilizadas para HashiCorp son `password` y `username`.

Ejemplo de ruta con `VaultPrefix=/v1/secret/data/zabbix/`:

```
database
```

Ejemplo de ruta sin VaultPrefix:

```
secret/zabbix/database
```

Las claves utilizadas para CyberArk son `Content` y `UserName`.

Ejemplo:

```
AppID=zabbix_server&Query=Safe=passwordSafe;Object=zabbix_server_database
```

Esta opción solo se puede usar si `DBUser` y `DBPassword` no están especificados.

VaultPrefix

Especifique un prefijo personalizado para la ruta o la consulta de Vault, según el Vault. Se usarán los valores predeterminados más adecuados si no se especifica.
 Tenga en cuenta que `data` se agrega automáticamente después del punto de montaje para HashiCorp si no se especifica `VaultPrefix`.

Ejemplo de prefijo para HashiCorp:

```
/v1/secret/data/zabbix/
```

Ejemplo de prefijo para Cyberark:

```
/AIMWebService/api/Accounts?
```

VaultTLSCertFile

Especifique el nombre del archivo de certificado SSL utilizado para la autenticación del cliente. El archivo de certificado debe estar en formato PEM1.
 Si el archivo de certificado también contiene la clave privada, deje vacío el campo del archivo de clave SSL.
 El directorio que contiene este archivo se especifica mediante el parámetro de configuración SSLCertLocation.
 Esta opción puede omitirse, pero se recomienda para el vault CyberArkCCP.

VaultTLSKeyFile

Especifique el nombre del archivo de clave privada SSL utilizado para la autenticación del cliente. El archivo de clave privada debe estar en formato PEM1.
El directorio que contiene este archivo se especifica mediante el parámetro de configuración SSLKeyLocation.
Esta opción puede omitirse, pero se recomienda para CyberArkCCP vault.

VaultToken

Especifique el token de autenticación de HashiCorp Vault. El token de autenticación de HashiCorp Vault debe haberse generado exclusivamente para el server de Zabbix con permiso de solo lectura para las rutas especificadas en Vault macros y permiso de solo lectura para la ruta especificada en el parámetro de configuración opcional VaultDBPath.
 Se produce un error si `VaultToken` y la variable de entorno `VAULT_TOKEN` se definen al mismo tiempo.
 No se puede usar con VaultAppRoleID.
 Compatible solo con HashiCorp Vault.

VaultURL

Especifique la URL del servidor de vault. Se utilizará el directorio de certificados CA de todo el sistema si no se especifica SSLCALocation.

Predeterminado: `https://127.0.0.1:8200`

VMwareCacheSize

Establece el tamaño de la memoria compartida para almacenar datos de VMware. Se puede usar una comprobación interna de VMware `zabbix[vmware,buffer,...]` para supervisar el uso de la caché de VMware (consulte Internal checks).
Tenga en cuenta que la memoria compartida no se asigna si no hay instancias de `vmware collector` configuradas para iniciarse.

Predeterminado: 8M
 Rango: 256K-2G

VMwareFrequency

Establece la frecuencia de recopilación de datos de un único servicio VMware (en segundos). Esta frecuencia debe configurarse con el intervalo de actualización más corto de cualquier item de monitoreo VMware.

Predeterminado: 60
 Rango: 10-86400

VMwarePerfFrequency

Establece la frecuencia de recuperación de las estadísticas de contadores de rendimiento de un único servicio VMware (en segundos).
Esta frecuencia debe configurarse con el intervalo de actualización más corto de cualquier item de supervisión de VMware que utilice contadores de rendimiento de VMware.

Predeterminado: 60
 Rango: 10-86400

VMwareTimeout

Establece el número máximo de segundos que un `vmware collector` esperará una respuesta del servicio VMware (vCenter o hipervisor ESX).

Predeterminado: 10
 Rango: 1-300

WebDriverURL

Especifique la URL de la interfaz de WebDriver.

Ejemplo (usado con Selenium WebDriver standalone server):

```
WebDriverURL=http://localhost:4444
```

WebServiceURL

Especifique la URL del servicio web en el formato `http[s]://host:port/report`.

Ejemplo:

```
WebServiceURL=http://localhost:10053/report
```

El esquema `http://` puede omitirse solo para conexiones sin TLS (HTTP); con TLS configurado, debe usarse `https://`.

Notas al pie

[1] Tenga en cuenta que demasiados procesos de recopilación de datos (pollers, unreachable pollers, ODBC pollers, HTTP pollers, Java pollers, pingers, trappers, proxypollers) junto con IPMI manager, SNMP trapper, preprocessing workers y discovery workers pueden agotar el límite de descriptores de archivo por proceso para el preprocessing manager.

Warning:
Agotar el límite de descriptores de archivo hará que el server de Zabbix se detenga, normalmente poco después del inicio, aunque a veces puede tardar más. Para evitar este tipo de problemas, revise el archivo de configuración del server de Zabbix para optimizar el número de comprobaciones y procesos simultáneos. Además, si es necesario, asegúrese de que el límite de descriptores de archivo esté configurado suficientemente alto comprobando y ajustando los límites del sistema.

Proxy

Descripción general El proxy de Zabbix es un proceso que puede recopilar datos de monitorización de uno o varios dispositivos monitorizados y enviar la información al server de Zabbix, trabajando esencialmente en nombre del server. Todos los datos recopilados se almacenan temporalmente de forma local y, posteriormente, se transfieren al server de Zabbix al que pertenece el proxy.

La implementación de un proxy es opcional, pero puede ser muy beneficiosa para distribuir la carga de un único server de Zabbix. Si solo los proxies recopilan datos, el procesamiento en el server requiere menos CPU y operaciones de E/S de disco.

Un proxy de Zabbix es la solución ideal para la monitorización centralizada de ubicaciones remotas, sucursales y redes sin administradores locales.

El proxy de Zabbix requiere una base de datos independiente.

Attention:
Tenga en cuenta que las bases de datos compatibles con el proxy de Zabbix son SQLite, MySQL y PostgreSQL.

Véase también:

- Monitorización distribuida mediante proxies
- Balanceo de carga y alta disponibilidad del proxy

- Sincronización de la configuración de monitorización
- Protocolo de intercambio de datos entre el server y el proxy
- Parámetros de configuración: proxy de Zabbix
- Configurar un proxy de Zabbix Cloud

Ejecutando proxy Si se instala como paquete

El proxy de Zabbix se ejecuta como un proceso demonio. El proxy puede iniciarse ejecutando:

```
systemctl start zabbix-proxy
```

Esto funcionará en la mayoría de los sistemas GNU/Linux. En otros sistemas puede que necesite ejecutar:

```
/etc/init.d/zabbix-proxy start
```

De manera similar, para detener/reiniciar/ver el estado del proxy de Zabbix, utilice los siguientes comandos:

```
systemctl stop zabbix-proxy
systemctl restart zabbix-proxy
systemctl status zabbix-proxy
```

Iniciar manualmente

Si lo anterior no funciona, debe iniciarlo manualmente. Busque la ruta al binario `zabbix_proxy` y ejecute:

```
zabbix_proxy
```

Puede usar los siguientes parámetros de línea de comandos con Zabbix proxy:

Parameter	Description
`-c --config <config-file>`	Ruta al archivo de configuración.
`-f --foreground`	Ejecutar Zabbix proxy en primer plano.
`-R --runtime-control <option>`	Realizar funciones administrativas.
`-T --test-config`	Validar el archivo de configuración y salir.
`-h --help`	Mostrar información de ayuda y salir.
`-V --version`	Mostrar el número de versión.

Ejemplos de ejecución de Zabbix proxy con parámetros de línea de comandos:

```
zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf
zabbix_proxy --help
zabbix_proxy -V
```

Control en tiempo de ejecución

Opciones de control en tiempo de ejecución:

Option	Description	Target
`config_cache_reload`	Recargar la caché de configuración. Se ignora si la caché se está cargando actualmente. El proxy activo de Zabbix se conectará al server de Zabbix y solicitará los datos de configuración. El proxy pasivo de Zabbix solicitará los datos de configuración al server de Zabbix la próxima vez que el server se conecte al proxy.	
`history_cache_clear=<target>`	Vaciar la caché de historial del item especificado por su ID. Afecta a todos los valores del item, excepto el primero y el último.	**target** - ID del item.
`diaginfo[=<section>]`	Recopilar información de diagnóstico en el archivo de registro del proxy.	`historycache` - estadísticas de la caché de historial; `preprocessing` - estadísticas del gestor de preprocesamiento; `locks` - lista de mutexes (está vacía en sistemas *BSD*).

Option	Description	Target
`snmp_cache_reload`	Recargar la caché SNMP: borrar las propiedades del motor SNMP (engine time, engine boots, engine id, credentials) para todos los hosts. Úselo para forzar un vaciado global de la caché al solucionar problemas de SNMP.	
`housekeeper_execute`	Iniciar el procedimiento de housekeeping. Se ignora si el procedimiento de housekeeping está actualmente en curso.	
`log_level_increase[=<target>]`	Aumentar el nivel de registro; afecta a todos los procesos si no se especifica target. No es compatible con sistemas *BSD*.	**process type** - todos los procesos del tipo especificado (por ejemplo, `poller`). Consulte todos los tipos de procesos del proxy. **process type,N** - tipo de proceso y número (por ejemplo, `poller,3`) **pid** - identificador de proceso (1 a 65535). Para valores mayores, especifique target como 'process type,N'.
`log_level_decrease[=<target>]`	Disminuir el nivel de registro; afecta a todos los procesos si no se especifica target. No es compatible con sistemas *BSD*.	
`prof_enable[=<target>]`	Habilitar el profiling. Afecta a todos los procesos si no se especifica target. El profiling habilitado proporciona detalles de todos los rwlocks/mutexes por nombre de función.	**process type** - todos los procesos del tipo especificado (por ejemplo, `history syncer`). Consulte todos los tipos de procesos del proxy. **process type,N** - tipo de proceso y número (por ejemplo, `history syncer,1`). **pid** - identificador de proceso (1 a 65535). Para valores mayores, especifique target como 'process type,N'. **scope** - `rwlock`, `mutex`, `processing` se pueden usar con el tipo de proceso y número (por ejemplo, `history syncer,1,processing`) o con todos los procesos de un tipo (por ejemplo, `history syncer,rwlock`).
`prof_disable[=<target>]`	Deshabilitar el profiling. Afecta a todos los procesos si no se especifica target.	**process type** - todos los procesos del tipo especificado (por ejemplo, `history syncer`). Consulte todos los tipos de procesos del proxy. **process type,N** - tipo de proceso y número (por ejemplo, `history syncer,1`). **pid** - identificador de proceso (1 a 65535). Para valores mayores, especifique target como 'process type,N'.

Cuando se especifica un tipo de proceso como target para el control en tiempo de ejecución, el comando se aplica solo a los procesos en ejecución de ese tipo. Si actualmente no se está ejecutando ningún proceso del tipo especificado, el comando devuelve un error:

```
zabbix_proxy -R log_level_increase='data sender'
Cannot redirect signal: "data sender" process does not exist
```

Ejemplo de uso del control en tiempo de ejecución para recargar la caché de configuración del proxy:

```
zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf -R config_cache_reload
```

Ejemplo de uso del control en tiempo de ejecución para vaciar la caché de historial de un item:

```
zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf -R history_cache_clear=42243
```

Ejemplos de uso del control en tiempo de ejecución para recopilar información de diagnóstico:

```
### Recopilar toda la información de diagnóstico disponible en el archivo de registro del proxy:
zabbix_proxy -R diaginfo

### Recopilar estadísticas de la caché de historial en el archivo de registro del proxy:
zabbix_proxy -R diaginfo=historycache
```

Ejemplo de uso del control en tiempo de ejecución para recargar la caché SNMP:

```
zabbix_proxy -R snmp_cache_reload
```

> **Attention:**
> Cuando una interfaz SNMPv3 se actualiza mediante la interfaz web de Zabbix, Zabbix recargará automáticamente las nuevas credenciales SNMPv3 para esa interfaz en la mayoría de los casos; use `-R snmp_cache_reload` solo si el sondeo sigue fallando después de cambiar las credenciales (por ejemplo, debido a inconsistencias de engineBoots/engineID o a dispositivos no RFC), o cuando necesite forzar un vaciado global de la caché SNMP para la resolución de problemas.

Ejemplo de uso del control en tiempo de ejecución para activar la ejecución de housekeeper:

```
zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf -R housekeeper_execute
```

Ejemplos de uso del control en tiempo de ejecución para cambiar el nivel de registro:

```
### Aumentar el nivel de registro de todos los procesos:
zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf -R log_level_increase

### Aumentar el nivel de registro del segundo proceso poller:
zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf -R log_level_increase=poller,2

### Aumentar el nivel de registro del proceso con PID 1234:
zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf -R log_level_increase=1234

### Disminuir el nivel de registro de todos los procesos http poller:
zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf -R log_level_decrease="http poller"
```

Procesar usuario

El proxy de Zabbix está diseñado para ejecutarse como un usuario que no sea root. Se ejecutará con el usuario que no sea root con el que se inicie. Por lo tanto, puede ejecutar el proxy con cualquier usuario que no sea root sin ningún problema.

Si intenta ejecutarlo como «root», cambiará al usuario «zabbix» codificado de forma fija, que debe estar presente en su sistema. Solo puede ejecutar el proxy como «root» si modifica adecuadamente el parámetro `AllowRoot` en el archivo de configuración del proxy.

Archivo de configuración

La ruta predeterminada al archivo de configuración depende del sistema operativo y del método de instalación de Zabbix:

- Para la instalación desde el código fuente: `/usr/local/etc/zabbix_proxy.conf`, o según lo establecido por las variables de tiempo de compilación `--sysconfdir` o `--prefix`.
- Para la instalación desde paquetes: `/etc/zabbix/zabbix_proxy.conf`.

Consulte las opciones del archivo de configuración para obtener detalles sobre cómo configurar `zabbix_proxy`.

Tipos de procesos y subprocesos del proxy

- `agent poller`: proceso de sondeo asíncrono para comprobaciones pasivas con un subproceso de trabajo;
- `apm manager`: proceso administrador para la monitorización del rendimiento de las aplicaciones (APM);
- `apm worker`: proceso para gestionar las tareas de monitorización del rendimiento de las aplicaciones (APM) del `apm manager`;
- `availability manager`: proceso para actualizar la disponibilidad de los hosts;
- `browser poller`: proceso de sondeo para comprobaciones de items de navegador;
- `configuration syncer`: proceso para gestionar la caché en memoria de los datos de configuración;
- `data sender`: proceso de envío de datos del proxy;
- `discovery manager`: proceso administrador para el descubrimiento de dispositivos;
- `discovery worker`: proceso para gestionar las tareas de descubrimiento del discovery manager;
- `history syncer`: proceso de escritura en la base de datos del historial;
- `housekeeper`: proceso para eliminar el historial obsoleto de los items;
- `http agent poller`: proceso de sondeo asíncrono para comprobaciones HTTP con un subproceso de trabajo;
- `http poller`: proceso de sondeo de la monitorización web;
- `icmp pinger`: proceso de sondeo para comprobaciones icmpping;
- `internal poller`: proceso de sondeo para comprobaciones internas;
- `ipmi manager`: administrador del proceso de sondeo IPMI;
- `ipmi poller`: proceso de sondeo para comprobaciones IPMI;
- `java poller`: proceso de sondeo para comprobaciones Java;

- odbc poller: proceso de sondeo para comprobaciones ODBC;
- poller: proceso de sondeo normal para comprobaciones pasivas;
- preprocessing manager: administrador de tareas de preprocesamiento con subprocesos de trabajo de preprocesamiento;
- preprocessing worker: subproceso para el preprocesamiento de datos;
- self-monitoring: proceso para recopilar estadísticas internas del server;
- snmp poller: proceso de sondeo asíncrono para comprobaciones SNMP con un subproceso de trabajo (solo items walk[OID] y get[OID]);
- snmp trapper: proceso trapper para traps SNMP;
- supervisor: coordina el inicio y el apagado de los procesos y subprocesos del server, y supervisa su disponibilidad;
- task manager: proceso para la ejecución remota de tareas solicitadas por otros componentes (por ejemplo, cerrar un problema, reconocer un problema, comprobar ahora el valor de un item y la funcionalidad de comandos remotos);
- telemetry query poller: proceso de sondeo para comprobaciones de items de consulta de telemetría;
- trapper: proceso trapper para comprobaciones activas, traps y comunicación con el proxy;
- unreachable poller: proceso de sondeo para dispositivos inaccesibles;
- vmware collector: recopilador de datos de VMware responsable de recopilar datos de los servicios de VMware.

El archivo de registro del proxy se puede utilizar para observar estos tipos de procesos.

El archivo de registro del proxy se crea con permisos de lectura y escritura únicamente para el propietario del archivo. Además, el grupo propietario puede leer el archivo. Se deniegan todos los demás permisos.

Se pueden supervisar varios tipos de procesos del proxy de Zabbix mediante el item interno zabbix[process,<type>,<mode>,<state>].

Estadísticas de transacciones del sincronizador de historial

El título del proceso del sincronizador de historial muestra estadísticas detalladas sobre las transacciones del sincronizador de historial.

```
205276 ?        S      0:00  zabbix_proxy: history syncer #1 [processed 1 values in 0.001179 (0.001167,0.0
205277 ?        S      0:00  zabbix_proxy: history syncer #2 [processed 0 values in 0.000022 (0.000000,0.0
```

Los tiempos, en processed...in N (<timings>) sec, son:

- Tiempo dedicado a escribir los valores de item en la base de datos.
- Tiempo dedicado a actualizar los datos del item (estado, errores).

Procedimiento de housekeeping

El proxy de Zabbix tiene el proceso housekeeper que elimina el historial y las tendencias obsoletos de los item. Se ejecuta en ciclos, con una frecuencia determinada por HousekeepingFrequency, y el límite de eliminación por ciclo determinado por ProxyLocalBuffer y ProxyOfflineBuffer. A diferencia del procedimiento de housekeeping del server de Zabbix, el proceso housekeeper del proxy no usa la tabla housekeeper; elimina todos los datos obsoletos una vez por cada ciclo de housekeeping.

Plataformas compatibles Zabbix proxy se ejecuta en la misma lista de plataformas compatibles que Zabbix server.

Búfer de memoria El búfer de memoria permite almacenar datos nuevos (valores de item, descubrimiento de red, registro automático de hosts) en el búfer y cargarlos en el server de Zabbix sin acceder a la base de datos. El búfer de memoria se introdujo para el proxy en Zabbix 7.0.

En las instalaciones anteriores a Zabbix 7.0, los datos recopilados se almacenaban en la base de datos antes de cargarlos en el server de Zabbix. En estas instalaciones, este sigue siendo el comportamiento predeterminado después de actualizar a Zabbix 7.0.

Para obtener un rendimiento optimizado, se recomienda configurar el uso del búfer de memoria en el proxy. Esto es posible modificando el valor de ProxyBufferMode de disk (valor predeterminado codificado para las instalaciones existentes) a hybrid (recomendado) o memory. También es necesario establecer el tamaño del búfer de memoria (parámetro ProxyMemoryBufferSize).

En el modo híbrido, el búfer está protegido contra la pérdida de datos, ya que los datos no enviados se vacían en la base de datos si el proxy se detiene, el búfer está lleno o los datos son demasiado antiguos. Cuando todos los valores se han vaciado en la base de datos, el proxy vuelve a utilizar el búfer de memoria.

En el modo de memoria, se utilizará el búfer de memoria; sin embargo, no existe protección contra la pérdida de datos. Si el proxy se detiene o la memoria se llena, los datos no enviados se descartarán.

El modo híbrido (ProxyBufferMode=hybrid) se aplica a todas las instalaciones nuevas desde Zabbix 7.0.

Parámetros adicionales, como ProxyMemoryBufferSize y ProxyMemoryBufferAge, definen respectivamente el tamaño del búfer de memoria y la antigüedad máxima de los datos en el búfer.

Tenga en cuenta que, si la configuración contiene conflictos, el proxy mostrará un error y no se iniciará, por ejemplo, si:

- `ProxyBufferMode` está establecido en `hybrid` o `memory` y `ProxyMemoryBufferSize` es 0.
- `ProxyBufferMode` está establecido en `hybrid` o `memory` y `ProxyLocalBuffer` no es 0.

Configuración regional Tenga en cuenta que el proxy requiere una configuración regional UTF-8 para que algunos items textuales puedan interpretarse correctamente. La mayoría de los sistemas modernos tipo Unix tienen una configuración regional UTF-8 como predeterminada; sin embargo, hay algunos sistemas en los que puede ser necesario establecerla específicamente.

Cálculo de colas durante el mantenimiento

> **Attention:**
> El proxy de Zabbix no tiene conocimiento de los períodos de mantenimiento; consulte
> tenance para obtener más detalles.

Uso del frontend de Zabbix detrás de un proxy Si el frontend de Zabbix se ejecuta detrás de un proxy, es necesario reescribir la ruta de las cookies en el archivo de configuración del proxy para que coincida con la ruta a través del proxy inverso. Consulte los ejemplos siguientes. Si no se reescribe la ruta de las cookies, los usuarios pueden experimentar problemas de autorización al intentar iniciar sesión en el frontend de Zabbix.

Ejemplo de configuración para NGINX

```
### ..
location / {
### ..
proxy_cookie_path /zabbix /;
proxy_pass http://192.168.0.94/zabbix/;
### ..
```

Ejemplo de configuración para Apache

```
### ..
ProxyPass "/" http://host/zabbix/
ProxyPassReverse "/" http://host/zabbix/
ProxyPassReverseCookiePath /zabbix /
ProxyPassReverseCookieDomain host zabbix.example.com
### ..
```

Supervisión distribuida mediante proxies

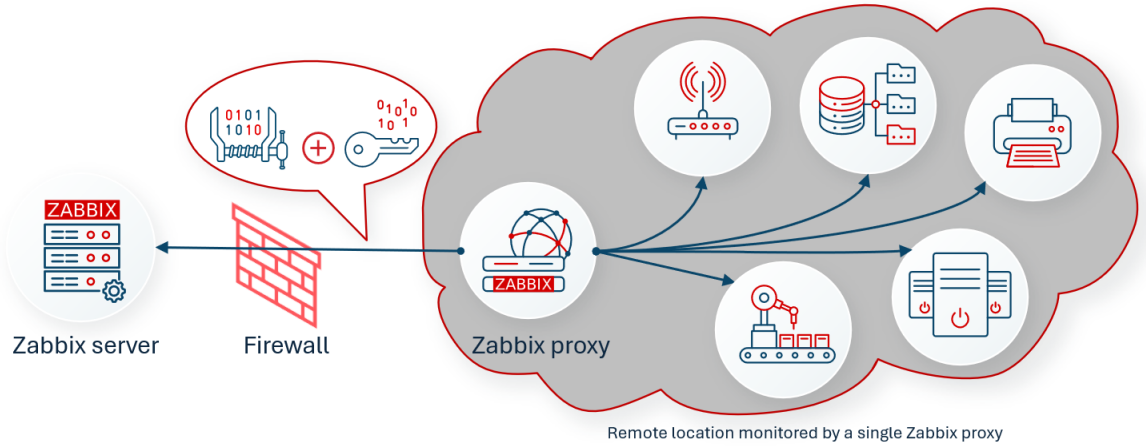
Descripción general

Los proxies de Zabbix se pueden utilizar para crear una configuración de monitorización distribuida eficaz y fiable.

Los proxies se implementan localmente para recopilar datos de rendimiento y disponibilidad. A continuación, los datos se envían a un server centralizado.

Un proxy de Zabbix se puede utilizar para:

- Monitorizar ubicaciones remotas
- Monitorizar ubicaciones con comunicaciones poco fiables
- Descargar de trabajo al server de Zabbix al monitorizar miles de dispositivos
- Recopilar datos de telemetría para la monitorización del rendimiento de aplicaciones (APM)



El proxy solo requiere una conexión TCP con el server de Zabbix. De esta forma, es más fácil atravesar un firewall, ya que solo es necesario configurar una regla de firewall.

Attention:
El proxy de Zabbix debe utilizar una base de datos independiente. Si se configura para utilizar la base de datos del server de Zabbix, la configuración se dañará.

Todos los datos recopilados por el proxy se almacenan localmente antes de transmitirse al server. De esta forma, no se pierden datos debido a problemas temporales de comunicación con el server. Los parámetros `ProxyLocalBuffer` y `ProxyOfflineBuffer` del archivo de configuración del proxy controlan durante cuánto tiempo se conservan los datos localmente.

Attention:
Puede ocurrir que un proxy que recibe directamente de la base de datos del server de Zabbix los cambios de configuración más recientes tenga una configuración más actualizada que el server de Zabbix, cuya configuración puede no actualizarse con la misma rapidez debido al valor de CacheUpdateFrequency. Como resultado, el proxy puede comenzar a recopilar datos y enviarlos al server de Zabbix, que ignora dichos datos.

El proxy de Zabbix es un recopilador de datos. No calcula triggers, procesa eventos ni envía alertas. Para obtener una descripción general de las funciones del proxy, consulte la siguiente tabla:

Función		Compatible con el proxy
Items		
	Comprobaciones de Zabbix agent	**Sí**
	Comprobaciones de Zabbix agent (activas)	**Sí** [1]
	Comprobaciones simples	**Sí**
	Items trapper	**Sí**
	Comprobaciones SNMP	**Sí**
	Traps SNMP	**Sí**
	Comprobaciones IPMI	**Sí**
	Comprobaciones JMX	**Sí**
	Monitorización de archivos de registro	**Sí**
	Comprobaciones internas	**Sí**
	Comprobaciones SSH	**Sí**
	Comprobaciones Telnet	**Sí**
	Comprobaciones externas	**Sí**
	Items dependientes	**Sí**
	Items de script	**Sí**
	Items de navegador	**Sí**
	Items de consulta de telemetría	**Sí**
Monitorización web integrada		**Sí**
Preprocesamiento de valores de item		**Sí**
Descubrimiento de red		**Sí**
Registro automático de agent activo		**Sí**
Descubrimiento de bajo nivel		**Sí** [2]
Comandos remotos		**Sí**
Recopilación de datos de telemetría (APM)		**Sí** [3]
Cálculo de triggers		*No*
Procesamiento de eventos		*No*
Correlación de eventos		*No*
Envío de alertas		*No*

Note:
[1] Para asegurarse de que un agent solicite al proxy (y no al server) las comprobaciones activas, el proxy debe aparecer en el parámetro **ServerActive** del archivo de configuración del agent.
 [2] Para el LLD, el proxy de Zabbix solo recopila y preprocesa los datos y, a continuación, los envía al server de Zabbix para su procesamiento posterior.
 [3] Los datos de telemetría solo se recopilan si APM está habilitado en el proxy (consulte la pestaña *APM* en la configuración del proxy); el server de Zabbix no puede recibir datos de telemetría.

Características del proxy

Al decidir si usar o no un proxy, deben tenerse en cuenta varias consideraciones.

	Proxy
Ligero	**Sí**
GUI	No
Funciona de forma independiente	**Sí**
Mantenimiento sencillo	**Sí**
Creación automática de la base de datos	**Sí**[1]
Administración local	No
Listo para hardware integrado	**Sí**
Conexiones TCP unidireccionales	**Sí**
Configuración centralizada	**Sí**
Genera notificaciones	No

[1] La función de creación automática de la base de datos solo funciona con SQLite. Las demás bases de datos compatibles requieren una configuración manual.

Attention:
El proxy de Zabbix no tiene conocimiento de los períodos de mantenimiento; consulte Cálculo de colas durante el mantenimiento para obtener más detalles.

Protección contra sobrecarga

Si el server de Zabbix estuvo caído durante algún tiempo y los proxies han recopilado muchos datos, cuando el server se inicie puede sobrecargarse (el uso de la caché de historial permanece en 95-100% durante algún tiempo). Esta sobrecarga podría provocar una degradación del rendimiento, en la que las comprobaciones se procesan más lentamente de lo que deberían. Se implementó una protección frente a este escenario para evitar los problemas que surgen debido a la sobrecarga de la caché de historial.

Cuando la caché de historial del server de Zabbix está llena, el acceso de escritura a la caché de historial se limita, lo que detiene los procesos de recopilación de datos del server. El caso más común de sobrecarga de la caché de historial es después de un tiempo de inactividad del server, cuando los proxies están cargando los datos recopilados. Para evitar esto, se añadió la limitación de proxies (actualmente no se puede deshabilitar).

Cuando el uso de la caché de historial alcanza el 80%, el server de Zabbix entra en modo de limitación. En modo de limitación, el server acepta datos de proxy solo cuando el uso de la caché de historial está por debajo del 60%, alternando los proxies aceptados. Una vez que el uso de la caché de historial baja por debajo del 20%, el server vuelve al modo normal.

Además, en modo normal, el server de Zabbix limita los proxies individuales que envían paquetes muy grandes (10,000+ registros) si el uso de la caché de historial supera el 60%. Esta decisión se aplica en el instante en que el server evalúa una carga de proxy y, por lo tanto, no siempre puede reflejarse de inmediato en los gráficos de uso de la caché de historial (el item interno `zabbix[wcache,history,pused]` y su intervalo de actualización pueden omitir picos breves).

Este modo de limitación continuará hasta que el uso de la caché vuelva a alcanzar el 80%, o baje al 20%, o la lista de limitación quede vacía. En el primer caso, el server dejará de aceptar datos de proxy nuevamente. En los otros dos casos, el server comenzará a funcionar normalmente, aceptando datos de todos los proxies.

La información anterior puede ilustrarse en la siguiente tabla:

Uso de la caché de escritura de historial	Modo del server de Zabbix	Acción del server de Zabbix
Alcanza el 80%	Espera	Deja de aceptar datos de proxy, pero mantiene una *lista de limitación* (lista priorizada de proxies que se contactarán más adelante).
Alcanza el 60%	Normal, pero preparado para ser limitado	Puede rechazar cargas de proxy muy grandes (más de 10k registros) al decidir si acepta datos; sigue aceptando otros datos de proxy.
Baja al 20%	Normal	Descarta la lista de limitación y comienza a aceptar datos de proxy normalmente.

Puede usar el item interno zabbix[wcache,history,pused] para correlacionar este comportamiento del server de Zabbix con una métrica.

Configuración

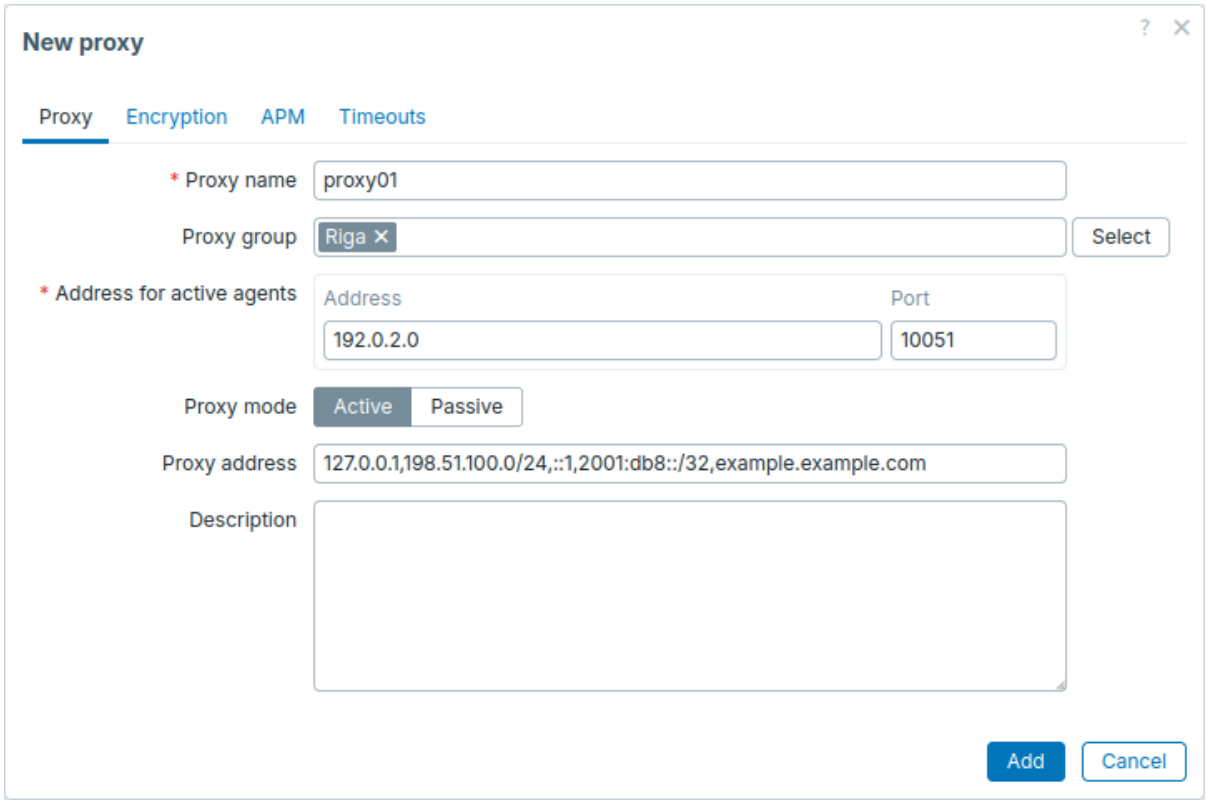
Una vez que haya instalado y configurado un proxy, es momento de configurarlo en el frontend de Zabbix.

Añadir proxies

> **Note:**
> En lugar de instalar un proxy en su propia infraestructura, puede implementar uno en Zabbix Cloud. Para hacerlo, haga clic en *Create proxy in Cloud* en *Administration > Proxies*; Zabbix Cloud se abrirá en una nueva pestaña del navegador. Después de crear allí un nodo proxy, configure el proxy en el frontend de Zabbix como se describe a continuación. Para obtener más información, consulte Configurar un proxy de Zabbix Cloud.

Para configurar un proxy en el frontend de Zabbix:

1. Vaya a *Administration > Proxies*.
2. Haga clic en *Create proxy*.
3. Introduzca los parámetros del proxy en el formulario.



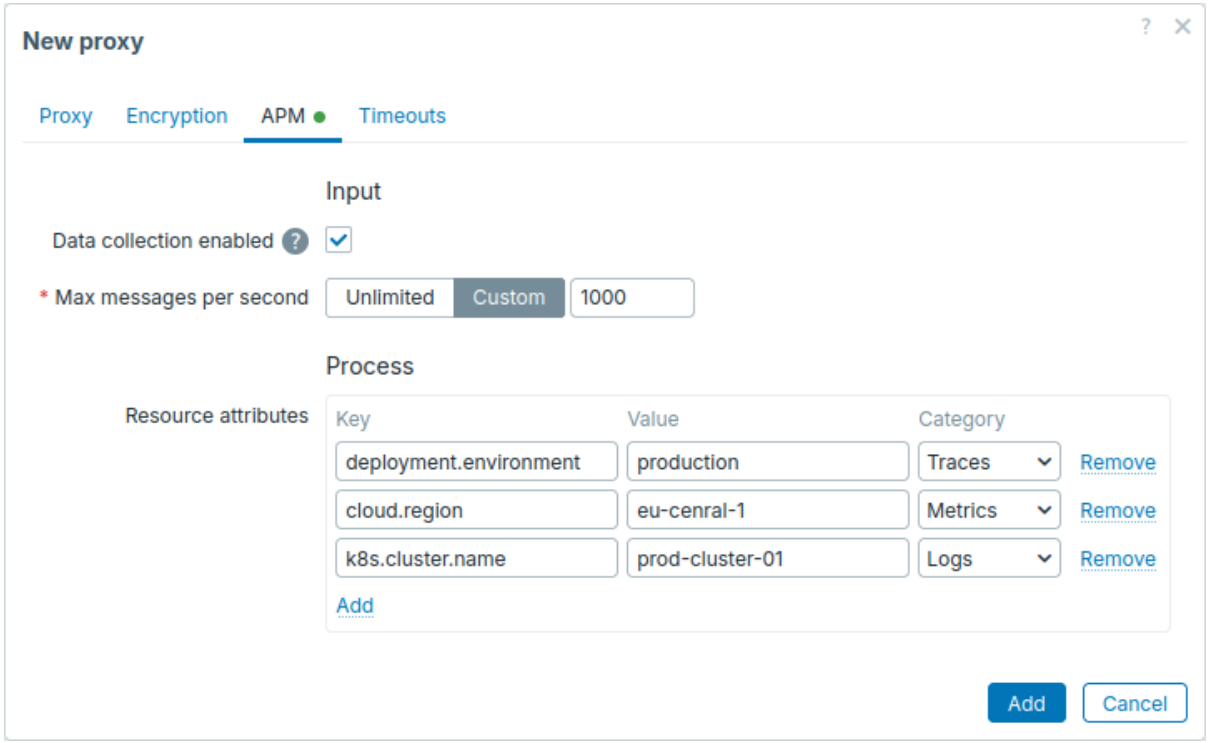
Parameter	Description
Proxy name	Introduzca el nombre del proxy. Debe ser el mismo nombre que aparece en el parámetro *Hostname* del archivo de configuración del proxy.
Proxy group	Seleccione un grupo de proxies para el equilibrio de carga/alta disponibilidad del proxy.
Address for active agents	Introduzca la dirección a la que deben conectarse los agentes activos o senders supervisados. Compatible **solo** con agentes de Zabbix 7.0 o posteriores. Esta dirección se utiliza para conectarse tanto a proxies activos como pasivos. Este campo solo está disponible si se selecciona un grupo de proxies en el campo *Proxy group*.
Address	Dirección IP/nombre DNS al que conectarse.
Port	Número de puerto TCP (10051 de forma predeterminada) al que conectarse. Se admiten macros de usuario.
Proxy mode	Seleccione el modo del proxy. **Active**: el proxy se conectará al Zabbix server y solicitará datos de configuración. **Passive**: el Zabbix server se conecta al proxy. *Tenga en cuenta* que, sin comunicaciones cifradas (confidenciales), los datos de configuración del proxy pueden quedar disponibles para las partes que tengan acceso al puerto trapper del Zabbix server cuando se utiliza un proxy activo. Esto es posible porque cualquiera puede hacerse pasar por un proxy activo y solicitar datos de configuración si no se realiza la autenticación o si las direcciones de los proxies no están limitadas en el campo *Proxy address*.

Parameter	Description
Proxy address	Si se especifica, las solicitudes de proxies activos solo se aceptan desde esta lista de direcciones IP separadas por comas, opcionalmente en notación CIDR, o de nombres DNS del proxy activo de Zabbix. Este campo solo está disponible si se selecciona un proxy activo en el campo *Proxy mode*. No se admiten macros.
Interface	Introduzca los detalles de la interfaz de un proxy pasivo. Este campo solo está disponible si se selecciona un proxy pasivo en el campo *Proxy mode*.
Address	Dirección IP/nombre DNS del proxy pasivo.
Port	Número de puerto TCP del proxy pasivo (10051 de forma predeterminada). Se admiten macros de usuario.
Description	Introduzca la descripción del proxy.

La pestaña **Encryption** permite exigir conexiones cifradas con el proxy.

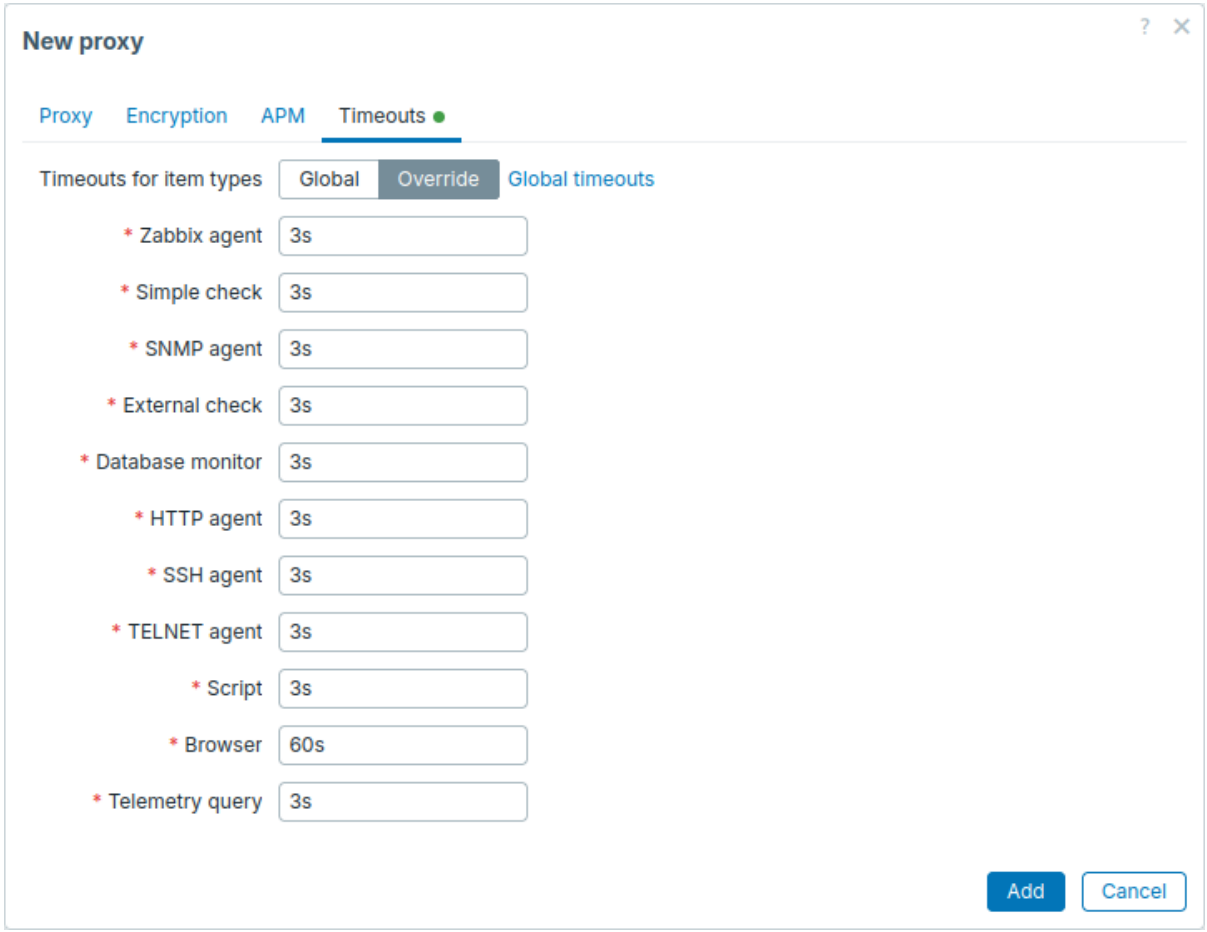
Parameter	Description
Connections to proxy	Cómo se conecta el server al proxy pasivo: sin cifrado (opción predeterminada), mediante PSK (clave precompartida) o certificado.
Connections from proxy	Seleccione qué tipos de conexiones se permiten desde el proxy activo. Se pueden seleccionar varios tipos de conexión al mismo tiempo (útil para realizar pruebas y cambiar a otro tipo de conexión). El valor predeterminado es "No encryption".
Issuer	Emisor permitido del certificado. Primero se valida el certificado con la CA (autoridad certificadora). Si es válido y está firmado por la CA, el campo *Issuer* puede utilizarse para restringir aún más las CA permitidas. Este campo es opcional y está pensado para utilizarse si la instalación de Zabbix emplea certificados de varias CA.
Subject	Asunto permitido del certificado. Primero se valida el certificado con la CA. Si es válido y está firmado por la CA, el campo *Subject* puede utilizarse para permitir un único valor de la cadena *Subject*. Si este campo está vacío, se acepta cualquier certificado válido firmado por la CA configurada.
PSK identity	Cadena de identidad de la clave precompartida. No incluya información confidencial en la identidad PSK, ya que se transmite sin cifrar por la red para informar al receptor de qué PSK debe utilizar.
PSK	Clave precompartida (cadena hexadecimal). Longitud máxima: 512 dígitos hexadecimales (PSK de 256 bytes) si Zabbix utiliza la biblioteca GnuTLS u OpenSSL; 64 dígitos hexadecimales (PSK de 32 bytes) si Zabbix utiliza la biblioteca mbed TLS (PolarSSL). Ejemplo: 1f87b595725ac58dd977beef14b97461a7c1045b9a1c963065002c5473194952

La pestaña **APM** permite configurar la supervisión del rendimiento de aplicaciones (APM) en el proxy. El proxy puede recopilar datos de telemetría (trazas, métricas y registros) enviados por aplicaciones o recopiladores de OpenTelemetry mediante el protocolo OpenTelemetry (OTLP/gRPC), y escribir los datos en el origen de datos de telemetría.



Parameter	Description
Data collection enabled	Marque la casilla para habilitar la recopilación de datos de telemetría en el proxy. Antes de habilitar la recopilación de datos de telemetría, establezca el parámetro `StartAPMCollector` en 1 en el archivo de configuración del proxy (requiere un proxy de Zabbix con compatibilidad con APM).
Max messages per second	Establezca el número máximo de mensajes por segundo que acepta el proxy: **Unlimited**: acepta todos los mensajes entrantes. **Custom**: acepta hasta el número especificado de mensajes por segundo. Un mensaje es una solicitud OTLP enviada por una aplicación, un recopilador de OpenTelemetry u otra fuente de telemetría. Su tamaño depende del emisor; por ejemplo, un recopilador de OpenTelemetry puede agrupar miles de spans en un solo mensaje, mientras que una aplicación sencilla puede enviar mensajes más pequeños (por ejemplo, uno por traza). Si se supera el límite, el proxy responde con el código de estado gRPC `UNAVAILABLE` (error temporal), por lo que el emisor puede volver a intentar enviar el mensaje más tarde, según su configuración.
Add resource attributes	Especifique los atributos de recurso que se añadirán a los datos de telemetría entrantes del tipo seleccionado (por ejemplo, para marcar todos los datos recibidos por este proxy con su entorno o región). Después puede utilizar estos atributos para filtrar los datos de telemetría, por ejemplo, en las condiciones de los items de consulta de telemetría. Si los datos entrantes ya contienen un atributo de recurso con el mismo nombre, se conserva el atributo entrante en lugar del especificado aquí.
Name	Introduzca el nombre del atributo (por ejemplo, `deployment.environment`, `cloud.region`). Debe ser único para el *Type* seleccionado.
Value	Introduzca el valor del atributo (por ejemplo, `production`, `eu-central-1`). Puede estar vacío.
Type	Seleccione el tipo de datos entrantes al que se aplica el atributo: *Traces*, *Metrics* o *Logs*.

La pestaña **Timeouts** permite anular los tiempos de espera globales para los tipos de item que lo admiten.



Parameter	Description
Timeouts for item types	Establezca el tiempo de espera del item (según su tipo): **Global**: utiliza el tiempo de espera global (mostrado en el campo *Timeout* atenuado de cada tipo de item). **Override**: establece un tiempo de espera personalizado (en el campo *Timeout* de cada tipo de item). Rango permitido: 1 - 600s (valor predeterminado: heredado de los tiempos de espera globales). Se admiten sufijos de tiempo, por ejemplo, 30s y 1m, y macros de usuario. Al hacer clic en el enlace *Global timeouts*, puede configurar los tiempos de espera globales. Tenga en cuenta que el enlace *Global timeouts* solo está visible para usuarios de tipo *Super admin* con permisos para la sección del frontend *Administration* > *General*. Tenga en cuenta que, aunque los tiempos de espera del proxy anulan los globales, estos serán anulados por los tiempos de espera de items individuales, si están configurados.

Note:

Si la versión principal del proxy no coincide con la versión principal del server, se mostrará el icono junto a *Timeouts for item types*, con el mensaje al pasar el cursor "Timeouts disabled because the proxy and server versions do not match". En estos casos, el proxy utilizará el parámetro `Timeout` del archivo de configuración del proxy.

El formulario de edición de un proxy existente tiene los siguientes botones adicionales:

- *Refresh configuration*: actualiza la configuración del proxy;
- *Clone*: crea un proxy nuevo basándose en las propiedades del proxy existente;
- *Delete*: elimina el proxy.

Configuración de host

Puede especificar que un host individual debe ser monitoreado por un proxy o grupo de proxy en el formulario de configuración de host, usando el campo *Monitored by*.



La actualización masiva de host es otra forma de especificar que los hosts deben ser monitoreados por un proxy o grupo de proxy.

Balanceo de carga y alta disponibilidad de proxy

Descripción general

Los proxy de Zabbix se pueden organizar en grupos de proxy para habilitar el equilibrio de carga y la alta disponibilidad de los proxy.

El equilibrio de carga y la alta disponibilidad de los proxy consisten en la redistribución automática de los hosts entre los proxy dentro de un grupo de proxy:

- Si un proxy se desconecta, sus hosts se moverán a otros proxy, manteniendo así una alta disponibilidad de los proxy.
- Si un proxy tiene un número de hosts mucho mayor o menor que los demás proxy, sus hosts se moverán a otros proxy para equilibrar la carga de los proxy.

La redistribución de hosts funciona únicamente entre los proxy de un grupo que cumplen las siguientes condiciones:

- Los proxy ejecutan Zabbix 7.0 o una versión posterior.
- La versión del proxy coincide con la versión del server de Zabbix. Si se utiliza Zabbix agent (pasivo), la versión del proxy debe coincidir con la versión del agent. Los agent activos solo requieren Zabbix 7.0 o una versión posterior.
- El grupo de proxy tiene un estado en línea.
- Los hosts están configurados para ser supervisados por un grupo de proxy en lugar de por proxy individuales.

Note:
El estado del grupo de proxy se puede supervisar mediante comprobaciones internas desde cualquier host asignado a un grupo de proxy. Sin embargo, para supervisar el estado de un único proxy de un grupo, asigne el host a ese proxy; de lo contrario, los resultados pueden ser incoherentes.

Redistribución de hosts

El balanceo de carga y la alta disponibilidad de los proxy es gestionado por Zabbix server a través del administrador de grupos de proxy, que supervisa continuamente el estado de todos los proxy en cada grupo de proxy y su distribución de hosts.

La alta disponibilidad de los proxy dentro de un grupo se garantiza mediante el failover de proxy: cuando un proxy se desconecta, sus hosts se redistribuyen inmediatamente a otros proxy. También se produce balanceo de carga de proxy, ya que los hosts se reasignan a los proxy con la menor cantidad de hosts asignados.

Además, el balanceo de carga de proxy se activa cuando el número de hosts de un proxy difiere del promedio del grupo en al menos 10 hosts y en un factor de 2 (exceso de hosts o déficit de hosts). Si el desequilibrio persiste después de un período de gracia (10 x failover delay), el grupo de proxy se pone en cola para la redistribución de hosts.

El administrador de grupos de proxy redistribuye los hosts usando la siguiente lógica:

1. Calcular el número promedio de hosts por proxy.
2. Para los proxy con exceso de hosts, mover los hosts excedentes al grupo de proxy no asignados.
3. Para los proxy con déficit de hosts, calcular cuántos hosts se necesitan para alcanzar el equilibrio.
4. Eliminar el número requerido de hosts de los proxy con más hosts.
5. Mover los hosts no asignados a los proxy con menos hosts.

Ejemplos de redistribución de hosts:

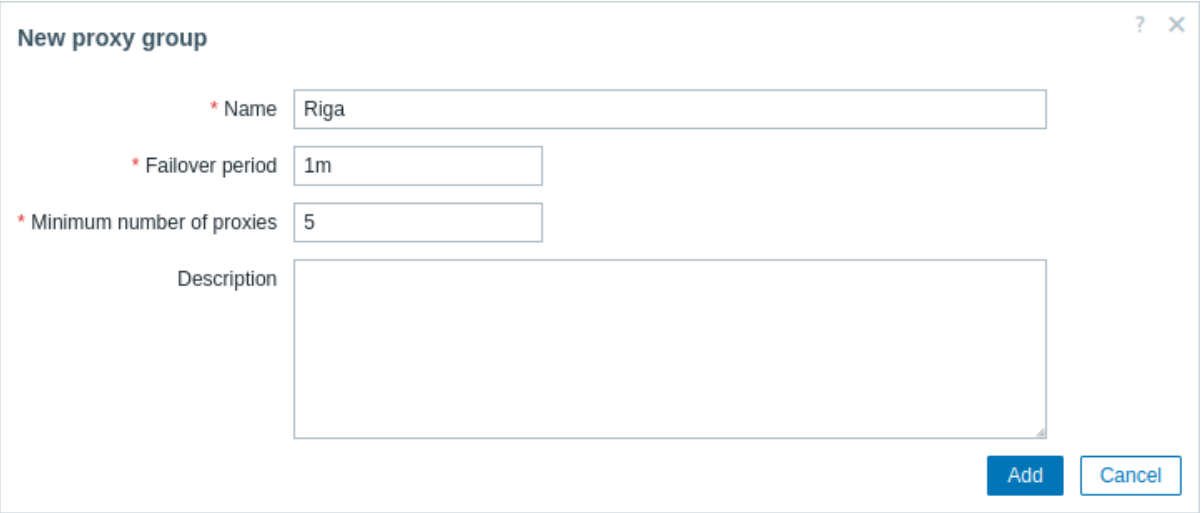
Hosts on proxy	Group average	Host reassignment
100	50	Yes
60	50	No
40	50	No
25	50	Yes
15	5	Yes
10	5	No

Attention:
Tener menos de 10 hosts monitored by un grupo de proxy puede provocar una distribución desigual de hosts entre los proxy del grupo.

Configuración de un grupo de proxies

Para configurar un grupo de proxies en el frontend de Zabbix:

1. Vaya a *Administración > Grupos de proxies*.
2. Haga clic en *Crear grupo de proxies*.
3. Introduzca los parámetros del grupo de proxies en el formulario.



Parámetro	Descripción
Nombre	Nombre del grupo de proxies.
Período de conmutación por error	Período, en segundos, durante el cual un proxy del grupo de proxies debe comunicarse con el server de Zabbix para considerarse en línea (valor predeterminado: 1m; rango: 10s-15m). Si el proxy no se comunica durante este período, su estado cambia a *Fuera de línea* y sus hosts se redistribuyen inmediatamente entre otros proxies. El equilibrio de carga de los proxies comienza después de 10 veces este período. Admite sufijos de tiempo (por ejemplo, 30s, 1m) y macros de usuario.
Número mínimo de proxies	Número mínimo de proxies en línea necesarios para mantener el grupo de proxies en línea (valor predeterminado: 1; rango: 1-1000). Admite macros de usuario. Este valor debe ser inferior al número total de proxies del grupo. Por ejemplo, en un grupo de 10 proxies, establecer el mínimo en 10 hará que el grupo pase a estar fuera de línea si algún proxy falla. Tenga en cuenta que los proxies en línea de un grupo fuera de línea continúan funcionando normalmente, pero no se realizará el equilibrio de carga ni la alta disponibilidad.
Descripción	Descripción del grupo de proxies.
Proxies	Muestra una lista de hasta cinco proxies (como enlaces o en texto sin formato, según los permisos del usuario sobre los proxies) al editar un grupo que contiene proxies.

Configuración del balanceo de carga de los proxy

Para utilizar el balanceo de carga de los proxy, debe configurar un grupo de proxy en el frontend de Zabbix (consulte arriba) y asegurarse de que los hosts estén supervisados por un grupo de proxy, no por proxy individuales (puede utilizar la actualización masiva de hosts para mover los hosts de los proxy al grupo de proxy).

Si utiliza Zabbix agent, configúrelo también de la siguiente manera:

- Para las comprobaciones pasivas, indique todos los proxy del grupo de proxy en el parámetro Server.
- Para las comprobaciones activas, se recomienda indicar todos los proxy del grupo de proxy o el server de Zabbix en el parámetro ServerActive. Tenga en cuenta que solo Zabbix agent 7.0 (o posterior) funcionará con grupos de proxy en su modo activo.

Si el parámetro `ServerActive` incluye solo un proxy de un grupo de proxy (o el server de Zabbix), el agent podrá conectarse igualmente al proxy correcto. Cuando el servicio del agent se inicia y se conecta al proxy especificado, el agent recibirá y almacenará en caché la lista completa de las direcciones IP de los proxy y su carga actual dentro del grupo. A continuación, las

comprobaciones activas se redirigirán al proxy en línea correcto para el host, según la asignación actual de proxy-host dentro del grupo de proxy.

Warning:
Especificar un único proxy en el parámetro `ServerActive` de Zabbix agent puede provocar la pérdida de datos de supervisión si el agent se inicia o reinicia mientras el proxy especificado está fuera de línea.

Note:
Al utilizar Zabbix sender, las solicitudes de datos también se redirigen al proxy en línea correcto para el host, según la asignación actual de proxy-host dentro del grupo de proxy. Sin embargo, si envía valores de varios hosts desde un archivo de entrada, utilice la opción `-g` para evitar enviar datos al proxy incorrecto.

Zabbix agent también debe poder conectarse a todos los proxy del grupo de proxy a través del firewall. De lo contrario, las comprobaciones activas pueden quedar bloqueadas o fallar durante la redirección o la conmutación por error. Por ejemplo:

- Durante las comprobaciones activas, un proxy puede redirigir el agent a otro proxy. Si ese proxy está bloqueado por un firewall, la comunicación quedará bloqueada mientras espera una respuesta.
- En configuraciones estables de alta disponibilidad sin un rebalanceo reciente, es posible que los agent nunca contacten con los proxy de respaldo. Si las reglas del firewall cambiaron y no se probaron, la conmutación por error puede fallar.

Prueba del balanceo de carga del proxy

Para probar el balanceo de carga del proxy:

1. Configure un grupo de proxies.
2. Asegúrese de que el grupo de proxies tenga un estado en línea.
3. Asegúrese de que los hosts estén supervisados por un grupo de proxies, no por proxies individuales (puede utilizar la actualización masiva de hosts para mover los hosts de los proxies al grupo de proxies).
4. Espere unos segundos a que se actualicen la configuración y la distribución de hosts entre los proxies del grupo de proxies. Observe el cambio actualizando la lista de hosts en *Administración* > *Proxies*.

Notas importantes

- Los traps SNMP no son compatibles con los proxy de un grupo de proxy.
- Las comprobaciones que dependen de una configuración externa (por ejemplo, los scripts para las comprobaciones externas o la configuración de ODBC para las comprobaciones de bases de datos) deben tener la misma configuración en todos los proxy del grupo de proxy.
- Las comprobaciones de bases de datos requieren permisos ampliados en el objeto o server de la base de datos.
- Los hosts de VMware supervisados por un grupo de proxy se distribuirán aleatoriamente entre los proxy del grupo. Esto hace que cada proxy almacene en caché todos los datos de VMware, lo que genera una carga adicional en vCenter.
- Los hosts creados a partir de datos de autorregistro de un proxy en un grupo de proxy se configuran para ser supervisados por ese grupo de proxy. Sin embargo, los hosts creados a partir de datos de descubrimiento de red de un proxy en un grupo de proxy se configuran para ser supervisados por ese proxy.

Sincronización de la configuración de monitorización

Descripción general

Esta página proporciona detalles sobre la actualización de la configuración de monitoreo para el proxy, es decir, cómo los cambios realizados en la configuración de monitoreo en el server se sincronizan con el proxy.

Actualización incremental

La actualización de la configuración del proxy es incremental.
Durante una sincronización de la configuración, solo se actualizan las entidades modificadas (por lo tanto, si no se ha modificado ninguna entidad, no se enviará nada).
Este enfoque permite ahorrar recursos y establecer un intervalo más corto (casi instantáneo) para la actualización de la configuración del proxy.

Los cambios en la configuración del proxy se registran mediante números de revisión. Solo se incluyen en los datos de configuración enviados al proxy las entidades cuyas revisiones sean superiores a la revisión de configuración del proxy.

Las entidades incluidas en una sincronización de la configuración son las siguientes:

Entity	Details
autoregistration tls data	Todos los datos TLS de autoregistro.

Entity	Details
expressions	Todas las expresiones (expresiones regulares, tablas de expresiones).
global configuration	Configuración global definida en la tabla 'config'.
host	Todas las propiedades, interfaces, inventario, items, preprocesamiento de items, parámetros de items y escenarios web de un host.
host macros	Todas las macros definidas en un host y todos los ID de template vinculados a él.
proxy discovery rule	Reglas de descubrimiento y comprobaciones asignadas a un proxy.

Esto significa lo siguiente:

- Si se modifica un item en un **host**, se sincronizará toda la configuración de ese host.
- Si se modifica una **expresión regular**, se sincronizarán todas las expresiones regulares.

Una excepción son las macros de host, que también se envían si se ha modificado cualquier elemento del host.

El comando `-R config_cache_reload` en el proxy también iniciará una actualización incremental.

Tenga en cuenta que se realizará una sincronización completa de la configuración al iniciar o reiniciar un proxy, durante una conmutación por error de HA, si el token de sesión ha cambiado o si la actualización de la configuración ha fallado en el proxy, por ejemplo, si se interrumpió la conexión mientras se recibían los datos de configuración.

Parámetros de configuración

El parámetro **ProxyConfigFrequency** determina con qué frecuencia se sincroniza la configuración del proxy con el server (10 segundos de forma predeterminada).

Tenga en cuenta que ProxyConfigFrequency es:

- parámetro del server para proxies pasivos
- parámetro del proxy para proxies activos

En los proxies activos, ProxyConfigFrequency es un parámetro nuevo desde Zabbix 6.4 y debe usarse en lugar del ahora obsoleto ConfigFrequency.

> **Attention:**
> Si se usan tanto ProxyConfigFrequency como ConfigFrequency, el proxy registrará un error y se cerrará.

Cálculo de colas durante el mantenimiento

> **Attention:**
> El proxy de Zabbix no tiene conocimiento de los períodos de mantenimiento; consulte Cálculo de colas durante el mantenimiento para obtener más detalles.

Protocolo de intercambio de datos entre server y proxy

Server - el intercambio de datos con proxy se basa en el formato JSON.

Los mensajes de solicitud y respuesta deben comenzar con header and data length.

proxy pasivo

Solicitud de configuración

El server enviará primero una solicitud `proxy config` vacía. Esta solicitud se envía cada `ProxyConfigFrequency` segundos (parámetro de configuración del server).

El proxy responde con la versión actual del proxy, el token de sesión y la revisión de la configuración. El server responde con los datos de configuración que deben actualizarse.

name	value type	description
server→proxy:		
request	*string*	'proxy config'
proxy→server:		
version	*string*	Versión del proxy (<major>.<minor>.<build>).
session	*string*	Token de sesión de la configuración del proxy.

name	value type	description
config_revision	*number*	Revisión de la configuración del proxy.
server→proxy:		
full_sync	*number*	1 - si se envían todos los datos de configuración; ausente en caso contrario (opcional).
data	*array*	Objeto con los datos de las tablas. Ausente si la configuración no ha cambiado (opcional).
<table>	*object*	Uno o más objetos con datos de <table> (opcional, según los cambios).
fields	*array*	Array de nombres de campos.
-	*string*	Nombre del campo.
data	*array*	Array de filas.
-	*array*	Array de columnas.
-	*string,number*	Valor de la columna con un tipo que depende del tipo de columna en el esquema de la base de datos.
macro.secrets	*object*	Información sobre macros secretas, ausente si no hay cambios en las macros de vault (opcional).
config_revision	*number*	Revisión de la caché de configuración; se envía con los datos de configuración (opcional).
del_hostids	*array*	Array de hostids eliminados (opcional).
-	*number*	Identificador del host.
del_macro_hostids	*array*	Array de hostids a los que se han eliminado todas las macros (opcional).
-	*number*	Identificador del host.
proxy→server:		
response	*string*	Información sobre el resultado correcto de la solicitud ('success' o 'failed').
version	*string*	Versión del proxy (<major>.<minor>.<build>).

Ejemplo:

server→proxy:

```
{
  "request":"proxy config"
}
```

proxy→server:

```
{
  "version": "8.0.0",
  "session": "0033124949800811e5686dbfd9bcea98",
  "config_revision": 0
}
```

server→proxy:

```
{
    "full_sync": 1,
    "data": {
        "hosts": {
            "fields": ["hostid", "host", "status", "ipmi_authtype", "ipmi_privilege", "ipmi_username", "ip
            "data": [
                [10084, "Zabbix server", 0, -1, 2, "", "", "Zabbix server", 1, 1, "", "", "", ""]
            ]
        },
        "interface": {
            "fields": ["interfaceid", "hostid", "main", "type", "useip", "ip", "dns", "port", "available"]
            "data": [
                [1, 10084, 1, 1, 1, "127.0.0.1", "", "10053", 1]
            ]
```

```
    },
    "interface_snmp": {
        "fields": ["interfaceid", "version", "bulk", "community", "securityname", "securitylevel", "au
        "data": []
    },
    "host_inventory": {
        "fields": ["hostid", "type", "type_full", "name", "alias", "os", "os_full", "os_short", "seria
        "data": [
            [10084, "", "", ..., "56.95387", "24.22067", ..., ""]
        ]
    },
    "items": {
        "fields": ["itemid", "type", "snmp_oid", "hostid", "key_", "delay", "history", "status", "valu
        "data": [
            [44161, 7, "", 10084, "agent.hostmetadata", "10s", "90d", 0, 1, "", "", "", "", 0, "", "",
            [44162, 0, "", 10084, "agent.ping", "10s", "90d", 0, 3, "", "", "", "", 0, "", "", "", "",
        ]
    },
    "item_rtdata": {
        "fields": ["itemid", "lastlogsize", "mtime"],
        "data": [
            [44161, 0, 0],
            [44162, 0, 0]
        ]
    },
    "item_preproc": {
        "fields": ["item_preprocid", "itemid", "step", "type", "params", "error_handler", "error_handl
        "data": []
    },
    "item_parameter": {
        "fields": ["item_parameterid", "itemid", "name", "value"],
        "data": []
    },
    "globalmacro": {
        "fields": ["globalmacroid", "macro", "value", "type"],
        "data": [
            [2, "{$SNMP_COMMUNITY}", "public", 0]
        ]
    },
    "hosts_templates": {
        "fields": ["hosttemplateid", "hostid", "templateid", "link_type"],
        "data": []
    },
    "hostmacro": {
        "fields": ["hostmacroid", "hostid", "macro", "value", "type", "automatic"],
        "data": [
            [5676, 10084, "{$M}", "AppID=zabbix_server&Query=Safe=passwordSafe;Object=zabbix:Content",
        ]
    },
    "drules": {
        "fields": ["druleid", "name", "iprange", "delay"],
        "data": [
            [2, "Red local", "127.0.0.1", "10s"]
        ]
    },
    "dchecks": {
        "fields": ["dcheckid", "druleid", "type", "key_", "snmp_community", "ports", "snmpv3_securityn
        "data": [
            [2, 2, 9, "system.uname", "", "10052", "", 0, "", "", 0, 0, 0, "", 1, 0]
        ]
    },
    "regexps": {
```

```
            "fields": ["regexpid", "name"],
            "data": [
                [1, "Sistemas de archivos para descubrimiento"],
                [2, "Interfaces de red para descubrimiento"],
                [3, "Dispositivos de almacenamiento para descubrimiento SNMP"],
                [4, "Nombres de servicios de Windows para descubrimiento"],
                [5, "Estados de inicio de servicios de Windows para descubrimiento"]
            ]
        },
        "expressions": {
            "fields": ["expressionid", "regexpid", "expression", "expression_type", "exp_delimiter", "case
            "data": [
                [1, 1, "^(btrfs|ext2|ext3|ext4|reiser|xfs|ffs|ufs|jfs|jfs2|vxfs|hfs|apfs|refs|ntfs|fat32|z
                [3, 3, "^(Physical memory|Virtual memory|Memory buffers|Cached memory|Swap space)$", 4, ",
                [5, 4, "^(MMCSS|gupdate|SysmonLog|clr_optimization_v2.0.50727_32|clr_optimization_v4.0.303
                [6, 5, "^(automatic|automatic delayed)$", 3, ",", 1],
                [7, 2, "^Software Loopback Interface", 4, ",", 1],
                [8, 2, "^(In)?[Ll]oop[Bb]ack[0-9._]*$", 4, ",", 1],
                [9, 2, "^NULL[0-9.]*$", 4, ",", 1],
                [10, 2, "^[Ll]o[0-9.]*$", 4, ",", 1],
                [11, 2, "^[Ss]ystem$", 4, ",", 1],
                [12, 2, "^Nu[0-9.]*$", 4, ",", 1]
            ]
        },
        "settings": {
            "fields": ["name", "type", "value_str", "value_int"],
            "data": [
                ["autoreg_tls_accept", 2, "", 1],
                ["hk_history_global", 2, "", 0],
                ["snmptrap_logging", 2, "", 1],
                ["proxy_secrets_provider", 2, "", 0],
                ["hk_history", 1, "31d", 0],
                ["timeout_db_monitor", 1, "3s", 0],
                ["timeout_external_check", 1, "3s", 0],
                ["timeout_http_agent", 1, "3s", 0],
                ["timeout_simple_check", 1, "3s", 0],
                ["timeout_snmp_agent", 1, "3s", 0],
                ["timeout_ssh_agent", 1, "3s", 0],
                ["timeout_telnet_agent", 1, "3s", 0],
                ["timeout_zabbix_agent", 1, "3s", 0],
                ["timeout_browser", 1, "30s", 0]
            ]
        },
        "httptest": {
            "fields": ["httptestid", "name", "delay", "agent", "authentication", "http_user", "http_passwo
            "data": []
        },
        "httptestitem": {
            "fields": ["httptestitemid", "httptestid", "itemid", "type"],
            "data": []
        },
        "httptest_field": {
            "fields": ["httptest_fieldid", "httptestid", "type", "name", "value"],
            "data": []
        },
        "httpstep": {
            "fields": ["httpstepid", "httptestid", "name", "no", "url", "timeout", "posts", "required", "s
            "data": []
        },
        "httpstepitem": {
            "fields": ["httpstepitemid", "httpstepid", "itemid", "type"],
            "data": []
```

```
        },
        "httpstep_field": {
            "fields": ["httpstep_fieldid", "httpstepid", "type", "name", "value"],
            "data": []
        },
        "config_autoreg_tls": {
            "fields": ["autoreg_tlsid", "tls_psk_identity", "tls_psk"],
            "data": [
                [1, "", ""]
            ]
        }
    },
    "macro.secrets": {
        "AppID=zabbix_server&Query=Safe=passwordSafe;Object=zabbix": {
            "Content": "738"
        }
    },
    "config_revision": 2
}
```

proxy→server:

```
{
  "response": "success",
  "version": "8.0.0"
}
```

Solicitud de datos

La solicitud `proxy data` se utiliza para obtener la disponibilidad de las interfaces del host, así como datos históricos, de descubrimiento y de autorregistro desde el proxy. Esta solicitud se envía cada `ProxyDataFrequency` (parámetro de configuración del server) segundos.

name	value type	description
server→proxy:		
request	*string*	'proxy data'
proxy→server:		
session	*string*	Token de sesión de datos.
interface availability	*array*	*(opcional)* Matriz de objetos de disponibilidad de interfaces.
interfaceid	*number*	Identificador de la interfaz.
available	*number*	Disponibilidad de la interfaz: **0**, *INTERFACE_AVAILABLE_UNKNOWN* - desconocida **1**, *INTERFACE_AVAILABLE_TRUE* - disponible **2**, *INTERFACE_AVAILABLE_FALSE* - no disponible
error	*string*	Mensaje de error de la interfaz o cadena vacía.
history data	*array*	*(opcional)* Matriz de objetos de datos históricos.
itemid	*number*	Identificador del item.
clock	*number*	Marca de tiempo del valor del item (segundos).
ns	*number*	Marca de tiempo del valor del item (nanosegundos).
value	*string*	*(opcional)* Valor del item.
id	*number*	Identificador del valor (contador ascendente, único dentro de una sesión de datos).
timestamp	*number*	*(opcional)* Marca de tiempo de los items de tipo log.
source	*string*	*(opcional)* Valor de origen del item eventlog.
severity	*number*	*(opcional)* Valor de severidad del item eventlog.
eventid	*number*	*(opcional)* Valor eventid del item eventlog.
state	*string*	*(opcional)* Estado del item: **0**, *ITEM_STATE_NORMAL* **1**, *ITEM_STATE_NOTSUPPORTED*

name	value type	description
lastlogsize	*number*	*(opcional)* Último tamaño de log de los items de tipo log.
mtime	*number*	*(opcional)* Hora de modificación de los items de tipo log.
discovery data	*array*	*(opcional)* Matriz de objetos de datos de descubrimiento.
clock	*number*	Marca de tiempo de los datos de descubrimiento.
druleid	*number*	Identificador de la regla de descubrimiento.
dcheckid	*number*	Identificador de la comprobación de descubrimiento o null para datos de la regla de descubrimiento.
type	*number*	Tipo de comprobación de descubrimiento: **-1** datos de la regla de descubrimiento **0**, *SVC_SSH* - comprobación del servicio SSH **1**, *SVC_LDAP* - comprobación del servicio LDAP **2**, *SVC_SMTP* - comprobación del servicio SMTP **3**, *SVC_FTP* - comprobación del servicio FTP **4**, *SVC_HTTP* - comprobación del servicio HTTP **5**, *SVC_POP* - comprobación del servicio POP **6**, *SVC_NNTP* - comprobación del servicio NNTP **7**, *SVC_IMAP* - comprobación del servicio IMAP **8**, *SVC_TCP* - comprobación de disponibilidad del puerto TCP **9**, *SVC_AGENT* - agent de Zabbix **10**, *SVC_SNMPv1* - agent SNMPv1 **11**, *SVC_SNMPv2* - agent SNMPv2 **12**, *SVC_ICMPPING* - ping ICMP **13**, *SVC_SNMPv3* - agent SNMPv3 **14**, *SVC_HTTPS* - comprobación del servicio HTTPS **15**, *SVC_TELNET* - comprobación de disponibilidad de Telnet
ip	*string*	Dirección IP del host.
dns	*string*	Nombre DNS del host.
port	*number*	*(opcional)* Número de puerto del servicio.
key_	*string*	*(opcional)* Clave del item para la comprobación de descubrimiento de tipo **9** *SVC_AGENT*
value	*string*	*(opcional)* Valor recibido del servicio; puede estar vacío para la mayoría de los servicios.
status	*number*	*(opcional)* Estado del servicio: **0**, *DOBJECT_STATUS_UP* - servicio UP **1**, *DOBJECT_STATUS_DOWN* - servicio DOWN
auto registration	*array*	*(opcional)* Matriz de objetos de datos de autorregistro.
clock	*number*	Marca de tiempo de los datos de autorregistro.
host	*string*	Nombre del host.
ip	*string*	*(opcional)* Dirección IP del host.
dns	*string*	*(opcional)* Nombre DNS resuelto a partir de la dirección IP.
port	*string*	*(opcional)* Puerto del host.
host_metadata	*string*	*(opcional)* Metadatos del host enviados por agent (según el parámetro de configuración de agent HostMetadata o HostMetadataItem).
tasks	*array*	*(opcional)* Matriz de tareas.
type	*number*	Tipo de tarea: **0**, *ZBX_TM_TASK_PROCESS_REMOTE_COMMAND_RESULT* - resultado de comando remoto
status	*number*	Estado de ejecución del comando remoto: **0**, *ZBX_TM_REMOTE_COMMAND_COMPLETED* - comando remoto completado correctamente **1**, *ZBX_TM_REMOTE_COMMAND_FAILED* - error en el comando remoto
error	*string*	*(opcional)* Mensaje de error.

name	value type	description
parent_taskid	*number*	ID de la tarea principal.
more	*number*	*(opcional)* 1 - hay más datos históricos por enviar.
clock	*number*	*(opcional)* Marca de tiempo de la transferencia de datos (segundos).
ns	*number*	*(opcional)* Marca de tiempo de la transferencia de datos (nanosegundos).
version	*string*	Versión del proxy (<major>.<minor>.<build>).
server→proxy:		
response	*string*	Información de éxito de la solicitud ('success' o 'failed').
tasks	*array*	*(opcional)* Matriz de tareas.
type	*number*	Tipo de tarea: **1**, *ZBX_TM_TASK_PROCESS_REMOTE_COMMAND* - comando remoto
clock	*number*	Hora de creación de la tarea.
ttl	*number*	Tiempo en segundos tras el cual la tarea expira.
commandtype	*number*	Tipo de comando remoto: **0**, *ZBX_SCRIPT_TYPE_CUSTOM_SCRIPT* - usar script personalizado **1**, *ZBX_SCRIPT_TYPE_IPMI* - usar IPMI **2**, *ZBX_SCRIPT_TYPE_SSH* - usar SSH **3**, *ZBX_SCRIPT_TYPE_TELNET* - usar Telnet **4**, *ZBX_SCRIPT_TYPE_GLOBAL_SCRIPT* - usar script global (actualmente equivalente funcionalmente a un script personalizado)
command	*string*	Comando remoto a ejecutar.
execute_on	*number*	Destino de ejecución para scripts personalizados: **0**, *ZBX_SCRIPT_EXECUTE_ON_AGENT* - ejecutar el script en el agent **1**, *ZBX_SCRIPT_EXECUTE_ON_SERVER* - ejecutar el script en el server **2**, *ZBX_SCRIPT_EXECUTE_ON_PROXY* - ejecutar el script en el proxy
port	*number*	*(opcional)* Puerto para comandos Telnet y SSH.
authtype	*number*	*(opcional)* Tipo de autenticación para comandos SSH.
username	*string*	*(opcional)* Nombre de usuario para comandos Telnet y SSH.
password	*string*	*(opcional)* Contraseña para comandos Telnet y SSH.
publickey	*string*	*(opcional)* Clave pública para comandos SSH.
privatekey	*string*	*(opcional)* Clave privada para comandos SSH.
parent_taskid	*number*	ID de la tarea principal.
hostid	*number*	ID del host de destino.

Ejemplo:

server→proxy:

```
{
  "request": "proxy data"
}
```

proxy→server:

```
{
    "session": "12345678901234567890123456789012"
    "interface availability": [
        {
            "interfaceid": 1,
            "available": 1,
            "error": ""
    },
        {
            "interfaceid": 2,
            "available": 2,
            "error": "Get value from agent failed: cannot connect to [[127.0.0.1]:10049]: [111] Connection
    },
        {
            "interfaceid": 3,
            "available": 1,
```

```
            "error": ""
    },
        {
            "interfaceid": 4,
            "available": 1,
            "error": ""
    }
    ],
    "history data":[
        {
            "itemid":"12345",
            "clock":1478609647,
            "ns":332510044,
            "value":"52956612",
            "id": 1
        },
        {
            "itemid":"12346",
            "clock":1478609647,
            "ns":330690279,
            "state":1,
            "value":"Cannot find information for this network interface in /proc/net/dev.",
            "id": 2
        }
    ],
    "discovery data":[
        {
            "clock":1478608764,
            "drule":2,
            "dcheck":3,
            "type":12,
            "ip":"10.3.0.10",
            "dns":"vdebian",
            "status":1
        },
        {
            "clock":1478608764,
            "drule":2,
            "dcheck":null,
            "type":-1,
            "ip":"10.3.0.10",
            "dns":"vdebian",
            "status":1
        }
    ],
    "auto registration":[
        {
            "clock":1478608371,
            "host":"Logger1",
            "ip":"10.3.0.1",
            "dns":"localhost",
            "port":"10050"
        },
        {
            "clock":1478608381,
            "host":"Logger2",
            "ip":"10.3.0.2",
            "dns":"localhost",
            "port":"10050"
        }
    ],
    "tasks":[
```

```
        {
            "type": 0,
            "status": 0,
            "parent_taskid": 10
        },
        {
            "type": 0,
            "status": 1,
            "error": "No permissions to execute task.",
            "parent_taskid": 20
        }
    ],
    "version":"8.0.0"
}
```

server→proxy:

```
{
  "response": "success",
  "tasks":[
      {
         "type": 1,
         "clock": 1478608371,
         "ttl": 600,
         "commandtype": 2,
         "command": "restart_service1.sh",
         "execute_on": 2,
         "port": 80,
         "authtype": 0,
         "username": "userA",
         "password": "password1",
         "publickey": "MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQCqGKukO1De7zhZj6+H0qtjTkVxwTCpvKe",
         "privatekey": "lsuusFncCzWBQ7RKNUSesmQRMSGkVb1/3j+skZ6UtW+5u09lHNsj6tQ5QCqGKukO1De7zhd",
         "parent_taskid": 10,
         "hostid": 10070
      },
      {
         "type": 1,
         "clock": 1478608381,
         "ttl": 600,
         "commandtype": 1,
         "command": "restart_service2.sh",
         "execute_on": 0,
         "authtype": 0,
         "username": "",
         "password": "",
         "publickey": "",
         "privatekey": "",
         "parent_taskid": 20,
         "hostid": 10084
      }
  ]
}
```

Solicitud de tareas

El intercambio de `proxy tasks` gestiona la entrega y la confirmación de tareas de comandos remotos entre el server y un proxy pasivo. Cuando el server consulta al proxy para obtener los resultados de las tareas, envía una solicitud vacía de `proxy tasks` en el intervalo definido por `ProxyDataFrequency`. El proxy responde con su versión actual y con cualquier resultado de tarea pendiente (incluidas las marcas de tiempo). A continuación, el server confirma la recepción devolviendo una `response` y puede incluir nuevas tareas para que el proxy las ejecute.

name	value type	description
server→proxy:		
request	*string*	'proxy tasks'
proxy→server:		
version	*string*	Versión del proxy (<major>.<minor>.<build>).
clock	*number*	*(opcional)* Marca de tiempo de la transferencia de datos (segundos).
ns	*number*	*(opcional)* Marca de tiempo de la transferencia de datos (nanosegundos).
server→proxy:		
response	*string*	Información de éxito de la solicitud ('success' o 'failed').

Ejemplo:

server→proxy:

```
{
  "request":"proxy tasks"
}
```

proxy→server:

```
{
  "version":"7.0.0",
  "clock":1721059872,
  "ns":848141522
}
```

server→proxy:

```
{
  "response":"success"
}
```

Proxy activo

Solicitud de configuración

La solicitud `proxy config` es enviada por el proxy activo para obtener los datos de configuración del proxy. Esta solicitud se envía cada `ProxyConfigFrequency` segundos (parámetro de configuración del proxy).

name	value type	description
proxy→server:		
request	*string*	'proxy config'
host	*string *	Nombre del proxy.
version	*string*	Versión del proxy (<major>.<minor>.<build>).
session	*string*	Token de sesión de configuración del proxy.
config_revision	*number*	Revisión de la configuración del proxy.
hostmap_revision	*number*	Revisión actual de la asignación de host a proxy.
server→proxy:		
full_sync	*number*	1 - si se envían los datos de configuración completos; ausente en caso contrario (opcional).
data	*array*	Objeto con los datos de las tablas. Ausente si la configuración no ha cambiado (opcional).
<table>	*object*	Uno o más objetos con datos de <table> (opcional, según los cambios).
fields	*array*	Matriz de nombres de campos.
-	*string*	Nombre del campo.
data	*array*	Matriz de filas.
-	*array*	Matriz de columnas.
-	*string,number*	Valor de la columna con el tipo correspondiente al tipo de columna del esquema de la base de datos.
macro.secrets	*object*	Información sobre macros secretas, ausente si no hay cambios en las macros de vault (opcional).
proxy_group	*string*	Nombre del grupo de proxies al que pertenece el proxy.

name	value type	description
config_revision	*number*	Revisión de la caché de configuración; se envía con los datos de configuración (opcional).
del_hostids	*array*	Matriz de hostids eliminados (opcional).
-	*number*	Identificador del host.
del_macro_hostids	*array*	Matriz de hostids a los que se les han eliminado todas las macros (opcional).
-	*number*	Identificador del host.

Ejemplo:

proxy→server:

```
{
  "request": "proxy config",
  "host": "Zabbix proxy",
  "version":"8.0.0",
  "session": "fd59a09ff4e9d1fb447de1f04599bcf6",
  "config_revision": 0
}
```

server→proxy:

```
{
"full_sync": 1,
"data": {
"hosts": {
"fields": ["hostid", "host", "status", "ipmi_authtype", "ipmi_privilege", "ipmi_username", "ipmi_password"
"data": [
[10084, "Zabbix server", 0, -1, 2, "", "", "Zabbix server", 1, 1, "", "", "", ""]
]
},
"interface": {
"fields": ["interfaceid", "hostid", "main", "type", "useip", "ip", "dns", "port", "available"],
"data": [
[1, 10084, 1, 1, 1, "127.0.0.1", "", "10053", 1]
]
},
"interface_snmp": {
"fields": ["interfaceid", "version", "bulk", "community", "securityname", "securitylevel", "authpassphrase
"data": []
},
"host_inventory": {
"fields": ["hostid", "type", "type_full", "name", "alias", "os", "os_full", "os_short", "serialno_a", "ser
"data": [
[10084, "", "", ..., "56.95387", "24.22067", ..., ""]
]
},
"items": {
"fields": ["itemid", "type", "snmp_oid", "hostid", "key_", "delay", "history", "status", "value_type", "tr
"data": [
[44161, 7, "", 10084, "agent.hostmetadata", "10s", "90d", 0, 1, "", "", "", "", 0, "", "", "", "", 0, null
[44162, 0, "", 10084, "agent.ping", "10s", "90d", 0, 3, "", "", "", "", 0, "", "", "", "", 0, 1, 0, "", nu
]
},
"item_rtdata": {
"fields": ["itemid", "lastlogsize", "mtime"],
"data": [
[44161, 0, 0],
[44162, 0, 0]
]
},
"item_preproc": {
```

```
"fields": ["item_preprocid", "itemid", "step", "type", "params", "error_handler", "error_handler_params"],
"data": []
},
"item_parameter": {
"fields": ["item_parameterid", "itemid", "name", "value"],
"data": []
},
"globalmacro": {
"fields": ["globalmacroid", "macro", "value", "type"],
"data": [
[2, "{$SNMP_COMMUNITY}", "public", 0]
]
},
"hosts_templates": {
"fields": ["hosttemplateid", "hostid", "templateid", "link_type"],
"data": []
},
"hostmacro": {
"fields": ["hostmacroid", "hostid", "macro", "value", "type", "automatic"],
"data": [
[5676, 10084, "{$M}", "AppID=zabbix_server&Query=Safe=passwordSafe;Object=zabbix:Content", 2, 0]
]
},
"drules": {
"fields": ["druleid", "name", "iprange", "delay"],
"data": [
[2, "Local network", "127.0.0.1", "10s"]
]
},
"dchecks": {
"fields": ["dcheckid", "druleid", "type", "key_", "snmp_community", "ports", "snmpv3_securityname", "snmpv
"data": [
[2, 2, 9, "system.uname", "", "10052", "", 0, "", "", 0, 0, 0, "", 1, 0]
]
},
"regexps": {
"fields": ["regexpid", "name"],
"data": [
[1, "File systems for discovery"],
[2, "Network interfaces for discovery"],
[3, "Storage devices for SNMP discovery"],
[4, "Windows service names for discovery"],
[5, "Windows service startup states for discovery"]
]
},
"expressions": {
"fields": ["expressionid", "regexpid", "expression", "expression_type", "exp_delimiter", "case_sensitive"]
"data": [
[1, 1, "^(btrfs|ext2|ext3|ext4|reiser|xfs|ffs|ufs|jfs|jfs2|vxfs|hfs|apfs|refs|ntfs|fat32|zfs)$", 3, ",", 0
[3, 3, "^(Physical memory|Virtual memory|Memory buffers|Cached memory|Swap space)$", 4, ",", 1],
[5, 4, "^(MMCSS|gupdate|SysmonLog|clr_optimization_v2.0.50727_32|clr_optimization_v4.0.30319_32)$", 4, ","
[6, 5, "^(automatic|automatic delayed)$", 3, ",", 1],
[7, 2, "^Software Loopback Interface", 4, ",", 1],
[8, 2, "^(In)?[Ll]oop[Bb]ack[0-9._]*$", 4, ",", 1],
[9, 2, "^NULL[0-9.]*$", 4, ",", 1],
[10, 2, "^[Ll]o[0-9.]*$", 4, ",", 1],
[11, 2, "^[Ss]ystem$", 4, ",", 1],
[12, 2, "^Nu[0-9.]*$", 4, ",", 1]
]
},
"settings": {
"fields": ["name", "type", "value_str", "value_int"],
```

```
"data": [
["autoreg_tls_accept", 2, "", 1],
["hk_history_global", 2, "", 0],
["snmptrap_logging", 2, "", 1],
["proxy_secrets_provider", 2, "", 0],
["hk_history", 1, "31d", 0],
["timeout_db_monitor", 1, "3s", 0],
["timeout_external_check", 1, "3s", 0],
["timeout_http_agent", 1, "3s", 0],
["timeout_simple_check", 1, "3s", 0],
["timeout_snmp_agent", 1, "3s", 0],
["timeout_ssh_agent", 1, "3s", 0],
["timeout_telnet_agent", 1, "3s", 0],
["timeout_zabbix_agent", 1, "3s", 0],
["timeout_browser", 1, "30s", 0]
]
},
"httptest": {
"fields": ["httptestid", "name", "delay", "agent", "authentication", "http_user", "http_password", "hostid
"data": []
},
"httptestitem": {
"fields": ["httptestitemid", "httptestid", "itemid", "type"],
"data": []
},
"httptest_field": {
"fields": ["httptest_fieldid", "httptestid", "type", "name", "value"],
"data": []
},
"httpstep": {
"fields": ["httpstepid", "httptestid", "name", "no", "url", "timeout", "posts", "required", "status_codes"
"data": []
},
"httpstepitem": {
"fields": ["httpstepitemid", "httpstepid", "itemid", "type"],
"data": []
},
"httpstep_field": {
"fields": ["httpstep_fieldid", "httpstepid", "type", "name", "value"],
"data": []
},
"config_autoreg_tls": {
"fields": ["autoreg_tlsid", "tls_psk_identity", "tls_psk"],
"data": [
[1, "", ""]
]
}
},
"macro.secrets": {
"AppID=zabbix_server&Query=Safe=passwordSafe;Object=zabbix": {
"Content": "738"
}
},
"config_revision": 2
}
```


Solicitud de datos

La solicitud `proxy data` se envía desde el proxy para proporcionar datos de disponibilidad de interfaces de host, historial, descubrimiento y autoregistro. Esta solicitud se envía cada `DataSenderFrequency` segundos (parámetro de configuración del proxy). Tenga en cuenta que el proxy activo seguirá consultando al server de Zabbix cada segundo para tareas de comandos remotos (con una solicitud `proxy data` vacía).

name	value type	description
proxy→server:		
request	*string*	'proxy data'
host	*string*	Nombre del proxy.
session	*string*	Token de sesión de datos.
interface availability	*array*	*(optional)* Matriz de objetos de disponibilidad de interfaces.
interfaceid	*number*	Identificador de la interfaz.
available	*number*	Disponibilidad de la interfaz: **0**, *INTERFACE_AVAILABLE_UNKNOWN* - desconocida **1**, *INTERFACE_AVAILABLE_TRUE* - disponible **2**, *INTERFACE_AVAILABLE_FALSE* - no disponible
error	*string*	Mensaje de error de la interfaz o cadena vacía.
history data	*array*	*(optional)* Matriz de objetos de datos históricos.
itemid	*number*	Identificador del item.
clock	*number*	Marca de tiempo del valor del item (segundos).
ns	*number*	Marca de tiempo del valor del item (nanosegundos).
value	*string*	*(optional)* Valor del item.
id	*number*	Identificador del valor (contador ascendente, único dentro de una sesión de datos).
timestamp	*number*	*(optional)* Marca de tiempo de items de tipo log.
source	*string*	*(optional)* Valor de origen del item eventlog.
severity	*number*	*(optional)* Valor de severidad del item eventlog.
eventid	*number*	*(optional)* Valor eventid del item eventlog.
state	*string*	*(optional)* Estado del item: **0**, *ITEM_STATE_NORMAL* **1**, *ITEM_STATE_NOTSUPPORTED*
lastlogsize	*number*	*(optional)* Último tamaño de log de items de tipo log.
mtime	*number*	*(optional)* Hora de modificación de items de tipo log.
discovery data	*array*	*(optional)* Matriz de objetos de datos de descubrimiento.
clock	*number*	Marca de tiempo de los datos de descubrimiento.
druleid	*number*	Identificador de la regla de descubrimiento.
dcheckid	*number*	Identificador de la comprobación de descubrimiento o null para datos de la regla de descubrimiento.
type	*number*	Tipo de comprobación de descubrimiento: **-1** datos de la regla de descubrimiento **0**, *SVC_SSH* - comprobación del servicio SSH **1**, *SVC_LDAP* - comprobación del servicio LDAP **2**, *SVC_SMTP* - comprobación del servicio SMTP **3**, *SVC_FTP* - comprobación del servicio FTP **4**, *SVC_HTTP* - comprobación del servicio HTTP **5**, *SVC_POP* - comprobación del servicio POP **6**, *SVC_NNTP* - comprobación del servicio NNTP **7**, *SVC_IMAP* - comprobación del servicio IMAP **8**, *SVC_TCP* - comprobación de disponibilidad del puerto TCP **9**, *SVC_AGENT* - agent de Zabbix **10**, *SVC_SNMPv1* - agent SNMPv1 **11**, *SVC_SNMPv2* - agent SNMPv2 **12**, *SVC_ICMPPING* - ping ICMP **13**, *SVC_SNMPv3* - agent SNMPv3 **14**, *SVC_HTTPS* - comprobación del servicio HTTPS **15**, *SVC_TELNET* - comprobación de disponibilidad de Telnet
ip	*string*	Dirección IP del host.
dns	*string*	Nombre DNS del host.
port	*number*	*(optional)* Número de puerto del servicio.
key_	*string*	*(optional)* Clave del item para la comprobación de descubrimiento de tipo **9** *SVC_AGENT*

name	value type	description
value	*string*	*(optional)* Valor recibido del servicio, puede estar vacío para la mayoría de los servicios.
status	*number*	*(optional)* Estado del servicio: **0**, *DOBJECT_STATUS_UP* - servicio UP **1**, *DOBJECT_STATUS_DOWN* - servicio DOWN
autoregistration	*array*	*(optional)* Matriz de objetos de autoregistro.
clock	*number*	Marca de tiempo de los datos de autoregistro.
host	*string*	Nombre del host.
ip	*string*	*(optional)* Dirección IP del host.
dns	*string*	*(optional)* Nombre DNS resuelto a partir de la dirección IP.
port	*string*	*(optional)* Puerto del host.
host_metadata	*string*	*(optional)* Metadatos del host enviados por el agent (basados en el parámetro de configuración del agent HostMetadata o HostMetadataItem).
tasks	*array*	*(optional)* Matriz de tareas.
type	*number*	Tipo de tarea: **0**, *ZBX_TM_TASK_PROCESS_REMOTE_COMMAND_RESULT* - resultado de comando remoto
status	*number*	Estado de ejecución del comando remoto: **0**, *ZBX_TM_REMOTE_COMMAND_COMPLETED* - el comando remoto se completó correctamente **1**, *ZBX_TM_REMOTE_COMMAND_FAILED* - el comando remoto falló
error	*string*	*(optional)* Mensaje de error.
parent_taskid	*number*	ID de la tarea padre.
more	*number*	*(optional)* 1 - hay más datos históricos para enviar
clock	*number*	*(optional)* Marca de tiempo de la transferencia de datos (segundos).
ns	*number*	*(optional)* Marca de tiempo de la transferencia de datos (nanosegundos).
version	*string*	Versión del proxy (<major>.<minor>.<build>).
server→proxy:		
response	*string*	Información de éxito de la solicitud (’success’ o ’failed’).
upload	*string*	Control de carga para datos históricos (history, autoregistration, disponibilidad de host, descubrimiento de red). Valores posibles: **enabled** - funcionamiento normal **disabled** - el server no acepta datos (posiblemente debido a que la caché interna ha superado el límite)
tasks	*array*	*(optional)* Matriz de tareas.
type	*number*	Tipo de tarea: **1**, *ZBX_TM_TASK_PROCESS_REMOTE_COMMAND* - comando remoto
clock	*number*	Hora de creación de la tarea.
ttl	*number*	Tiempo en segundos tras el cual la tarea expira.
commandtype	*number*	Tipo de comando remoto: **0**, *ZBX_SCRIPT_TYPE_CUSTOM_SCRIPT* - usar script personalizado **1**, *ZBX_SCRIPT_TYPE_IPMI* - usar IPMI **2**, *ZBX_SCRIPT_TYPE_SSH* - usar SSH **3**, *ZBX_SCRIPT_TYPE_TELNET* - usar Telnet **4**, *ZBX_SCRIPT_TYPE_GLOBAL_SCRIPT* - usar script global (actualmente funcionalmente equivalente a un script personalizado)
command	*string*	Comando remoto a ejecutar.
execute_on	*number*	Destino de ejecución para scripts personalizados: **0**, *ZBX_SCRIPT_EXECUTE_ON_AGENT* - ejecutar el script en el agent **1**, *ZBX_SCRIPT_EXECUTE_ON_SERVER* - ejecutar el script en el server **2**, *ZBX_SCRIPT_EXECUTE_ON_PROXY* - ejecutar el script en el proxy
port	*number*	*(optional)* Puerto para comandos Telnet y SSH.
authtype	*number*	*(optional)* Tipo de autenticación para comandos SSH.

name	value type	description
username	*string*	*(optional)* Nombre de usuario para comandos Telnet y SSH.
password	*string*	*(optional)* Contraseña para comandos Telnet y SSH.
publickey	*string*	*(optional)* Clave pública para comandos SSH.
privatekey	*string*	*(optional)* Clave privada para comandos SSH.
parent_taskid	*number*	ID de la tarea padre.
hostid	*number*	ID del host de destino.

Ejemplo:

proxy→server:


```
{
"request": "proxy data",
"host": "Zabbix proxy",
"session": "818cdd1b537bdc5e50c09ed4969235b6",
"interface availability": [{
"interfaceid": 1,
"available": 1,
"error": ""
}],
"history data": [{
"id": 1114,
"itemid": 44162,
"clock": 1665730632,
"ns": 798953105,
"value": "1"
}, {
"id": 1115,
"itemid": 44161,
"clock": 1665730633,
"ns": 811684663,
"value": "58"
}],
"auto registration": [{
"clock": 1665730633,
"host": "Zabbix server",
"ip": "127.0.0.1",
"dns": "localhost",
"port": "10053",
"host_metadata": "58",
"tls_accepted": 1
}],
"discovery data": [{
"clock": 1665732232,
"drule": 2,
"dcheck": 2,
"ip": "127.0.0.1",
"dns": "localhost",
"port": 10052,
"status": 1
}, {
"clock": 1665732232,
"drule": 2,
"dcheck": null,
"ip": "127.0.0.1",
"dns": "localhost",
"status": 1
}],
"host data": [{
"hostid": 10084,
"active_status": 1
}],
```

```
"tasks": [{
"type": 3,
"clock": 1665730985,
"ttl": 0,
"status": -1,
"info": "Remote commands are not enabled",
"parent_taskid": 3
}],
"version": "8.0.0",
"clock": 1665730643,
"ns": 65389964
}
```


server→proxy:


```
{
"upload": "enabled",
"response": "success",
"tasks": [{
"type": 2,
"clock": 1665730986,
"ttl": 600,
"commandtype": 0,
"command": "ping -c 3 127.0.0.1; case $? in [01]) true;; *) false;; esac",
"execute_on": 2,
"port": 0,
"authtype": 0,
"username": "",
"password": "",
"publickey": "",
"privatekey": "",
"alertid": 0,
"parent_taskid": 4,
"hostid": 10084
}]
}
```


Parámetros de configuración: Zabbix proxy

Descripción general

Los parámetros de configuración permiten personalizar el comportamiento del proxy de Zabbix.

Especifique los valores de los parámetros en el archivo de configuración del proxy de Zabbix (`zabbix_proxy.conf`). Todos los parámetros son **opcionales**, a menos que se indique explícitamente que el parámetro es obligatorio.

Esta página está estructurada de la siguiente manera:

- Descripción general de los parámetros (para consultar todos los detalles de un parámetro, haga clic en su nombre)
- Detalles de los parámetros

Parámetro	Descripción
AllowRoot	Permitir que el proxy se ejecute como `root`.
AllowUnsupportedDBVersions	Permitir que el proxy funcione con versiones de base de datos no compatibles.
APMListenIP	Especificar la dirección IP en la que el recopilador de APM debe escuchar los datos de telemetría entrantes.
APMListenPort	Especificar el puerto en el que el recopilador de APM debe escuchar los datos de telemetría entrantes.
APMTLSCAFile	Especificar la ruta completa del archivo que contiene los certificados de las CA de nivel superior para verificar los certificados de las aplicaciones o los recopiladores de OpenTelemetry que envían datos al recopilador de APM.
APMTLSCertFile	Especificar la ruta completa del archivo que contiene el certificado o la cadena de certificados del recopilador de APM.
APMTLSKeyFile	Especificar la ruta completa del archivo que contiene la clave privada del recopilador de APM.

Parámetro	Descripción
CacheSize	Establecer el tamaño de la caché de configuración.
ConfigFrequency	Este parámetro está **obsoleto** (utilice ProxyConfigFrequency en su lugar). Establecer la frecuencia de recuperación de los datos de configuración del server de Zabbix.
DataSenderFrequency	Establecer la frecuencia de envío de los datos recopilados al server de Zabbix.
DBHost	Especificar el host de la base de datos (o el directorio del socket).
DBName	Especificar el nombre de la base de datos o la ruta al archivo de base de datos para SQLite3.
DBPassword	Especificar la contraseña de la base de datos.
DBPort	Especificar el puerto del host de la base de datos de Zabbix.
DBSchema	Especificar el nombre del esquema de la base de datos. Se utiliza para PostgreSQL.
DBSocket	Especificar la ruta al archivo de socket de MySQL.
DBUser	Especificar el usuario de la base de datos.
DBTLSConnect	Especificar un valor para forzar una conexión TLS a la base de datos.
DBTLSCAFile	Especificar la ruta completa del archivo que contiene los certificados de las CA de nivel superior para la verificación del certificado de la base de datos.
DBTLSCertFile	Especificar la ruta completa del archivo que contiene el certificado del proxy de Zabbix para autenticarse en la base de datos.
DBTLSKeyFile	Especificar la ruta completa del archivo que contiene la clave privada para autenticarse en la base de datos.
DBTLSCipher	Especificar la lista de cifrados de encriptación que el proxy de Zabbix permite para los protocolos TLS hasta TLS v1.2. Compatible únicamente con MySQL.
DBTLSCipher13	Especificar la lista de suites de cifrado que el proxy de Zabbix permite para el protocolo TLS v1.3. Compatible únicamente con MySQL a partir de la versión 8.0.16.
DebugLevel	Establecer el nivel de depuración.
EnableRemoteCommands	Permitir comandos remotos del server de Zabbix.
ExternalScripts	Especificar la ubicación de los scripts externos.
Fping6Location	Especificar la ubicación de fping6.
FpingLocation	Especificar la ubicación de fping.
HistoryCacheSize	Establecer el tamaño de la caché del historial.
HistoryIndexCacheSize	Establecer el tamaño de la caché de índices del historial.
Hostname	Especificar el nombre del proxy.
HostnameItem	Especificar el item utilizado para establecer Hostname (si no está definido).
HousekeepingFrequency	Establecer la frecuencia del procedimiento de mantenimiento (en horas).
Include	Especificar un directorio o archivos individuales que se incluirán en el archivo de configuración.
JavaGateway	Especificar la dirección IP (o el nombre de host) del Java gateway de Zabbix.
JavaGatewayPort	Especificar el puerto del Java gateway de Zabbix.
ListenBacklog	Establecer el número máximo de conexiones pendientes en la cola TCP.
ListenIP	Especificar la dirección IP de escucha del trapper.
ListenPort	Especificar el puerto de escucha del trapper.
LoadModule	Especificar el módulo que se cargará al iniciar el proxy.
LoadModulePath	Especificar la ruta completa a la ubicación de los módulos del proxy.
LogFile	Especificar el archivo de registro del proxy.
LogFileSize	Establecer el tamaño máximo del archivo de registro.
LogRemoteCommands	Habilitar el registro de los comandos de shell ejecutados como advertencias.
LogSlowQueries	Establecer la duración máxima que puede tener una consulta de base de datos antes de registrarse (en milisegundos).
LogType	Especificar el tipo de salida del registro.
MaxConcurrentChecksPerPoller	Establecer el número máximo de comprobaciones asíncronas que cada poller de HTTP agent, poller de agent o poller de SNMP puede ejecutar simultáneamente.
PidFile	Especificar el archivo PID.
ProxyBufferMode	Especificar el mecanismo de almacenamiento de los datos de historial, descubrimiento y registro automático (disco/memoria/híbrido).
ProxyConfigFrequency	Establecer la frecuencia de recuperación de los datos de configuración del server de Zabbix, en segundos.
ProxyLocalBuffer	Establecer la duración de conservación local de los datos (en horas), incluso si ya se han sincronizado con el server.
ProxyMemoryBufferAge	Establecer la antigüedad máxima de los datos en el búfer de memoria del proxy, en segundos.
ProxyMemoryBufferSize	Establecer el tamaño de la caché de memoria compartida para los datos recopilados de historial, descubrimiento y registro automático.
ProxyMode	Establecer el modo de funcionamiento del proxy (activo/pasivo).
ProxyOfflineBuffer	Establecer la duración de conservación de los datos (en horas) en caso de no haber conectividad con el server de Zabbix.

Parámetro	Descripción
Server	Si ProxyMode está establecido en `active`: especificar la dirección IP o el nombre DNS del server de Zabbix (address:port) o el clúster (address:port;address2:port) del que se obtendrán los datos de configuración y al que se enviarán los datos. Si ProxyMode está establecido en `passive`: especificar las direcciones IP o los nombres DNS del server de Zabbix.
SNMPTrapperFile	Especificar el archivo temporal utilizado para pasar datos del demonio de traps SNMP al proxy.
SocketDir	Especificar el directorio para almacenar los sockets IPC utilizados por los servicios internos de Zabbix.
SourceIP	Especificar la dirección IP de origen.
SSHKeyLocation	Especificar la ubicación de las claves públicas y privadas para las comprobaciones y acciones SSH.
SSLCertLocation	Especificar la ubicación de los archivos de certificados de cliente SSL para la autenticación del cliente.
SSLKeyLocation	Especificar la ubicación de los archivos de claves privadas SSL para la autenticación del cliente.
SSLCALocation	Especificar la ubicación de los archivos de la autoridad certificadora (CA) para la verificación del certificado del server SSL.
StartAgentPollers	Establecer el número de instancias prefork de pollers asíncronos de Zabbix agent.
StartAPMCollector	Establecer en 1 para iniciar el recopilador de APM.
StartBrowserPollers	Establecer el número de instancias prefork de pollers de items de navegador.
StartDBSyncers	Establecer el número de instancias prefork de sincronizadores del historial.
StartDiscoverers	Establecer el número de instancias prefork de trabajadores de descubrimiento.
StartHTTPAgentPollers	Establecer el número de instancias prefork de pollers asíncronos de HTTP agent.
StartHTTPPollers	Establecer el número de instancias prefork de pollers HTTP.
StartIPMIPollers	Establecer el número de instancias prefork de pollers IPMI.
StartJavaPollers	Establecer el número de instancias prefork de pollers Java.
StartODBCPollers	Establecer el número de instancias prefork de pollers ODBC.
StartPingers	Establecer el número de instancias prefork de pingers ICMP.
StartPollersUnreachable	Establecer el número de instancias prefork de pollers para hosts inaccesibles (incluidos IPMI y Java).
StartPollers	Establecer el número de instancias prefork de pollers.
StartPreprocessors	Establecer el número de instancias iniciadas previamente de trabajadores de preprocesamiento.
StartSNMPPollers	Establecer el número de instancias prefork de pollers asíncronos SNMP.
StartSNMPTrapper	Establecer en ”1” para iniciar un proceso trapper SNMP.
StartTelemetryQueryPollers	Establecer el número de instancias prefork de pollers asíncronos de consultas de telemetría.
StartTrappers	Establecer el número de instancias prefork de trappers.
StartVMwareCollectors	Establecer el número de instancias prefork de recopiladores de VMware.
StatsAllowedIP	Especificar las direcciones IP o los nombres DNS de las instancias externas de Zabbix autorizadas a solicitar estadísticas.
TelemetryProvider	Establecer el origen de los datos de telemetría.
Timeout	Especificar cuánto tiempo esperar (en segundos) para establecer la conexión e intercambiar datos con el proxy de Zabbix, el agent, el servicio web y para las comprobaciones SNMP (excepto los items SNMP `walk[OID]` y `get[OID]`).
TLSAccept	Especificar el nivel de encriptación requerido para las conexiones entrantes del server de Zabbix.
TLSCAFile	Especificar la ruta completa del archivo que contiene los certificados de las CA de nivel superior para la verificación del certificado del par, utilizados para las comunicaciones cifradas entre los componentes de Zabbix.
TLSCertFile	Especificar la ruta completa del archivo que contiene el certificado del server o la cadena de certificados, utilizados para las comunicaciones cifradas entre los componentes de Zabbix.
TLSCipherAll	Especificar la cadena de prioridad de GnuTLS o la cadena de cifrado de OpenSSL (TLS 1.2). Anular los criterios predeterminados de selección de suites de cifrado para la encriptación basada en certificados y PSK.
TLSCipherAll13	Especificar la cadena de cifrado para OpenSSL 1.1.1 o posterior en TLS 1.3. Anular los criterios predeterminados de selección de suites de cifrado para la encriptación basada en certificados y PSK.
TLSCipherCert	Especificar la cadena de prioridad de GnuTLS o la cadena de cifrado de OpenSSL (TLS 1.2). Anular los criterios predeterminados de selección de suites de cifrado para la encriptación basada en certificados.
TLSCipherCert13	Especificar la cadena de cifrado para OpenSSL 1.1.1 o posterior en TLS 1.3. Anular los criterios predeterminados de selección de suites de cifrado para la encriptación basada en certificados.
TLSCipherPSK	Especificar la cadena de prioridad de GnuTLS o la cadena de cifrado de OpenSSL (TLS 1.2). Anular los criterios predeterminados de selección de suites de cifrado para la encriptación basada en PSK.

Parámetro	Descripción
TLSCipherPSK13	Especificar la cadena de cifrado para OpenSSL 1.1.1 o posterior en TLS 1.3. Anular los criterios predeterminados de selección de suites de cifrado para la encriptación basada en PSK.
TLSConnect	Especificar el nivel de encriptación requerido para las conexiones con el server de Zabbix.
TLSCRLFile	Especificar la ruta completa del archivo que contiene los certificados revocados. Este parámetro se utiliza para las comunicaciones cifradas entre los componentes de Zabbix.
TLSKeyFile	Especificar la ruta completa del archivo que contiene la clave privada del proxy, utilizada para las comunicaciones cifradas entre los componentes de Zabbix.
TLSListen	Requerir conexiones exclusivamente TLS en el socket del trapper.
TLSPSKFile	Especificar la ruta completa del archivo que contiene la clave precompartida del proxy, utilizada para las comunicaciones cifradas con el server de Zabbix.
TLSPSKIdentity	Especificar la cadena de identidad de la clave precompartida, utilizada para las comunicaciones cifradas con el server de Zabbix.
TLSServerCertIssuer	Especificar el emisor permitido del certificado del server.
TLSServerCertSubject	Especificar el sujeto permitido del certificado del server.
TmpDir	Especificar el directorio temporal.
TrapperTimeout	Especificar el tiempo de espera en segundos para: - la recuperación de los datos de configuración del server de Zabbix; - la ejecución de scripts globales o la ejecución de comandos remotos.
UnavailableDelay	Establecer la frecuencia de comprobación de la disponibilidad del host durante el período de indisponibilidad.
UnreachableDelay	Establecer la frecuencia de comprobación de la disponibilidad del host durante el período de inaccesibilidad.
UnreachablePeriod	Establecer después de cuántos segundos de inaccesibilidad se debe considerar que el host no está disponible.
User	Abandonar los privilegios y utilizar un usuario específico existente en el sistema.
Vault	Especificar el proveedor de vault.
VaultAppRoleID	Especificar el ID de AppRole del vault de HashiCorp.
VaultAppSecretID	Especificar el ID secreto de AppRole del vault de HashiCorp.
VaultDBPath	Especificar la ubicación para recuperar las credenciales de la base de datos mediante claves.
VaultPrefix	Especificar un prefijo personalizado para la ruta o consulta del vault.
VaultTLSCertFile	Especificar el nombre del archivo de certificado SSL utilizado para la autenticación del cliente.
VaultTLSKeyFile	Especificar el nombre del archivo de clave privada SSL utilizado para la autenticación del cliente.
VaultToken	Especificar el token de autenticación del vault de HashiCorp.
VaultURL	Especificar la URL del server de vault.
VMwareCacheSize	Establecer el tamaño de la memoria compartida para almacenar datos de VMware.
VMwareFrequency	Establecer la frecuencia de recopilación de datos de un único servicio de VMware, en segundos.
VMwarePerfFrequency	Establecer la frecuencia de recuperación de estadísticas de contadores de rendimiento de un único servicio de VMware, en segundos.
VMwareTimeout	Establecer el número máximo de segundos que un `vmware collector` esperará una respuesta del servicio de VMware.
WebDriverURL	Especificar la URL de la interfaz de WebDriver.

Tenga en cuenta lo siguiente:

- Los valores predeterminados reflejan los valores predeterminados del demonio, no los valores de los archivos de configuración distribuidos.
- Los valores admiten variables de entorno.
- Zabbix admite archivos de configuración únicamente con codificación UTF-8 y sin BOM.
- Los comentarios que comienzan con # solo se admiten al principio de la línea.

Detalles del parámetro

AllowRoot

Permite que el proxy se ejecute como `root`. Si está deshabilitado y el proxy se inicia como `root`, el proxy intentará cambiar al usuario `zabbix`. No tiene efecto si se inicia con un usuario normal.

Predeterminado: 0
 Valores: 0 - no permitir; 1 - permitir

AllowUnsupportedDBVersions

Permitir que proxy funcione con versiones de base de datos no compatibles.

Predeterminado: 0
 Valores: 0 - no permitir; 1 - permitir

APMListenIP

Especifique la dirección IP en la que el APM collector debe escuchar los datos de telemetría entrantes.
 Si no se especifica, el APM collector escuchará en todas las interfaces de red.

Valor predeterminado: `0.0.0.0`

APMListenPort

Especifique el puerto en el que el colector APM debe escuchar los datos de telemetría entrantes.
 El puerto predeterminado es el puerto estándar de OpenTelemetry (OTLP/gRPC), utilizado de forma predeterminada por los SDK y colectores de OpenTelemetry.

Predeterminado: 4317
 Rango: 1024-32767

APMTLSCAFile

Especifique la ruta completa del archivo que contiene los certificados de la CA de nivel superior para verificar los certificados de las aplicaciones o los recopiladores de OpenTelemetry que envían datos al recopilador de APM.
 Si se establece este parámetro, el recopilador de APM acepta únicamente conexiones cifradas; de lo contrario, acepta únicamente conexiones no cifradas.

APMTLSCertFile

Especifique la ruta completa del archivo que contiene el certificado o la cadena de certificados del colector de APM.
 Este parámetro debe configurarse junto con `APMTLSKeyFile` y solo si `APMTLSCAFile` está configurado.

APMTLSKeyFile

Especifique la ruta completa del archivo que contiene la clave privada del collector de APM.
 Este parámetro debe configurarse junto con `APMTLSCertFile` y solo si `APMTLSCAFile` está configurado.

CacheSize

Establece el tamaño de la caché de configuración, en bytes. El tamaño de la memoria compartida para almacenar datos de host e item.

Predeterminado: 32M
 Rango: 128K-64G

ConfigFrequency

Este parámetro está **obsoleto** (use `ProxyConfigFrequency` en su lugar).
Establece la frecuencia de recuperación de los datos de configuración desde el server de Zabbix (en segundos).
Parámetro de proxy activo. Se ignora para los proxies pasivos (consulte el parámetro `ProxyMode`).

Predeterminado: 3600
 Rango: 1-604800

DataSenderFrequency

Establece la frecuencia de envío de los datos recopilados desde el proxy al server de Zabbix. Tenga en cuenta que un proxy activo seguirá consultando al server de Zabbix cada segundo para tareas de comandos remotos.
Parámetro de proxy activo. Se ignora para proxies pasivos (consulte el parámetro `ProxyMode`).

Predeterminado: 1
 Rango: 1-3600

DBHost

Especifique el host de la base de datos (o el directorio del socket).

Para MySQL:

- `localhost` o una cadena vacía usa el socket UNIX de dominio predeterminado.

Para PostgreSQL:

- `localhost` se resuelve mediante DNS (normalmente a `127.0.0.1`).
- Una cadena vacía usa el socket UNIX de dominio predeterminado.
- Una ruta (por ejemplo, `/var/run/pgbouncer`) hace que se use el socket UNIX en esa ruta.
- Una lista separada por comas puede contener varios valores `host:port`. Cada host se prueba en orden hasta que se establece una conexión de lectura y escritura. Ejemplo: `DBHost=localhost:5431,127.0.0.1:20051,zabbix.domain,[::1]:300`

Predeterminado: `localhost`

DBName

Especifique el nombre de la base de datos o la ruta al archivo de base de datos para SQLite3 (la arquitectura multiproceso de Zabbix no permite utilizar una base de datos en memoria, por ejemplo, `:memory:`, `file::memory:?cache=shared` o `file:memdb1?mode=memory&cache=shared`).
Advertencia: No intente utilizar la misma base de datos que está utilizando el server de Zabbix.

Obligatorio: Sí

DBPassword

Especifique la contraseña de la base de datos. Comente esta línea si no se utiliza ninguna contraseña. Se ignora para SQLite.

DBPort

Especifique el puerto del host de la base de datos de Zabbix (consulte DBHost). Se ignora para SQLite.
 DBSocket y DBPort son mutuamente excluyentes en la configuración de proxy. Especifique solo uno, o deje ambos sin definir.

Valor predeterminado para MySQL: 3306

Valor predeterminado para PostgreSQL: 5432

Rango: 1024-65535

DBSchema

Especifique el nombre del esquema de la base de datos. Se utiliza para PostgreSQL.

DBSocket

Especifique la ruta al archivo de socket de MySQL.
DBSocket y DBPort se excluyen mutuamente en la configuración de proxy. Especifique solo uno, o deje ambos sin definir.

DBUser

Especifique el usuario de la base de datos. Se ignora para SQLite.

DBTLSConnect

Especifique un valor para forzar una conexión TLS a la base de datos: - *required* - conectarse usando TLS; - *verify_ca* - conectarse usando TLS y verificar el certificado; - *verify_full* - conectarse usando TLS, verificar el certificado y verificar que la identidad de la base de datos especificada por DBHost coincida con su certificado
En MySQL a partir de la versión 5.7.11 y en PostgreSQL se admiten los siguientes valores: ”required”, ”verify”, ”verify_full”.
En MariaDB a partir de la versión 10.2.6 se admiten los valores ”required” y ”verify_full”.
De forma predeterminada, no se establece ninguna opción y el comportamiento depende de la configuración de la base de datos.

DBTLSCAFile

Especifique la ruta completa del archivo que contiene los certificados de la CA de nivel superior para la verificación del certificado de la base de datos.

Obligatorio: no (sí, si DBTLSConnect está configurado en *verify_ca* o *verify_full*)

DBTLSCertFile

Especifique la ruta completa del archivo que contiene el certificado del proxy de Zabbix para autenticarse en la base de datos.

DBTLSKeyFile

Especifique la ruta completa del archivo que contiene la clave privada para autenticarse en la base de datos.

DBTLSCipher

Especifique la lista de cifrados de cifrado que Zabbix proxy permite para los protocolos TLS hasta TLS v1.2. Solo compatible con MySQL.

DBTLSCipher13

Especifique la lista de conjuntos de cifrado que Zabbix proxy permite para el protocolo TLS v1.3. Solo es compatible con MySQL, a partir de la versión 8.0.16.

DebugLevel

Establezca el nivel de depuración: - *0* - información básica sobre el inicio y la detención de los procesos de Zabbix; - *1* - información crítica; - *2* - información de error; - *3* - advertencias; - *4* - para depuración (genera mucha información); - *5* - depuración extendida (genera aún más información).

Predeterminado: 3
 Rango: 0-5

EnableRemoteCommands

Permitir comandos remotos desde el server de Zabbix.

Predeterminado: 0
 Valores: 0 - no permitir; 1 - permitir

ExternalScripts

Especifique la ubicación del script externo (depende de la variable de instalación en tiempo de compilación `datadir`).

Predeterminado: `/usr/local/share/zabbix/externalscripts`

Fping6Location

Especifique la ubicación de fping6. Asegúrese de que el binario fping6 tenga como propietario a root y que el indicador SUID esté establecido. Déjelo vacío (`Fping6Location=`) si su utilidad `fping` es capaz de procesar direcciones IPv6.

Predeterminado: `/usr/sbin/fping6`

FpingLocation

Especifique la ubicación de fping. Asegúrese de que el binario de fping tenga como propietario a root y que el indicador SUID esté establecido.

Predeterminado: `/usr/sbin/fping`

HistoryCacheSize

Establece el tamaño de la caché de historial, en bytes. El tamaño de la memoria compartida para almacenar datos de historial.

Predeterminado: `16M`
 Rango: 128K-16G

HistoryIndexCacheSize

Establece el tamaño de la caché del índice de historial, en bytes. El tamaño de la memoria compartida para indexar los datos de historial almacenados en la caché de historial. El tamaño de la caché de índices necesita aproximadamente 100 bytes para almacenar en caché un item.

Predeterminado: `4M`
 Rango: 128K-16G

Nombre de host

Especifique un nombre de proxy único y sensible a mayúsculas y minúsculas. Asegúrese de que el nombre de proxy sea conocido por el server.
Caracteres permitidos: alfanuméricos, '.', ' ', '_' y '-'. Longitud máxima: 128

Predeterminado: establecido por HostnameItem

HostnameItem

Especifique un item para establecer `Hostname` (si no está definido).
 Este item se ejecutará en el proxy de forma similar a como lo hace en un agent.
 Se ignora si `Hostname` está definido.
No admite parámetros de usuario, contadores de rendimiento ni alias; sí admite `system.run[]`.

Predeterminado: `system.hostname`

HousekeepingFrequency

Establezca la frecuencia del procedimiento de housekeeping (en horas). Housekeeping consiste en eliminar información obsoleta de la base de datos.
Nota: Para reducir la carga al iniciar el proxy, el housekeeping se pospone durante 30 minutos después del inicio del proxy. Por lo tanto, si `HousekeepingFrequency` es 1, el primer procedimiento de housekeeping después del inicio del proxy se ejecutará después de 30 minutos y se repetirá cada hora a partir de entonces.
Es posible deshabilitar el housekeeping automático estableciendo HousekeepingFrequency en 0. En este caso, el procedimiento de housekeeping solo puede iniciarse mediante la opción de control en tiempo de ejecución `housekeeper_execute`.
Consulte también las opciones de control en tiempo de ejecución y los detalles sobre el procedimiento de housekeeping.

Predeterminado: `1`
 Rango: 0-24

Include

Especifique un directorio o archivos individuales que se incluirán en el archivo de configuración.
 Durante la instalación de Zabbix, este parámetro se establece en `/usr/local/etc`, a menos que lo cambie durante el tiempo de compilación.

Reglas generales para el parámetro `Include`:

- La ruta puede ser relativa a la ubicación del archivo `zabbix_proxy.conf`.
- Puede especificar varios parámetros `Include` en un solo archivo de configuración.
- Zabbix procesa el parámetro `Include` en cuanto lo lee, así que considere colocarlo al final del archivo de configuración.

Reglas para los archivos incluidos:

- Cada archivo debe ser legible.
- Cada archivo debe tener una sintaxis correcta. De lo contrario, el componente no se iniciará.

Reglas para incluir directorios o patrones de archivos:

- No asuma ningún orden de inclusión (por ejemplo, los archivos no se incluyen en orden alfabético).

- No defina el mismo parámetro en más de un archivo incluido (por ejemplo, para sobrescribir una configuración general con una específica), ya que no puede predecir qué valor se usará.
- Tenga cuidado con las copias de seguridad que algunos editores de texto crean automáticamente (por ejemplo, al editar `include/my_specific.conf` puede crearse `include/my_specific.conf.BAK`), ya que ambos archivos se incluirán.

Para incluir solo los archivos relevantes de un directorio, use el comodín (`*`) para la coincidencia de patrones.

Ejemplo:

```
Include=/absolute/path/to/config/files/*.conf
```

JavaGateway

Especifique la dirección IP o el nombre de host de Zabbix Java gateway. Solo es necesario si se han iniciado los Java pollers.

JavaGatewayPort

Especifique el puerto de Java gateway de Zabbix.

Predeterminado: 10052
 Rango: 1024-32767

ListenBacklog

Establece el número máximo de conexiones pendientes en la cola TCP.
El valor predeterminado es una constante codificada, que depende del sistema.
El valor máximo admitido también depende del sistema; valores demasiado altos pueden truncarse silenciosamente al ”maximum specified by the implementation”.

Predeterminado: SOMAXCONN
 Rango: 0 - INT_MAX

ListenIP

Especifique la dirección IP de escucha del trapper. Una lista delimitada por comas.
El trapper escuchará en todas las interfaces de red si no se especifica este parámetro.

Predeterminado: 0.0.0.0

ListenPort

Especifique el puerto de escucha del trapper.

Predeterminado: 10051
 Rango: 1024-32767

LoadModule

Especifique el módulo que se cargará al iniciar el proxy.
Los módulos se utilizan para ampliar la funcionalidad del proxy.
El módulo debe encontrarse en el directorio especificado por LoadModulePath o la ruta debe preceder al nombre del módulo.
Si la ruta precedente es absoluta (comienza por /), LoadModulePath se ignora.
 Formatos: - LoadModule=<module.so> - LoadModule=<path/module.so> - LoadModule=</abs_path/module.so>

Se pueden especificar varios parámetros LoadModule.

LoadModulePath

La ruta completa a la ubicación de los módulos de proxy. El valor predeterminado depende de las opciones de compilación.

LogFile

Especifique el archivo de registro del proxy.

Obligatorio: Sí, si LogType está establecido en file; de lo contrario, no

LogFileSize

Establece el tamaño máximo del archivo de registro (en MB).
0 - deshabilita la rotación automática del registro.
Nota: Si se alcanza el límite de tamaño del archivo de registro y la rotación del archivo falla, por cualquier motivo, el archivo de registro existente se trunca y se inicia uno nuevo.

Predeterminado: 1
 Rango: 0-1024

LogRemoteCommands

Habilita el registro de los comandos de shell ejecutados como advertencias.

Predeterminado: 0
 Valores: 0 - deshabilitado, 1 - habilitado

LogSlowQueries

Establece la duración máxima que puede tardar una consulta de base de datos antes de registrarse (en milisegundos).
0 - no registrar consultas lentas.
Esta opción se habilita a partir de DebugLevel=3.

Predeterminado: 0
 Rango: 0-3600000

LogType

Especifique el tipo de salida del log: - *file* - escribe el log en el archivo especificado por el parámetro LogFile; - *system* - escribe el log en syslog; - *console* - escribe el log en la salida estándar.

Predeterminado: file

MaxConcurrentChecksPerPoller

Establece el número máximo de comprobaciones asíncronas que puede ejecutar a la vez cada HTTP agent poller, agent poller o SNMP poller. Consulte StartHTTPAgentPollers, StartAgentPollers y StartSNMPPollers.

Predeterminado: 1000
 Rango: 1-1000

PidFile

Especifique el archivo PID.

Predeterminado: /tmp/zabbix_proxy.pid

ProxyBufferMode

Especifique el mecanismo de almacenamiento para los datos de history, discovery y autoregistration: - *disk* - los datos se almacenan en la base de datos y se cargan desde la base de datos; - *memory* - los datos se almacenan en memoria y se cargan desde memoria. Si el buffer se queda sin memoria, los datos antiguos se descartarán. Al apagarse, el buffer se descarta. - *hybrid* - el buffer de proxy normalmente funciona como en el modo de memoria hasta que se queda sin memoria o el registro más antiguo supera la antigüedad configurada. Si eso ocurre, el buffer se vacía en la base de datos y funciona como en el modo de disco hasta que todos los datos se hayan cargado y vuelve a funcionar con memoria. Al apagarse, el buffer de memoria se vacía en la base de datos.

Véase también: Proxy memory buffer.

Predeterminado: disk
 Valores: disk; memory; hybrid

ProxyConfigFrequency

Establece la frecuencia de recuperación de los datos de configuración desde el server de Zabbix en segundos.
Parámetro del proxy activo. Se ignora para los proxies pasivos (consulte el parámetro ProxyMode).

Predeterminado: 10
 Rango: 1-604800

ProxyLocalBuffer

Establezca la duración de conservación de los datos localmente (en horas), incluso si los datos ya se han sincronizado con el server..
Este parámetro puede utilizarse si los datos locales serán usados por aplicaciones de terceros.

Predeterminado: 0
 Rango: 0-720

ProxyMemoryBufferAge

Establece la antigüedad máxima de los datos en el búfer de memoria del proxy en segundos.
 Cuando está habilitado (distinto de cero) y los registros en el búfer de memoria del proxy son más antiguos, fuerza al búfer del proxy a cambiar al modo de base de datos hasta que todos los registros se carguen en el server. Este parámetro debe ser menor o igual que el parámetro ProxyOfflineBuffer.

Predeterminado: 0
 Rango: 0;600-864000

ProxyMemoryBufferSize

Establece el tamaño de la caché de memoria compartida para los datos de history, discovery y auto registration recopilados (en bytes). Si está habilitado (distinto de cero), el proxy mantendrá en memoria los datos de history, discovery y autoregistration, a menos que la caché esté llena o que los registros almacenados sean más antiguos que ProxyMemoryBufferAge. Este parámetro no se puede usar junto con el parámetro ProxyLocalBuffer.

Predeterminado: 0
 Rango: 0;128K-2G

ProxyMode

Establezca el modo de funcionamiento del proxy.
0 - proxy en modo activo
1 - proxy en modo pasivo
Nota que los datos de configuración del proxy (sensibles) pueden quedar disponibles para terceros que tengan acceso al puerto trapper del server Zabbix cuando se utiliza un proxy activo. Esto es posible porque cualquiera puede hacerse pasar por un proxy activo y solicitar datos de configuración; no se realiza autenticación.

Predeterminado: 0
 Rango: 0-1

ProxyOfflineBuffer

Establece la duración de conservación de los datos (en horas) en caso de no haber conectividad con el server de Zabbix.
Los datos más antiguos se perderán.

Predeterminado: 1
 Rango: 1-720

Server

Si `ProxyMode` está configurado como `active`: especifique la dirección IP o el nombre DNS del server de Zabbix (address:port) o cluster (address:port;address2:port) desde el que obtener los datos de configuración y al que enviar los datos.
Si no se especifica el puerto, se usa el puerto predeterminado.
Los nodos del clúster deben separarse con un punto y coma.

Si `ProxyMode` está configurado como `passive`: especifique direcciones IP, opcionalmente en notación CIDR, o nombres DNS del server de Zabbix. Una lista delimitada por comas.
 Las conexiones entrantes solo se aceptarán desde las direcciones enumeradas aquí. Si la compatibilidad con IPv6 está habilitada, entonces `127.0.0.1`, `::127.0.0.1`, `::ffff:127.0.0.1` se tratan por igual.
`::/0` permitirá cualquier dirección IPv4 o IPv6. `0.0.0.0/0` se puede usar para permitir cualquier dirección IPv4.

Ejemplo:

```
Server=127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.example.com
```

Obligatorio: sí

SNMPTrapperFile

Especifique el archivo temporal utilizado para pasar datos desde el daemon de SNMP trap al proxy.
Debe ser el mismo que en `zabbix_trap_receiver.pl` o en el archivo de configuración de SNMPTT.

Predeterminado: `/tmp/zabbix_traps.tmp`

SocketDir

Especifique el directorio para almacenar los sockets IPC utilizados por los servicios internos de Zabbix.

Predeterminado: `/tmp`

SourceIP

Especifique la dirección IP de origen para:

- conexiones salientes al server de Zabbix
- conexiones sin agent (VMware, SSH, JMX, SNMP, Telnet y comprobaciones simples)
- conexiones de HTTP agent
- solicitudes HTTP de JavaScript del item script
- solicitudes HTTP de JavaScript del preprocesamiento
- conexiones al Vault

SSHKeyLocation

Especifique la ubicación de las claves pública y privada para las comprobaciones y acciones de SSH.

SSLCertLocation

Especifique la ubicación de los archivos de certificados del cliente SSL para la autenticación del cliente.
Este parámetro se utiliza en la monitorización web, para la comunicación con Vault y para la fuente de datos de telemetría ClickHouse.

SSLKeyLocation

Especifique la ubicación de los archivos de clave privada SSL para la autenticación del cliente.
Este parámetro se utiliza en la monitorización web, para la comunicación con Vault y para la fuente de datos de telemetría ClickHouse.

SSLCALocation

Especifique la ubicación de los archivos de la autoridad certificadora (CA) para la verificación del certificado del server SSL.
Tenga en cuenta que el valor de este parámetro se establecerá como la opción `CURLOPT_CAPATH` de libcurl. En las versiones de libcurl anteriores a la 7.42.0, esto solo tiene efecto si libcurl se compiló para utilizar OpenSSL. Para obtener más información, consulte la página web de cURL.
Este parámetro se utiliza en la monitorización web, la autenticación SMTP, los items de HTTP agent, la comunicación con Vault y la fuente de datos de telemetría ClickHouse.

StartAgentPollers

Establece el número de instancias pre-fork de Zabbix agent `pollers`. Consulte `MaxConcurrentChecksPerPoller`.

Predeterminado: 1
 Rango: 0-1000

StartAPMCollector

Establezca 1 para iniciar una instancia pre-forked de un recopilador de APM (los procesos `apm manager` y `apm worker`).

Valor predeterminado: 0
 Rango: 0-1

StartBrowserPollers

Establece el número de instancias pre-forked de browser item `pollers`.

Predeterminado: 1
 Rango: 0-1000

StartDBSyncers

Establece el número de instancias pre-forked de `history syncers`.
Nota: Ten cuidado al cambiar este valor; aumentarlo puede hacer más daño que beneficio.

Predeterminado: 4
 Rango: 1-100

StartDiscoverers

Establece el número de instancias preforkeadas de `discovery workers`.

Predeterminado: 5
 Rango: 0-1000

StartHTTPAgentPollers

Establece el número de instancias pre-forked de `HTTP agent pollers`. Consulte `MaxConcurrentChecksPerPoller`.

Predeterminado: 1
 Rango: 0-1000

StartHTTPPollers

Establece el número de instancias pre-forked de `HTTP pollers`.

Predeterminado: 1
 Rango: 0-1000

StartIPMIPollers

Establece el número de instancias pre-forked de `IPMI pollers`.

Predeterminado: 0
 Rango: 0-1000

StartJavaPollers

Establece el número de instancias preforkeadas de `Java pollers`.

Predeterminado: 0
 Rango: 0-1000

StartODBCPollers

Establece el número de instancias pre-forked de `ODBC pollers`.

Predeterminado: 1
 Rango: 0-1000

StartPingers

Establece el número de instancias pre-forked de `ICMP pingers`.

Predeterminado: 1
 Rango: 0-1000

StartPollersUnreachable

Establece el número de instancias pre-forked de pollers para hosts inalcanzables (incluidos IPMI y Java). Debe estar en ejecución al menos un poller para hosts inalcanzables si se inician pollers normales, IPMI o Java.

Predeterminado: 1
 Rango: 0-1000

StartPollers

Establece el número de instancias pre-forked de `pollers`.

Predeterminado: 5
 Rango: 0-1000

StartPreprocessors

Establezca el número de instancias preiniciadas de los hilos de preprocesamiento `worker` en un valor no inferior al número de núcleos de CPU disponibles. Se deben configurar más workers si el preprocesamiento no está limitado por la CPU y realiza muchas solicitudes de red.

Predeterminado: 16
 Rango: 1-1000

StartSNMPPollers

Establece el número de instancias pre-forked de SNMP pollers. Consulte MaxConcurrentChecksPerPoller.

Predeterminado: 1
 Rango: 0-1000

StartTelemetryQueryPollers

Establece el número de instancias precreadas de pollers de consultas de Telemetry asíncronas. Consulta MaxConcurrentChecksPerPoller.

Valor predeterminado: 1
 Rango: 0-1000

StartSNMPTrapper

Establézcalo en "1" para iniciar un proceso de SNMP trapper.

Predeterminado: 0
 Rango: 0-1

StartTrappers

Establece el número de instancias pre-forked de trappers.
Los trappers aceptan conexiones entrantes de Zabbix sender y de los agents activos.

Predeterminado: 5
 Rango: 0-1000

StartVMwareCollectors

Establece el número de instancias de vmware collector preforkeadas.

Predeterminado: 0
 Rango: 0-250

StatsAllowedIP

Especifique direcciones IP (opcionalmente en notación CIDR) o nombres DNS de instancias externas de Zabbix que tienen □□□□□□ para solicitar estadísticas. Una lista delimitada por comas.
 Si este parámetro no está configurado, no se aceptará ninguna solicitud de estadísticas.
Si el soporte IPv6 está habilitado, entonces `127.0.0.1`, `::127.0.0.1`, `::ffff:127.0.0.1` se tratan por igual y `::/0` permitirá cualquier dirección IPv4 o IPv6. `0.0.0.0/0` puede usarse para permitir cualquier dirección IPv4.

Ejemplo:

```
StatsAllowedIP=127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.example.com
```

TelemetryProvider

Establezca la fuente de datos de telemetría con el formato `<provider>[;<options>]`, donde:

- `<provider>`: una de las fuentes de datos de telemetría compatibles: `clickhouse`
- `<options>`: pares `key=value` delimitados por comas. Los valores pueden estar entre comillas.

Opciones para ClickHouse:

- `url`: establezca la URL HTTP o HTTPS utilizada para conectarse a la fuente de datos de telemetría.
- `db`: establezca el nombre de la base de datos.
- `username`: establezca el nombre de usuario utilizado para iniciar sesión en la base de datos.
- `password`: establezca la contraseña utilizada para iniciar sesión en la base de datos.
- `ssl_cert_file`, `ssl_key_file`, `ssl_key_password`, `ssl_verify_peer`, `ssl_verify_host`, `ssl_ca_location`, `ssl_cert_location`, `ssl_key_location`: establezca estas opciones para configurar una conexión segura (TLS/SSL) a la base de datos.
- `source_ip`: establezca la dirección IP de origen utilizada para las conexiones salientes a la fuente de datos de telemetría.
- `vault_path`: establezca la ruta a un secreto de Vault que contenga los valores de `username` y `password`. No lo utilice junto con las opciones `username` y `password`.

Ejemplo para ClickHouse:

```
TelemetryProvider=clickhouse;url=http://localhost:8123,db=zabbix,username=zabbix,password=zabbix
```

Ejemplo para ClickHouse con Vault:

```
TelemetryProvider=clickhouse;url=http://localhost:8123,db=zabbix,vault_path=clickhouse
```

Timeout

Especifique cuánto tiempo esperar (en segundos) para establecer la conexión e intercambiar datos con el server de Zabbix, el agent, el servicio web y las comprobaciones SNMP heredadas (número OID único o cadena).

Este parámetro define la duración de varias operaciones de comunicación:

- Ejecución de comandos remotos en el agent de Zabbix
- Ejecución de comandos SSH / Telnet
- Reprogramación de items cuando la interfaz IPMI deja de estar disponible
- Envío de la respuesta al server de Zabbix cuando falla el intercambio de datos debido a problemas de permisos o cifrado
- Límite de tiempo para sockets asíncronos de IPC y opciones de control en tiempo de ejecución
- Solicitudes DNS del poller asíncrono
- Respuesta para el heartbeat de comprobación activa
- Recuperación de datos (valores) del agent de Zabbix desde agents activos
- Recuperación de datos desde Zabbix sender
- Envío de la lista de comprobaciones activas al agent de Zabbix

Este timeout **no** se utilizará para aquellas comprobaciones que tengan configurados ajustes de timeout flexible en el frontend (a nivel global, de proxy o por item). Por ejemplo, los items SNMP `walk[OID]` y `get[OID]` usan el timeout configurado en el frontend; las comprobaciones SNMP heredadas siguen usando el valor de timeout del server.

Predeterminado: 3
 Rango: 1-30

TLSAccept

Especifique el nivel de cifrado requerido para las conexiones entrantes desde Zabbix server. Se usa para un proxy pasivo y se ignora en un proxy activo. Se pueden especificar varios valores, separados por comas: - *unencrypted* - acepta conexiones sin cifrado (predeterminado); - *psk* - acepta conexiones con TLS y una clave precompartida (PSK); - *cert* - acepta conexiones con TLS y un certificado.

Obligatorio: sí para un proxy pasivo, si se definen parámetros de certificado TLS o PSK (incluso para una conexión *unencrypted*); en caso contrario, no

TLSCAFile

Especifique la ruta completa del archivo que contiene los certificados de la CA de nivel superior para la verificación del certificado del par, utilizados para las comunicaciones cifradas entre los componentes de Zabbix.

TLSCertFile

Especifique la ruta completa del archivo que contiene el certificado del proxy o la cadena de certificados, utilizado para las comunicaciones cifradas entre los componentes de Zabbix.

TLSCipherAll

Especifique la cadena de prioridad de GnuTLS o la cadena de cifrado de OpenSSL (TLS 1.2). Anule los criterios de selección de la suite de cifrado predeterminada para el cifrado basado en certificados y PSK.

Ejemplo:

```
TLS_AES_256_GCM_SHA384:TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
```

TLSCipherAll13

Especifique la cadena de cifrado para OpenSSL 1.1.1 o posterior en TLS 1.3. Anule los criterios de selección predeterminados de la suite de cifrado para el cifrado basado en certificados y PSK.

Ejemplo para GnuTLS:

```
NONE:+VERS-TLS1.2:+ECDHE-RSA:+RSA:+ECDHE-PSK:+PSK:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL
```

Ejemplo para OpenSSL:

```
EECDH+aRSA+AES128:RSA+aRSA+AES128:kECDHEPSK+AES128:kPSK+AES128
```

TLSCipherCert

Especifique la cadena de prioridad de GnuTLS o la cadena de cifrado de OpenSSL (TLS 1.2). Anule los criterios predeterminados de selección del conjunto de cifrado para el cifrado basado en certificados.

Ejemplo para GnuTLS:

```
NONE:+VERS-TLS1.2:+ECDHE-RSA:+RSA:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL:+COMP-NULL:+SIG
```

Ejemplo para OpenSSL:

```
EECDH+aRSA+AES128:RSA+aRSA+AES128
```

TLSCipherCert13

Especifique la cadena de cifrado para OpenSSL 1.1.1 o una versión posterior en TLS 1.3. Anule los criterios de selección predeterminados de la suite de cifrado para el cifrado basado en certificados.

TLSCipherPSK

Especifique la cadena de prioridad de GnuTLS o la cadena de cifrado de OpenSSL (TLS 1.2). Anule los criterios predeterminados de selección de la suite de cifrado para el cifrado basado en PSK.

Ejemplo para GnuTLS:

```
NONE:+VERS-TLS1.2:+ECDHE-PSK:+PSK:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL:+COMP-NULL:+SIG
```

Ejemplo para OpenSSL:

```
kECDHEPSK+AES128:kPSK+AES128
```

TLSCipherPSK13

Especifique la cadena de cifrado para OpenSSL 1.1.1 o una versión posterior en TLS 1.3. Anule los criterios de selección predeterminados del conjunto de cifrado para el cifrado basado en PSK.

Ejemplo:

```
TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
```

TLSConnect

Especifique el nivel de cifrado requerido para las conexiones del proxy con el server de Zabbix. Se usa para un proxy activo y se ignora en un proxy pasivo. Solo se puede especificar un valor: - *unencrypted* - conectar sin cifrado (predeterminado); - *psk* - conectar usando TLS y una clave precompartida (PSK); - *cert* - conectar usando TLS y un certificado.

Obligatorio: sí para un proxy activo, si se definen parámetros de certificado TLS o PSK (incluso para una conexión *unencrypted*); en caso contrario, no

TLSCRLFile

Especifique la ruta completa del archivo que contiene los certificados revocados. Este parámetro se utiliza para las comunicaciones cifradas entre los componentes de Zabbix.

TLSKeyFile

Especifique la ruta completa del archivo que contiene la clave privada del proxy, utilizada para las comunicaciones cifradas entre los componentes de Zabbix.

TLSListen

Requiere conexiones solo TLS en el socket del trapper.

Valores admitidos:

- required - aceptar solo conexiones TLS

TLSPSKFile

Especifique la ruta completa del archivo que contiene la clave precompartida del proxy, utilizada para las comunicaciones cifradas con el server de Zabbix.

TLSPSKIdentity

Especifique la cadena de identidad de la clave previamente compartida, utilizada para comunicaciones cifradas con Zabbix server.

TLSServerCertIssuer

Especifique el emisor permitido del certificado del server.

TLSServerCertSubject

Especifique el subject del certificado de servidor permitido.

TmpDir

Especifique el directorio temporal.

Predeterminado: `/tmp`

TrapperTimeout

Especifique el tiempo de espera en segundos para:
 - la obtención de datos de configuración del server Zabbix;
 - la ejecución de scripts globales o la ejecución de comandos remotos.

Predeterminado: 300
 Rango: 1-300

UnavailableDelay

Establece la frecuencia de comprobación de la disponibilidad del host durante el período de indisponibilidad en segundos.

Predeterminado: 60
 Rango: 1-3600

UnreachableDelay

Establece la frecuencia de comprobación de la disponibilidad del host durante el período de inaccesibilidad en segundos.

Predeterminado: 15
 Rango: 1-3600

UnreachablePeriod

Establece después de cuántos segundos de inaccesibilidad tratar al host como no disponible.

Predeterminado: 45
 Rango: 1-3600

Usuario

Reduce los privilegios a un usuario específico y existente en el sistema.
Solo tiene efecto si se ejecuta como `root` y `AllowRoot` está deshabilitado.

Predeterminado: `zabbix`

Vault

Especifique el proveedor de vault: - *HashiCorp* - HashiCorp KV Secrets Engine versión 2; - *CyberArk* - CyberArk Central Credential Provider
Debe coincidir con el proveedor de vault configurado en el frontend.

Predeterminado: `HashiCorp`

VaultAppRoleID

Especifique el AppRole ID de HashiCorp vault.
 Si se establece, habilita la autenticación AppRole. No se puede usar con VaultToken.
 Solo es compatible con HashiCorp vault.

VaultAppSecretID

Especifique el secret ID de AppRole de HashiCorp Vault.
 Solo es compatible con HashiCorp Vault.

VaultDBPath

Especifique la ubicación para recuperar las credenciales de la base de datos mediante claves. Puede ser una ruta o una consulta de Vault, según el Vault.

Las claves utilizadas por HashiCorp son `password` y `username`.

Ejemplo de ruta con `VaultPrefix=/v1/secret/data/zabbix/`:

```
database
```

Ejemplo de ruta sin VaultPrefix:

```
secret/zabbix/database
```

Las claves utilizadas por CyberArk son `Content` y `UserName`.

Ejemplo:

```
AppID=zabbix_server&Query=Safe=passwordSafe;Object=zabbix_proxy_database
```

Esta opción solo puede utilizarse si `DBUser` y `DBPassword` no están especificados.

VaultPrefix

Especifique un prefijo personalizado para la ruta o consulta de Vault, según el Vault. Se usarán los valores predeterminados más adecuados si no se especifica.
Tenga en cuenta que 'data' se añade automáticamente después del punto de montaje para HashiCorp si `VaultPrefix` no se especifica.
Tenga en cuenta que 'data' se añade automáticamente después del punto de montaje para HashiCorp si `VaultPrefix` no se especifica.

Ejemplo de prefijo para HashiCorp:

```
/v1/secret/data/zabbix/
```

Ejemplo de prefijo para Cyberark:

```
/AIMWebService/api/Accounts?
```

VaultTLSCertFile

Especifique el nombre del archivo de certificado SSL utilizado para la autenticación del cliente. El archivo de certificado debe estar en formato PEM1.
Si el archivo de certificado también contiene la clave privada, deje vacío el campo del archivo de clave SSL.
El directorio que contiene este archivo se especifica mediante el parámetro de configuración SSLCertLocation.
Esta opción se puede omitir, pero se recomienda para CyberArkCCP vault.

VaultTLSKeyFile

Especifique el nombre del archivo de clave privada SSL utilizado para la autenticación del cliente. El archivo de clave privada debe estar en formato PEM1.
El directorio que contiene este archivo se especifica mediante el parámetro de configuración SSLKeyLocation.
Esta opción puede omitirse, pero se recomienda para el vault CyberArkCCP.

VaultToken

Especifique el token de autenticación de HashiCorp Vault que debe haberse generado exclusivamente para el proxy de Zabbix con permiso de solo lectura para la ruta especificada en el parámetro de configuración opcional VaultDBPath.
Es un error si `VaultToken` y la variable de entorno `VAULT_TOKEN` están definidos al mismo tiempo.
 No se puede usar con VaultAppRoleID.
 Compatible solo con HashiCorp Vault.

VaultURL

Especifique la URL del servidor de vault. Se utilizará el directorio de certificados CA de todo el sistema si no se especifica SSLCALocation.

Predeterminado: `https://127.0.0.1:8200`

VMwareCacheSize

Establezca el tamaño de la memoria compartida para almacenar datos de VMware.
Se puede usar una comprobación interna de VMware `zabbix[vmware,buffer,...]` para supervisar el uso de la caché de VMware (consulte Internal checks).
Tenga en cuenta que la memoria compartida no se asigna si no hay instancias de recopilador de VMware configuradas para iniciarse.

Predeterminado: 8M
 Rango: 256K-2G

VMwareFrequency

Establezca la frecuencia de recopilación de datos de un único servicio VMware (en segundos).
Esta frecuencia debe configurarse según el intervalo de actualización más corto de cualquier item de monitoreo de VMware.

Predeterminado: 60
 Rango: 10-86400

VMwarePerfFrequency

Establece la frecuencia de recuperación de las estadísticas de contadores de rendimiento de un único servicio VMware (en segundos).
Esta frecuencia debe configurarse con el intervalo de actualización más corto de cualquier item de supervisión de VMware que utilice contadores de rendimiento de VMware.

Predeterminado: 60
 Rango: 10-86400

VMwareTimeout

Establece el número máximo de segundos que un `vmware collector` esperará una respuesta del servicio VMware (vCenter o hipervisor ESX).

Predeterminado: 10
 Rango: 1-300

WebDriverURL

Especifique la URL de la interfaz de WebDriver.

Ejemplo (usado con Selenium WebDriver standalone server):

```
WebDriverURL=http://localhost:4444
```

agent

Descripción general El agent de Zabbix se implementa en un objetivo de monitorización para supervisar activamente los recursos y las aplicaciones locales (discos duros, memoria, estadísticas del procesador, etc.).

El agent recopila información operativa localmente y envía los datos al server de Zabbix para su posterior procesamiento.
En caso de fallos (como un disco duro lleno o un proceso de servicio bloqueado), el server de Zabbix puede alertar activamente a los administradores de la máquina específica que informó del fallo.

Los agents de Zabbix son muy eficientes porque utilizan llamadas nativas del sistema para recopilar información estadística.

Es posible ejecutar varias instancias del agent en un host.

Véase también:

- Protocolos de comprobaciones pasivas y activas
- Ejecución del agent como root
- Parámetros de configuración: Zabbix agent
- Parámetros de configuración: Zabbix agent (Windows)
- Zabbix agent en Microsoft Windows
- Zabbix agent en IBM AIX

Comprobaciones pasivas y activas

Los agentes de Zabbix pueden realizar comprobaciones pasivas y activas:

- Comprobaciones pasivas - El agent de Zabbix responde a una solicitud de Zabbix server (o proxy). Por ejemplo, el server solicita datos (p. ej., carga de CPU), y el agent devuelve el resultado.
- Comprobaciones activas - El agent de Zabbix recopila y envía datos sin esperar una solicitud de Zabbix server (o proxy). Primero, obtiene una lista de items de monitoreo del server (carga de CPU, memoria disponible, etc.), luego recopila los datos requeridos y envía periódicamente nuevos valores de vuelta a este.

El tipo de comprobación del agent se configura seleccionando el correspondiente tipo de item de monitoreo. El agent de Zabbix procesa items de tipo *Zabbix agent* o *Zabbix agent (active)*.

Plataformas compatibles Para las plataformas compatibles, consulte la página Requisitos.

Agent en sistemas tipo Unix Zabbix agent en sistemas tipo Unix se ejecuta en el host que se está supervisando.

Instalación

Zabbix agent puede instalarse en sistemas basados en Linux mediante uno de los siguientes métodos:

- Paquetes de Zabbix: seleccione el componente Agent (después de seleccionar la versión de Zabbix, la distribución del sistema operativo y la versión del sistema operativo) y siga las instrucciones proporcionadas;
- Fuentes de Zabbix: descargue los archivos fuente y compile Zabbix agent configurando las fuentes con la opción `--enable-agent`.

Attention:
En general, los Zabbix agents de 32 bits funcionarán en sistemas de 64 bits, pero en algunos casos pueden producirse errores.

Note:
Hay binarios de Zabbix agent precompilados disponibles para descargar para macOS, IBM AIX, FreeBSD, OpenBSD y Solaris. Hay binarios heredados, compatibles con la versión actual de Zabbix server/proxy, disponibles para NetBSD y HP-UX. Para instalar y ejecutar el agent en IBM AIX, consulte Zabbix agent en IBM AIX.

Si se instala como paquete

El agent de Zabbix se ejecuta como un proceso demonio. El agent puede iniciarse ejecutando:

```
systemctl start zabbix-agent
```

Esto funcionará en la mayoría de los sistemas GNU/Linux. En otros sistemas puede que necesite ejecutar:

```
/etc/init.d/zabbix-agent start
```

Para detener, reiniciar o comprobar el estado del agent de Zabbix, utilice los siguientes comandos:

```
systemctl stop zabbix-agent
systemctl restart zabbix-agent
systemctl status zabbix-agent
```

Si se ha instalado desde un archivo binario

Después de extraer el archivo, localice el binario `zabbix_agentd` y el archivo de configuración `zabbix_agentd.conf`. Su ubicación dentro del archivo depende de la plataforma (por ejemplo, `sbin/` y `conf/`, o `usr/sbin/` y `etc/zabbix/`). Ejecute el binario directamente desde el directorio extraído y especifique el archivo de configuración con la opción `-c`. Por ejemplo:

```
/opt/zabbix/sbin/zabbix_agentd -c /opt/zabbix/etc/zabbix/zabbix_agentd.conf
```

En sistemas similares a UNIX, Zabbix agent está diseñado para ejecutarse como un usuario que no sea root. Si se inicia como root, cambia al usuario `zabbix` (o al usuario especificado por el parámetro [User](<PARAMS>#user)), a menos que [AllowRoot=1](<PARAMS>#allowroot) esté establecido en el archivo de configuración. Este usuario debe existir en el sistema; de lo contrario, el agent no se inicia.

En IBM AIX, el agent también puede ejecutarse con el System Resource Controller (SRC) e iniciarse durante el arranque del sistema; consulte Zabbix agent en IBM AIX.

Iniciar manualmente

Puede iniciar el agent de Zabbix ejecutando directamente el binario `zabbix_agentd`, por ejemplo:

```
zabbix_agentd
```

Esto funciona cuando el directorio que contiene `zabbix_agentd` está incluido en `PATH`.

Agent en sistemas Windows Zabbix agent en Windows se ejecuta como un servicio de Windows.

Instalación

Zabbix agent se puede instalar en Windows mediante uno de los siguientes métodos:

- Binarios precompilados de Zabbix agent - descargue el paquete instalador MSI de Zabbix agent y siga las instrucciones de la página Instalación de Zabbix agent en Windows desde MSI;
- Fuentes de Zabbix - descargue los archivos fuente y siga las instrucciones de la página Compilación de Zabbix agent en Windows.

Consulte la página Zabbix agent en Microsoft Windows para obtener detalles adicionales sobre la instalación de Zabbix agent (desde un archivo ZIP) como servicio de Windows.

Opciones Los siguientes parámetros de línea de comandos se pueden utilizar con Zabbix agent:

Parámetro	Descripción
agent UNIX y Windows	
`-c --config <config-file>`	Ruta al archivo de configuración. Puede utilizar esta opción para especificar un archivo de configuración que no sea el predeterminado.
`-f --foreground`	Ejecutar Zabbix agent en primer plano (valor predeterminado: `true`).
`-p --print`	Mostrar los items conocidos y salir. Tenga en cuenta que, para devolver también los resultados de los parámetros de usuario, debe especificar el archivo de configuración (si no se encuentra en la ubicación predeterminada).
`-t --test <item key>`	Probar el item especificado y salir. Tenga en cuenta que, para devolver también los resultados de los parámetros de usuario, debe especificar el archivo de configuración (si no se encuentra en la ubicación predeterminada).
`-T --test-config`	Validar el archivo de configuración y salir.
`-h --help`	Mostrar información de ayuda.
`-V --version`	Mostrar el número de versión.
Solo agent UNIX	
`-R --runtime-control <option>`	Realizar funciones administrativas. Consulte control en tiempo de ejecución.
Solo agent Windows	
`-m --multiple-agents`	Utilizar varias instancias del agent (con las opciones `-i`, `-d`, `-s`, `-x`). Para distinguir los nombres de servicio de las instancias, cada nombre de servicio incluirá el valor de Hostname del archivo de configuración especificado.

Parámetro	Descripción
`-S --startup-type <value>`	Establecer el tipo de inicio del servicio de Zabbix Windows agent. Valores permitidos: `automatic` - *(predeterminado)* iniciar el servicio automáticamente al iniciar Windows; `delayed` - retrasar el inicio del servicio hasta que los servicios iniciados automáticamente hayan completado su inicio (disponible en Windows Server 2008/Vista y versiones posteriores); `manual` - iniciar el servicio manualmente (por un usuario o una aplicación); `disabled` - deshabilitar el servicio para que no pueda ser iniciado por un usuario o una aplicación. Puede utilizar esta opción junto con la opción `-i` o por separado para modificar el tipo de inicio de un servicio ya instalado.
`-i --install`	Instalar Zabbix Windows agent como servicio.
`-d --uninstall`	Desinstalar el servicio de Zabbix Windows agent.
`-s --start`	Iniciar el servicio de Zabbix Windows agent.
`-x --stop`	Detener el servicio de Zabbix Windows agent.

Ejemplos específicos del uso de parámetros de línea de comandos:

- Mostrar todos los items integrados del agent con sus valores.
- Probar un parámetro de usuario con la clave `mysql.ping` definida en el archivo de configuración especificado.
- Instalar un servicio de Zabbix Agent para Windows utilizando la ruta predeterminada al archivo de configuración.
- Instalar un servicio de Zabbix Agent [Hostname] para Windows utilizando el archivo de configuración ubicado en la misma carpeta que el ejecutable del agent y hacer que el nombre del servicio sea único ampliándolo con el valor de `Hostname` del archivo de configuración.
- Modificar el tipo de inicio de un servicio de Zabbix Agent para Windows instalado utilizando el archivo de configuración ubicado en la misma carpeta que el ejecutable del agent.

```
zabbix_agentd --print
zabbix_agentd -t "mysql.ping" -c /etc/zabbix/zabbix_agentd.conf
zabbix_agentd.exe -i
zabbix_agentd.exe -i -m -c zabbix_agentd.conf
zabbix_agentd.exe -c zabbix_agentd.conf -S delayed
```

Control en tiempo de ejecución

Con las opciones de control en tiempo de ejecución puede cambiar el nivel de registro de los procesos del agent.

Opción	Descripción	Destino
`log_level_increase [=<target>]`	Aumentar el nivel de registro. Si no se especifica el destino, todos los procesos se ven afectados.	El destino se puede especificar como: **tipo de proceso**: todos los procesos del tipo especificado (por ejemplo, `listener`) Consulte todos los tipos de procesos del agent. **tipo de proceso,N**: tipo y número de proceso (por ejemplo, `listener,3`) **pid**: identificador del proceso (1 a 65535). Para valores mayores, especifique el destino como «tipo de proceso,N».
`log_level_decrease [=<target>]`	Reducir el nivel de registro. Si no se especifica el destino, todos los procesos se ven afectados.	
`userparameter_reload`	Volver a cargar los valores de las opciones `UserParameter` e `Include` del archivo de configuración actual.	

Ejemplos:

- Aumentar el nivel de registro de todos los procesos
- Aumentar el nivel de registro del tercer proceso `listener`
- Aumentar el nivel de registro del proceso con PID 1234
- Reducir el nivel de registro de todos los procesos de comprobación activa

```
zabbix_agentd -R log_level_increase
zabbix_agentd -R log_level_increase=listener,3
```

```
zabbix_agentd -R log_level_increase=1234
zabbix_agentd -R log_level_decrease="active checks"
```

> **Note:**
> El control en tiempo de ejecución no es compatible con OpenBSD, NetBSD ni Windows.

Tipos de procesos del agent Los tipos de procesos del agent son los siguientes:

- `active checks` - proceso para realizar active checks;
- `collector` - proceso para la recopilación de datos;
- `listener` - proceso para escuchar passive checks.

El archivo de registro del agent puede utilizarse para observar estos tipos de procesos.

El archivo de registro del agent se crea con permisos de lectura y escritura solo para el propietario del archivo. Además, el archivo es legible por el grupo propietario. Todos los demás permisos están denegados.

Ejecutar como usuario Zabbix agent en UNIX está diseñado para ejecutarse como un usuario que no sea root. Se ejecutará con el usuario que no sea root con el que se inicie. Por lo tanto, puede ejecutar el agent con cualquier usuario que no sea root sin problemas.

Si intenta ejecutarlo como «root», cambiará al usuario «zabbix» codificado de forma fija, que debe estar presente en su sistema. Solo puede ejecutar el agent como «root» si modifica adecuadamente el parámetro `AllowRoot` en el archivo de configuración del agent.

Archivo de configuración La ruta predeterminada al archivo de configuración depende del sistema operativo y del método de instalación de Zabbix:

- En UNIX, la ruta predeterminada es:
  - Para la instalación desde las fuentes: `/usr/local/etc/zabbix_agentd.conf`, o la establecida mediante las variables de compilación `--sysconfdir` o `--prefix`.
  - Para la instalación desde paquetes y el appliance: `/etc/zabbix/zabbix_agentd.conf`.
- En Windows, la ruta predeterminada es `C:\Program Files\Zabbix Agent\zabbix_agentd.conf`.

Si se está ejecutando una única instancia del agent en un host, puede utilizar el archivo de configuración predeterminado o un archivo de configuración especificado en la línea de comandos. En caso de varias instancias, cada instancia del agent debe tener su propio archivo de configuración (una de las instancias puede utilizar el archivo de configuración predeterminado).

Para obtener más información sobre la configuración del agent de Zabbix, consulte las opciones del archivo de configuración del agent de Zabbix para Unix o del agent de Zabbix para Windows.

Configuración regional Tenga en cuenta que el agent requiere una configuración regional UTF-8 para que algunos items textuales del agent puedan devolver el contenido esperado. La mayoría de los sistemas Unix modernos tienen una configuración regional UTF-8 como predeterminada; sin embargo, hay algunos sistemas en los que puede ser necesario configurarla específicamente.

Código de salida Zabbix agent devuelve 0 en caso de salida correcta y 1 en caso de fallo.

Protocolos de comprobación pasiva y activa

Esta sección proporciona detalles del protocolo sobre las comprobaciones pasivas y activas realizadas por Zabbix agent y Zabbix agent 2.

Zabbix utiliza un protocolo de comunicación basado en JSON para comunicarse con los agents.

Los protocolos de Zabbix agent y Zabbix agent 2 se han unificado desde Zabbix 7.0. La diferencia entre las solicitudes/respuestas de Zabbix agent y Zabbix agent 2 se expresa mediante el valor de la etiqueta ”variant”.

Comprobaciones pasivas

Una comprobación pasiva es una simple solicitud de datos. El server o proxy de Zabbix solicita algunos datos (por ejemplo, la carga de CPU) y el agent de Zabbix envía el resultado al server.

Las comprobaciones pasivas se ejecutan de forma asíncrona: no es necesario recibir la respuesta a una solicitud antes de iniciar otras comprobaciones. La resolución DNS también es asíncrona.

El agente de sondeo intentará conectarse a todas las direcciones devueltas por la búsqueda DNS. Esto garantiza que, si una dirección IP no está disponible, el sondeador intentará la siguiente dirección disponible, lo que aumenta la probabilidad de una conexión correcta. Esta mejora se aplica tanto al server como al proxy de Zabbix.

La concurrencia máxima de las comprobaciones asíncronas es 1000 (definida por MaxConcurrentChecksPerPoller).

El número de sondeadores asíncronos de agent se define mediante el parámetro StartAgentPollers.

Solicitud del server

Para la definición de la cabecera y la longitud de los datos, consulte los detalles del protocolo.

```
{
  "request": "passive checks",
  "data": [
    {
      "key": "agent.version",
      "timeout": 3
    }
  ]
}
```

Field		Type	Mandatory	Value
request		*string*	yes	`"passive checks"`
data		*array of object*	yes	Passive check item.
	key	*string*	yes	Item key with expanded macros.
	timeout	*number*	yes	Communication timeout.

Respuesta del agent

```
{
  "version": "8.0.0",
  "variant": 2,
  "data": [
    {
      "value": "8.0.0"
    }
  ]
}
```

Field		Type	Mandatory	Value
version		*string*	yes	The agent version number.
variant		*number*	yes	The agent variant (*1* - Zabbix agent, *2* - Zabbix agent 2).
data		*array of object*	yes	Contains the result of the check.
	value	*string*	no	The item value if the check was successful.
	error	*string*	no	The error message if the check was not successful.

Por ejemplo, para items compatibles:

1. El server abre una conexión TCP
2. El server envía **<HEADER><DATALEN>{"request":"passive checks","data":[{"key":"agent.ping","timeout":3}]}**
3. El agent lee la solicitud y responde con **<HEADER><DATALEN>{"version":"8.0.0","variant":2,"data":[{"value":1}]}**
4. El server procesa los datos para obtener el valor, '1' en nuestro caso
5. La conexión TCP se cierra

Para items no compatibles:

1. El server abre una conexión TCP
2. El server envía **<HEADER><DATALEN>{"request":"passive checks","data":[{"key":"vfs.fs.size[/nono]","timeout":3}]}**
3. El agent lee la solicitud y responde con **<HEADER><DATALEN>{"version":"8.0.0","variant":2,"data":[{"error":"Unsupported item key."}]}**
4. El server procesa los datos, cambia el estado del item a no compatible con el mensaje de error especificado
5. La conexión TCP se cierra

Conmutación al protocolo antiguo

Para garantizar que Zabbix server o proxy puedan funcionar con agents de versiones anteriores a la 7.2, que utilizan el protocolo de texto plano, se implementa una conmutación al protocolo antiguo.

Las comprobaciones pasivas se realizan usando el protocolo JSON (7.0 y posteriores) después de reiniciar o cuando se cambia la configuración de la interfaz.
Si no se recibe una respuesta JSON válida (el agent envió "ZBX_NOTSUPPORTED"), Zabbix almacenará en caché la interfaz como protocolo antiguo y **reintentará** la comprobación enviando solo la clave del item.

Tenga en cuenta que cada hora Zabbix server/proxy volverá a intentar trabajar con el nuevo protocolo con todas las interfaces, y recurrirá al protocolo antiguo si es necesario.

Comprobaciones activas

Las comprobaciones activas requieren un procesamiento más complejo. Primero, el agent debe obtener del server/proxy una lista de items y/o comandos remotos para procesarlos de forma independiente.

Los servers/proxies de los que se obtienen las comprobaciones activas se indican en el parámetro `ServerActive` del archivo de configuración del agent. La frecuencia con la que se solicitan estas comprobaciones se establece mediante el parámetro `RefreshActiveChecks` en el mismo archivo de configuración. Sin embargo, si se produce un error al actualizar las comprobaciones activas, se vuelve a intentar después de 60 segundos, un intervalo definido de forma fija.

Note:
Desde Zabbix 6.4, el agent (en modo activo) ya no recibe del server/proxy una copia completa de la configuración una vez cada dos minutos (valor predeterminado). En su lugar, para reducir el tráfico de red y el uso de recursos, se realiza una sincronización incremental de la configuración cada 5 segundos (valor predeterminado), tras la cual el server/proxy proporciona una copia completa de la configuración **solo** si el agent aún no la ha recibido, o si algo ha cambiado en la configuración del host, en las macros globales o en las expresiones regulares globales.

El agent luego envía periódicamente los nuevos valores al server(s). Si el agent recibió algún remote commands para ejecutar, el resultado de la ejecución también se enviará. Tenga en cuenta que la ejecución de remote command en un agent activo es compatible desde Zabbix agent 7.0.

Note:
Si un agent está detrás del firewall, podría considerar usar solo Active checks porque en este caso no necesitaría modificar el firewall para permitir las conexiones entrantes iniciales.

Obtención de la lista de items

Solicitud del agent

La solicitud de active checks se utiliza para obtener los active checks que deben ser procesados por el agent.
Esta solicitud la envía el agent al iniciarse y luego en intervalos de RefreshActiveChecks.

```
{
  "request": "active checks",
  "host": "Zabbix server",
  "host_metadata": "mysql,nginx",
  "interface": "zabbix.server.lan",
  "ip": "159.168.1.1",
  "port": 12050,
  "version": "8.0.0",
  "variant": 2,
  "config_revision": 1,
  "session": "e3dcbd9ace2c9694e1d7bbd030eeef6e"
}
```

Field	Type	Mandatory	Value
request	*string*	yes	`active checks`
host	*string*	yes	Nombre del host.
host_metadata	*string*	no	El parámetro de configuración HostMetadata o el valor de la métrica HostMetadataItem.
interface	*string*	no	El parámetro de configuración HostInterface o el valor de la métrica HostInterfaceItem.
ip	*string*	no	La primera IP del parámetro de configuración ListenIP, si está definido.

Field	Type	Mandatory	Value
port	*number*	no	El valor del parámetro de configuración ListenPort, si está definido y no es el puerto de escucha predeterminado del agent.
version	*string*	yes	El número de versión del agent.
variant	*number*	yes	La variante del agent (*1* - Zabbix agent, *2* - Zabbix agent 2).
config_revision	*number*	no	Identificador de configuración para sincronización incremental de configuración.
session	*string*	no	Identificador de sesión para sincronización incremental de configuración.

Respuesta del server

La respuesta de active checks la envía el server de vuelta al agent después de procesar la solicitud de active checks.

```
{
  "response": "success",
  "config_revision": 2,
  "data": [
    {
      "key": "system.uptime",
      "itemid": 1234,
      "delay": "10s",
      "lastlogsize": 0,
      "mtime": 0
    },
    {
      "key": "agent.version",
      "itemid": 5678,
      "delay": "10m",
      "lastlogsize": 0,
      "mtime": 0,
      "timeout": "30s"
    }
  ],
  "commands": [
    {
      "command": "df -h --output=source,size / | awk 'NR>1 {print $2}'",
      "id": 1324,
      "wait": 1
    }
  ]
}
```

Field	Type	Mandatory	Value
response	*string*	yes	`success` | `failed`
info	*string*	no	Información de error en caso de fallo.
data	*array of objects*	no	Items de active check. Se omite si la configuración del host no ha cambiado.
key	*string*	no	Clave del item con macros expandidas.
itemid	*number*	no	Identificador del item.
delay	*string*	no	Intervalo de actualización del item. Los intervalos flexibles/de programación son compatibles tanto con Zabbix agent como con Zabbix agent 2 desde Zabbix 7.0.
lastlogsize	*number*	no	lastlogsize del item.
mtime	*number*	no	mtime del item.
timeout	*string*	no	Tiempo de espera del item.
refresh_unsupported	*number*	no	Intervalo de actualización de items no compatibles.
regexp	*array of objects*	no	Expresiones regulares globales.
name	*string*	no	Nombre de la expresión regular global.
expression	*string*	no	Expresión regular global.
expression_type	*number*	no	Tipo de expresión regular global.

Field	Type	Mandatory	Value
exp_delimiter	*string*	no	Delimitador de la expresión regular global.
case_sensitive	*number*	no	Configuración de distinción entre mayúsculas y minúsculas de la expresión regular global.
commands	*array of objects*	no	Comandos remotos a ejecutar. Se incluyen si la ejecución de comandos remotos ha sido activada por una operación de acción o por la ejecución manual de un script. Tenga en cuenta que la ejecución de comandos remotos en un active agent es compatible desde Zabbix agent 7.0. Los active agents anteriores ignorarán cualquier comando remoto incluido en la respuesta del server de active checks.
command	*string*	no	Comando remoto.
id	*number*	no	Identificador del comando remoto.
wait	*number*	no	Modo de ejecución del comando remoto (”0” (nowait) para comandos de operaciones de acción; ”1” (wait) para comandos de ejecución manual de script).
timeout	*number*	no	Tiempo de espera de ejecución del comando remoto en la configuración de server/proxy.
config_revision	*number*	no	Identificador de configuración para sincronización incremental de configuración. Se omite si la configuración del host no ha cambiado. Se incrementa si la configuración del host cambia.

El server debe responder con success.

Por ejemplo:

1. El agent abre una conexión TCP
2. El agent solicita la lista de checks
3. El server responde con una lista de items y comandos remotos a ejecutar
4. El agent analiza la respuesta
5. La conexión TCP se cierra
6. El agent inicia la recopilación periódica de datos y ejecuta comandos remotos (compatible desde Zabbix agent 7.0)

Attention:
Tenga en cuenta que los datos de configuración (sensibles) pueden quedar disponibles para terceros que tengan acceso al puerto trapper del Zabbix server al usar un active check. Esto es posible porque cualquiera puede hacerse pasar por un active agent y solicitar datos de configuración de items; la autenticación no tiene lugar a menos que use opciones de encryption.

Envío de datos recopilados

Agent envía

La solicitud de datos del agent contiene los valores de item recopilados y los valores de los comandos remotos ejecutados (si los hay).

```
{
  "request": "agent data",
  "data": [
    {
      "id": 1,
      "itemid": 5678,
      "value": "7.0.0",
      "clock": 1712830783,
      "ns": 76808644
    },
    {
      "id": 2,
      "itemid": 1234,
      "value": "69672",
      "clock": 1712830783,
      "ns": 77053975
    }
  ],
  "commands": [
```

```
    {
      "id": 1324,
      "value": "16G"
    }
  ],
  "session": "8495cd52070e6ca52b371f29c8574165",
  "host": "Zabbix server",
  "version": "8.0.0",
  "variant": 2
}
```

Field	Type	Mandatory	Value
request	*string*	yes	`agent data`
data	*array of objects*	yes	Valores de item.
id	*number*	yes	El identificador del valor (contador incremental usado para comprobar valores duplicados en caso de problemas de red).
itemid	*number*	yes	El identificador del item.
value	*string*	no	El valor del item.
lastlogsize	*number*	no	El lastlogsize del item.
mtime	*number*	no	El mtime del item.
state	*number*	no	El estado del item.
source	*string*	no	El origen del registro de eventos del valor.
eventid	*number*	no	El eventid del registro de eventos del valor.
severity	*number*	no	La severidad del registro de eventos del valor.
timestamp	*number*	no	La marca de tiempo del registro de eventos del valor.
clock	*number*	yes	La marca de tiempo del valor (segundos desde Epoch).
ns	*number*	yes	Los nanosegundos de la marca de tiempo del valor.
commands	*array of objects*	no	Resultado de la ejecución de comandos remotos. Tenga en cuenta que la ejecución de comandos remotos en un agent activo es compatible desde Zabbix agent 7.0. Los agent activos anteriores ignorarán cualquier comando remoto incluido en la respuesta del server a las comprobaciones activas.
id	*number*	no	Identificador del comando remoto.
value	*string*	no	Resultado de la ejecución del comando remoto si la ejecución fue correcta.
error	*string*	no	Mensaje de error de la ejecución del comando remoto si la ejecución falló.
session	*string*	yes	Identificador único de sesión generado cada vez que se inicia el agent (32 dígitos hexadecimales).
host	*string*	yes	Nombre del host.
version	*string*	yes	El número de versión del agent.
variant	*number*	yes	La variante del agent (*1* - Zabbix agent, *2* - Zabbix agent 2).

A cada valor se le asigna un ID virtual. El ID de valor es un contador simple ascendente, único dentro de una sesión de datos (identificada por el token de sesión). Este ID se usa para descartar valores duplicados que podrían enviarse en entornos con conectividad deficiente.

Respuesta del server

La respuesta de datos del agent es enviada por el server de vuelta al agent después de procesar la solicitud de datos del agent.

```
{
  "response": "success",
  "info": "processed: 2; failed: 0; total: 2; seconds spent: 0.003534"
}
```

Field	Type	Mandatory	Value
response	*string*	yes	`success` | `failed`
info	*string*	yes	Resultados del procesamiento de item.

Attention:
Si el envío de algunos valores falla en el server (por ejemplo, porque el host o el item han sido deshabilitados o eliminados), el agent no reintentará el envío de esos valores.

Por ejemplo:

1. El agent abre una conexión TCP
2. El agent envía una lista de valores
3. El server procesa los datos y devuelve el estado
4. La conexión TCP se cierra

Attention:
El mensaje de error se truncará a 2048 símbolos en el lado del server.

Mensaje de latido

El agent envía

El mensaje de latido se envía desde un agent activo al server/proxy de Zabbix cada `HeartbeatFrequency` segundos (configurado en el archivo de configuración de Zabbix agent o Zabbix agent 2).

Se utiliza para supervisar la disponibilidad de las comprobaciones activas.

```
{
  "request": "active check heartbeat",
  "host": "Zabbix server",
  "heartbeat_freq": 60,
  "version": "8.0.0",
  "variant": 2
}
```

Campo	Tipo	Obligatorio	Valor
request	*string*	sí	`active check heartbeat`
host	*string*	sí	El nombre del host.
heartbeat_freq	*number*	sí	La frecuencia de latido del agent (parámetro de configuración HeartbeatFrequency).
version	*string*	sí	El número de versión del agent.
variant	*number*	sí	La variante del agent (*1* - Zabbix agent, *2* - Zabbix agent 2).

Respuesta de redirección

Cuando un host se ha reasignado, el server puede indicar al agent que redirija su latido (y las comprobaciones activas posteriores) a otra instancia de proxy o server.

```
{
  "response": "failed",
  "redirect": {
    "revision": 2,
    "address": "192.0.2.1:10055"
  }
}
```

Campo		Tipo	Obligatorio	Valor
response		*string*	sí	`success` | `failed`
redirect		*object*	sí	Instrucciones de redirección.
	revision	*number*	sí	Identificador de revisión de la configuración.
	address	*string*	sí	Dirección del server/proxy de destino.

Protocolo XML antiguo

Note:
Zabbix aceptará hasta 16 MB de datos XML codificados en Base64, pero un único valor decodificado no debe superar los 64 KB; de lo contrario, se truncará a 64 KB durante la decodificación.

Ejecutar agent como root

Desde Zabbix **5.0.0**, el archivo de servicio systemd para Zabbix agent en los paquetes oficiales incluye explícitamente directivas para `User` y `Group`. Ambas están configuradas como `zabbix`.

Ya no es posible configurar con qué usuario se ejecuta Zabbix agent mediante el archivo `zabbix_agentd.conf`, porque el agent omitirá esta configuración y se ejecutará como el usuario especificado en el archivo de servicio systemd. Para ejecutar Zabbix agent como root, debe realizar las modificaciones descritas a continuación.

Zabbix agent

Para anular el usuario y el grupo predeterminados de Zabbix agent, ejecute:

```
systemctl edit zabbix-agent
```

A continuación, añada el siguiente contenido:

```
[Service]
User=root
Group=root
```

Recargue los daemons y reinicie el servicio zabbix-agent:

```
systemctl daemon-reload
systemctl restart zabbix-agent
```

Para **Zabbix agent**, esto vuelve a habilitar la funcionalidad de configurar el usuario en el archivo de configuración del agent. Ahora debe establecer los parámetros de configuración `User=root` y `AllowRoot=1` en el archivo de configuración.

Zabbix agent 2

Para sobrescribir el usuario y el grupo predeterminados de Zabbix agent 2, ejecute:

```
systemctl edit zabbix-agent2
```

Luego, agregue el siguiente contenido:

```
[Service]
User=root
Group=root
```

Recargue los daemons y reinicie el servicio zabbix-agent2:

```
systemctl daemon-reload
systemctl restart zabbix-agent2
```

Para **Zabbix agent2**, esto determina por completo el usuario con el que se ejecuta. No se requieren modificaciones adicionales.

Parámetros de configuración: Zabbix agent

Descripción general

Los parámetros de configuración permiten personalizar el comportamiento del Zabbix agent.

Especifique los valores de los parámetros en el archivo de configuración del agent de Zabbix (`zabbix_agentd.conf`). Todos los parámetros son **opcionales**, a menos que se indique explícitamente que el parámetro es obligatorio.

Esta página está estructurada de la siguiente manera:

- Descripción general de los parámetros (para consultar todos los detalles de un parámetro, haga clic en su nombre)
- Detalles de los parámetros

Parámetro	Descripción
Alias	Establece un alias para una clave de item.
AllowKey	Permite la ejecución de las claves de item que coincidan con un patrón.
AllowKeyRegexp	Permite la ejecución de las claves de item que coincidan con un patrón basado en una expresión regular.
AllowRoot	Permite que el agent se ejecute como `root`.
BufferSend	Establece la frecuencia de envío de valores del búfer al Zabbix server/proxy.
BufferSize	Establece el número máximo de valores en el búfer de memoria.
DebugLevel	Establece el nivel de depuración.
DenyKey	Deniega la ejecución de las claves de item que coincidan con un patrón.

Parámetro	Descripción
DenyKeyRegexp	Deniega la ejecución de las claves de item que coincidan con un patrón basado en una expresión regular.
EnableRemoteCommands	Permite comandos remotos del Zabbix server. Este parámetro está obsoleto; utilice `AllowKey`, `AllowKeyRegexp`, `DenyKey` o `DenyKeyRegexp` en su lugar.
HeartbeatFrequency	Establece la frecuencia de los mensajes de latido en segundos.
HostInterface	Especifica la interfaz del host.
HostInterfaceItem	Especifica un item para obtener la interfaz del host.
HostMetadata	Especifica los metadatos del host.
HostMetadataItem	Especifica un item del Zabbix agent para obtener los metadatos del host.
Hostname	Especifica el nombre del host.
HostnameItem	Especifica un item del Zabbix agent para obtener el nombre del host.
Include	Especifica un directorio o archivos individuales que se incluirán en el archivo de configuración.
ListenBacklog	Establece el número máximo de conexiones pendientes en la cola TCP.
ListenIP	Especifica la dirección IP de escucha del agent para las conexiones del server/proxy.
ListenPort	Especifica el puerto de escucha del agent.
LoadModule	Especifica el módulo que se cargará al iniciar el agent.
LoadModulePath	Especifica la ruta completa a la ubicación de los módulos del agent.
LogFile	Especifica el archivo de registro del agent.
LogFileSize	Establece el tamaño máximo del archivo de registro.
LogRemoteCommands	Habilita el registro de los comandos de shell ejecutados como advertencias.
LogType	Especifica el tipo de salida del registro.
MaxLinesPerSecond	Establece el número máximo de líneas nuevas por segundo que se enviarán al Zabbix server/proxy al procesar comprobaciones activas `log` o `logrt`.
PidFile	Especifica el archivo PID.
RefreshActiveChecks	Especifica la frecuencia de actualización de la lista de comprobaciones activas.
Server	Especifica las direcciones IP o los nombres DNS de los Zabbix server/proxy.
ServerActive	Especifica la dirección del Zabbix server/proxy o la configuración del clúster desde la que se obtendrán las comprobaciones activas.
SourceIP	Especifica la dirección IP de origen.
StartAgents	Establece el número de instancias prefork de `zabbix_agentd` para procesar comprobaciones pasivas.
Timeout	Especifica cuánto tiempo esperar (en segundos) para establecer la conexión e intercambiar datos con el Zabbix proxy o server.
TLSAccept	Especifica el nivel de cifrado requerido para las conexiones entrantes.
TLSCAFile	Especifica la ruta completa del archivo que contiene los certificados de la CA de nivel superior para la verificación del certificado del par, utilizados para las comunicaciones cifradas entre componentes de Zabbix.
TLSCertFile	Especifica la ruta completa del archivo que contiene el certificado o la cadena de certificados del agent, utilizados para las comunicaciones cifradas entre componentes de Zabbix.
TLSCipherAll	Especifica la cadena de prioridad de GnuTLS o la cadena de cifrado de OpenSSL (TLS 1.2). Anula los criterios predeterminados de selección del conjunto de cifrado para el cifrado basado en certificados y PSK.
TLSCipherAll13	Especifica la cadena de cifrado para OpenSSL 1.1.1 o posterior en TLS 1.3. Anula los criterios predeterminados de selección del conjunto de cifrado para el cifrado basado en certificados y PSK.
TLSCipherCert	Especifica la cadena de prioridad de GnuTLS o la cadena de cifrado de OpenSSL (TLS 1.2). Anula los criterios predeterminados de selección del conjunto de cifrado para el cifrado basado en certificados.
TLSCipherCert13	Especifica la cadena de cifrado para OpenSSL 1.1.1 o posterior en TLS 1.3. Anula los criterios predeterminados de selección del conjunto de cifrado para el cifrado basado en certificados.
TLSCipherPSK	Especifica la cadena de prioridad de GnuTLS o la cadena de cifrado de OpenSSL (TLS 1.2). Anula los criterios predeterminados de selección del conjunto de cifrado para el cifrado basado en PSK.
TLSCipherPSK13	Especifica la cadena de cifrado para OpenSSL 1.1.1 o posterior en TLS 1.3. Anula los criterios predeterminados de selección del conjunto de cifrado para el cifrado basado en PSK.
TLSConnect	Especifica el nivel de cifrado requerido para las conexiones con el Zabbix server o proxy.
TLSCRLFile	Especifica la ruta completa del archivo que contiene los certificados revocados. Este parámetro se utiliza para las comunicaciones cifradas entre componentes de Zabbix.
TLSKeyFile	Especifica la ruta completa del archivo que contiene la clave privada del agent, utilizada para las comunicaciones cifradas entre componentes de Zabbix.
TLSPSKFile	Especifica la ruta completa del archivo que contiene la clave precompartida del agent, utilizada para las comunicaciones cifradas con el Zabbix server o proxy.

Parámetro	Descripción
TLSPSKIdentity	Especifica la cadena de identidad de la clave precompartida, utilizada para las comunicaciones cifradas con el Zabbix server o proxy.
TLSServerCertIssuer	Especifica el emisor permitido del certificado del server (proxy).
TLSServerCertSubject	Especifica el asunto permitido del certificado del server (proxy).
UnsafeUserParameters	Permite pasar todos los caracteres en los argumentos de los parámetros definidos por el usuario.
User	Abandona los privilegios para utilizar un usuario específico existente en el sistema.
UserParameter	Especifica un parámetro definido por el usuario que se supervisará.
UserParameterDir	Especifica la ruta de búsqueda predeterminada para los comandos UserParameter.

Tenga en cuenta lo siguiente:

- Los valores predeterminados reflejan los valores predeterminados del demonio, no los valores de los archivos de configuración distribuidos.
- Los valores admiten variables de entorno.
- Zabbix admite archivos de configuración únicamente con codificación UTF-8 sin BOM.
- Los comentarios que comienzan con # solo se admiten al principio de la línea.

Detalles del parámetro

Alias

Establece un alias para una clave de item. Puede utilizarse para sustituir una clave de item larga y compleja por otra más corta y sencilla.
 Puede haber varios parámetros `Alias`. No se permiten varios parámetros con la misma clave `Alias`.
 Distintas claves `Alias` pueden hacer referencia a la misma clave de item.
 Los alias pueden utilizarse en `HostMetadataItem`, pero no en el parámetro `HostnameItem`.

Ejemplo 1: Obtener el ID del usuario `zabbix`.

```
Alias=zabbix.userid:vfs.file.regexp[/etc/passwd,"^zabbix:.:([0-9]+)",,,,\1]
```

Ahora se puede utilizar la clave abreviada `zabbix.userid` para obtener datos.

Ejemplo 2: Obtener la utilización de la CPU con parámetros predeterminados y personalizados.

```
Alias=cpu.util:system.cpu.util
Alias=cpu.util[*]:system.cpu.util[*]
```

Esto permite utilizar la clave `cpu.util` para obtener el porcentaje de utilización de la CPU con los parámetros predeterminados, así como utilizar `cpu.util[all, idle, avg15]` para obtener datos específicos sobre la utilización de la CPU.

Ejemplo 3: Ejecutar varias reglas de descubrimiento de bajo nivel que procesen los mismos items de descubrimiento.

```
Alias=vfs.fs.discovery[*]:vfs.fs.discovery
```

Ahora es posible configurar varias reglas de descubrimiento utilizando `vfs.fs.discovery`, con distintos parámetros para cada regla, por ejemplo, `vfs.fs.discovery[foo]`, `vfs.fs.discovery[bar]`, etc.

AllowKey

Permite la ejecución de aquellas claves de item que coincidan con un patrón. El patrón de clave es una expresión comodín que admite el carácter `*` para coincidir con cualquier número de caracteres.
 Se pueden definir varias reglas de coincidencia de claves en combinación con `DenyKey`, `AllowKeyRegexp` y `DenyKeyRegexp`. Los parámetros se procesan uno por uno según el orden en que aparecen. El orden determina qué regla coincidente tendrá prioridad. Véase también: Restricting agent checks.

AllowKeyRegexp

Permite la ejecución de los item keys que coincidan con un patrón.
El patrón de la key es una expresión regular.
 Se pueden definir varias reglas de coincidencia de keys en combinación con `DenyKeyRegexp`, `AllowKey` y `DenyKey`. Los parámetros se procesan uno por uno según el orden en que aparecen. El orden determina qué regla coincidente tendrá prioridad. Consulte también: Restricción de las comprobaciones del agent.

Ejemplo:

```
AllowKeyRegexp=^system\.run\[(?:/usr)?/bin/hostname(?: -[fsd])?\]$
```

AllowRoot

Permite que el agent se ejecute como `root`. Si está deshabilitado y el agent se inicia como `root`, el agent intentará cambiar al usuario `zabbix` en su lugar. No tiene efecto si se inicia con un usuario normal.

Predeterminado: 0
 Valores: 0 - no permitir; 1 - permitir

BufferSend

Establezca la frecuencia de envío de valores desde el búfer al Zabbix server/proxy (en segundos).

Predeterminado: 5
 Rango: 1-3600

BufferSize

Establezca el número máximo de valores en el búfer de memoria.
 El agent enviará todos los datos recopilados al server o proxy de Zabbix si el búfer está lleno.

Predeterminado: 100
 Rango: 2-65535

DebugLevel

Establezca el nivel de depuración: - *0* - información básica sobre el inicio y la detención de los procesos de Zabbix; - *1* - información crítica; - *2* - información de error; - *3* - advertencias; - *4* - para depuración (genera mucha información); - *5* - depuración ampliada (genera aún más información).

Predeterminado: 3
 Rango: 0-5

DenyKey

Deniega la ejecución de los item keys que coincidan con un patrón. El patrón de la clave es una expresión comodín que admite el carácter * para coincidir con cualquier cantidad de caracteres.
 Se pueden definir varias reglas de coincidencia de claves en combinación con `AllowKey`, `AllowKeyRegexp` y `DenyKeyRegexp`. Los parámetros se procesan uno por uno según el orden en que aparecen. El orden determina qué regla coincidente tendrá prioridad. Véase también: Restricción de las comprobaciones del agent.

Predeterminado: `DenyKey=system.run[*]`

DenyKeyRegexp

Deniega la ejecución de los item keys que coincidan con un patrón.
El patrón de la key es una expresión regular.
 Se pueden definir varias reglas de coincidencia de keys en combinación con `AllowKeyRegexp`, `AllowKey` y `DenyKey`. Los parámetros se procesan uno por uno según el orden en que aparecen. El orden determina qué regla de coincidencia tendrá prioridad.
 Todos los items `system.run[]` se deniegan de forma predeterminada (`DenyKey=system.run[*]`), a menos que se especifique otra regla `system.run[*]`.

Véase también: Restricción de las comprobaciones del agent.

EnableRemoteCommands

Permitir comandos remotos desde el server de Zabbix.
 Este parámetro está **obsoleto**; utilice `AllowKey=system.run[*]`, `AllowKeyRegexp=^system\.run\[.*\]$`, `DenyKey=system.run[*]` o `DenyKeyRegexp=^system\.run\[.*\]$` en su lugar.
 Es un alias interno de los parámetros `AllowKey`, `AllowKeyRegexp`, `DenyKey` y `DenyKeyRegexp`, según el valor: - 0 - `DenyKey=system.run[*]` (`DenyKeyRegexp=^system\.run\[.*\]$`) - 1 - `AllowKey=system.run[*]` (`AllowKeyRegexp=^system\.run\[.*\]$`)

Predeterminado: 0
 Valores: 0 - no permitir, 1 - permitir

HeartbeatFrequency

Establece la frecuencia de los mensajes de heartbeat en segundos. Se utiliza para supervisar la disponibilidad de las comprobaciones activas.
0 - mensajes de heartbeat deshabilitados.

Predeterminado: 60
 Rango: 0-3600

HostInterface

Especifique la interfaz del host (dirección IP o nombre DNS) utilizada durante el autoregistro del host. Este valor se usará para rellenar la interfaz en el host recién creado y permite configurar explícitamente una dirección IP o DNS. Para más detalles, consulte Using DNS as default interface.

Si no se define, el valor se obtendrá de `HostInterfaceItem`.

El agent devolverá un error y no se iniciará si el valor supera el límite de 255 caracteres.

Rango: 0-255 caracteres

HostInterfaceItem

Especifique un item para determinar la interfaz del host (dirección IP o nombre DNS) durante el registro automático del host. Este valor se utiliza únicamente si `HostInterface` no está definido. Para obtener más información, consulte Uso de DNS como interfaz predeterminada.

Durante una solicitud de registro automático, el agent registrará un mensaje de advertencia si el valor devuelto por el item especificado supera el límite de 255 caracteres.

El item `system.run[]` es compatible independientemente de la configuración de `AllowKey`, `AllowKeyRegexp`, `DenyKey` o `DenyKeyRegexp`.

HostMetadata

Especifique los metadatos para identificar o distinguir el host durante el registro automático del host (agent activo).
`HostMetadata` permite distinguir entre hosts más allá del nombre del host.

Si no se define, el valor se obtendrá de `HostMetadataItem`.

El agent generará un error y no se iniciará si el valor especificado supera el límite de 2034 bytes o si es una cadena que no está codificada en UTF-8.
Cuando un parámetro espera una dirección IP o un nombre DNS, los valores que sean UTF-8 válido, pero no sean direcciones IP ni nombres DNS válidos, también se rechazarán y se notificarán como no válidos.

No se admiten metadatos de varias líneas: la salida se truncará en el primer salto de línea.

Rango: 0-2034 bytes

HostMetadataItem

Especifique un item del Zabbix agent para obtener los metadatos del host. Esta opción solo se utiliza cuando `HostMetadata` no está definido.

El valor de `HostMetadataItem` se obtiene en cada intento de registro automático y se utiliza únicamente durante el proceso de registro automático del host (agent activo). `HostMetadataItem` permite distinguir entre hosts más allá del nombre del host.

Se admiten los parámetros de usuario y los alias. El item `system.run[]` se admite independientemente de la configuración de `AllowKey`, `AllowKeyRegexp`, `DenyKey` o `DenyKeyRegexp`.

Durante una solicitud de registro automático, el agent registrará un mensaje de advertencia si el valor devuelto por el item especificado supera el límite de 65535 puntos de código UTF-8. El valor devuelto por el item debe ser una cadena UTF-8; de lo contrario, se ignorará. Cuando un parámetro espera una dirección IP o un nombre DNS, también se rechazarán y se notificarán como no válidos los valores que sean UTF-8 válido, pero no sean IP ni nombres DNS válidos.

No se admiten metadatos de varias líneas: la salida se truncará en el primer salto de línea.

Nombre de host

Especifique nombres de host únicos y sensibles a mayúsculas y minúsculas. Una lista delimitada por comas.
 Obligatorio para las comprobaciones activas y debe coincidir con los nombres de host configurados en el server.
 El valor se obtiene de `HostnameItem` si no está definido.
Caracteres permitidos: alfanuméricos, '.', ' ', '_' y '-'. Longitud máxima: 128 caracteres por nombre de host, 2048 caracteres para toda la línea.

Predeterminado: establecido por `HostnameItem`

HostnameItem

Especifique un item del agent de Zabbix para obtener el nombre del host.
 Esta opción solo se utiliza cuando Hostname no está definido.
 Los parámetros de usuario o los alias no son compatibles, pero el item `system.run[]` es compatible independientemente de los valores de `AllowKey`, `AllowKeyRegexp`, `DenyKey` o `DenyKeyRegexp`.

Valor predeterminado: `system.hostname`

Include

Especifique un directorio o archivos individuales que se incluirán en el archivo de configuración.
 Durante la instalación de Zabbix, este parámetro se establece en `/usr/local/etc`, a menos que lo cambie durante el tiempo de compilación.

Reglas generales para el parámetro `Include`:

- La ruta puede ser relativa a la ubicación del archivo `zabbix_agentd.conf`.
- Puede especificar varios parámetros `Include` en un solo archivo de configuración.
- Zabbix procesa el parámetro `Include` en cuanto lo lee, así que considere colocarlo al final del archivo de configuración.

Reglas para los archivos incluidos:

- Cada archivo debe ser legible.
- Cada archivo debe tener una sintaxis correcta. De lo contrario, el componente no se iniciará.

Reglas para incluir directorios o patrones de archivos:

- No asuma ningún orden de inclusión (por ejemplo, los archivos no se incluyen en orden alfabético).

- No defina el mismo parámetro en más de un archivo incluido (por ejemplo, para sobrescribir una configuración general con una específica), ya que no puede predecir qué valor se utilizará.
- Tenga cuidado con las copias de seguridad que algunos editores de texto crean automáticamente (por ejemplo, editar `include/my_specific.conf` puede crear `include/my_specific.conf.BAK`), ya que ambos archivos se incluirán.

Para incluir solo los archivos relevantes de un directorio, use el comodín (`*`) para la coincidencia de patrones.

Ejemplo:

```
Include=/absolute/path/to/config/files/*.conf
```

ListenBacklog

Establece el número máximo de conexiones pendientes en la cola TCP.
 El valor predeterminado es una constante codificada, que depende del sistema.
 El valor máximo admitido también depende del sistema; los valores demasiado altos pueden truncarse silenciosamente al ”implementation-specified maximum”.

Predeterminado: `SOMAXCONN`
 Rango: 0 - INT_MAX

ListenIP

Especifique la dirección IP de escucha del agent para conexiones desde el server/proxy. Una lista delimitada por comas.

Predeterminado: `0.0.0.0`

ListenPort

Especifique el puerto de escucha del agent para conexiones desde el server/proxy.

Predeterminado: 10050
 Rango: 1024-32767

LoadModule

Especifique el módulo que se cargará al iniciar el agent.
 Los módulos se utilizan para ampliar la funcionalidad del agent.
 El módulo debe estar ubicado en el directorio especificado por `LoadModulePath` o la ruta debe preceder al nombre del módulo. Si la ruta precedente es absoluta (empieza con `/`), entonces `LoadModulePath` se ignora.
 Formatos: - `LoadModule=<module.so>` - `LoadModule=<path/module.so>` - `LoadModule=</abs_path/module.so>`

Se permite incluir varios parámetros `LoadModule`.

LoadModulePath

Especifique la ruta completa a la ubicación de los módulos del agent.
 El valor predeterminado depende de las opciones de compilación.

LogFile

Especifique el archivo de registro del agent.

Obligatorio: Sí, si `LogType` está establecido en `file`; en caso contrario, no

LogFileSize

Establezca el tamaño máximo del archivo de registro (en MB).
0 - deshabilita la rotación automática de registros.
Nota: Si se alcanza el límite de tamaño del archivo de registro y la rotación del archivo falla, por cualquier motivo, el archivo de registro existente se trunca y se inicia uno nuevo.

Predeterminado: 1
 Rango: 0-1024

LogRemoteCommands

Habilita el registro de los comandos de shell ejecutados como advertencias.
 Los comandos solo se registrarán si se ejecutan de forma remota.
 No se crearán entradas de registro si `system.run[]` se inicia localmente mediante los parámetros `HostMetadataItem`, `HostInterfaceItem` o `HostnameItem`.

Predeterminado: 0
 Valores: 0 - deshabilitado, 1 - habilitado

LogType

Especifique el tipo de salida del registro: - *file* - escribir el registro en el archivo especificado por el parámetro `LogFile`; - *system* - escribir el registro en syslog; - *console* - escribir el registro en la salida estándar.

Predeterminado: `file`

MaxLinesPerSecond

Establece el número máximo de nuevas líneas por segundo para enviar al server/proxy de Zabbix al procesar comprobaciones activas `log` o `logrt`.
 El valor proporcionado será reemplazado por el parámetro `maxlines`, especificado en la clave del item

log o logrt.
Nota: Zabbix procesará 10 veces más nuevas líneas que las establecidas en MaxLinesPerSecond para buscar la cadena requerida en items de log.

Predeterminado: 20
 Rango: 1-1000

PidFile

Especifique el archivo PID.

Predeterminado: /tmp/zabbix_agentd.pid

RefreshActiveChecks

Especifique la frecuencia de actualización de la lista de comprobaciones activas, en segundos.
 Tenga en cuenta que, después de que falle la actualización de las comprobaciones activas, el siguiente intento se realizará en 60 segundos.

Predeterminado: 5
 Rango: 1-86400

Server

Especifique las direcciones IP, opcionalmente en notación CIDR, o los nombres DNS de los servers/proxies de Zabbix. Una lista delimitada por comas.
 Las conexiones entrantes solo se aceptarán desde los hosts enumerados aquí. Si la compatibilidad con IPv6 está habilitada, entonces 127.0.0.1, ::127.0.0.1, ::ffff:127.0.0.1 se tratan por igual y ::/0 permitirá cualquier dirección IPv4 o IPv6. 0.0.0.0/0 puede usarse para permitir cualquier dirección IPv4. Tenga en cuenta que las "direcciones IPv6 compatibles con IPv4" (0000::/96 prefix) son compatibles, pero están obsoletas según RFC4291. Se permiten espacios.

Ejemplo:

```
Server=127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.example.com
```

Obligatorio: sí, si StartAgents no está establecido explícitamente en 0

ServerActive

Especifique la dirección del server/proxy de Zabbix o la configuración del clúster para obtener active checks de él.
 La dirección del server/proxy es una dirección IP o un nombre DNS, y un puerto opcional separado por dos puntos.
La configuración del clúster es una o más direcciones de miembros del grupo de server o proxy separadas por punto y coma. Se pueden especificar varios servers/clusters de Zabbix y proxies de Zabbix, separados por comas. A menos que se usen grupos de proxy, no se debe especificar más de un proxy de Zabbix para cada server/clúster de Zabbix. Si se especifica un proxy de Zabbix, entonces no se debe especificar el server/clúster de Zabbix para ese proxy.
Se pueden proporcionar varias direcciones separadas por comas para usar varios servers de Zabbix independientes en paralelo. Se permiten espacios.
Si no se especifica el puerto, se usa el puerto predeterminado.
Las direcciones IPv6 deben ir entre corchetes si se especifica el puerto para ese host. Si no se especifica el puerto, los corchetes para direcciones IPv6 son opcionales.
Si no se especifica este parámetro, los active checks se deshabilitan.

Ejemplo para proxy de Zabbix:

```
ServerActive=127.0.0.1:10051
```

Ejemplo para grupo de proxy de Zabbix:

```
ServerActive=proxy1.example.com;proxy2.example.com;proxy3.example.com;proxy4.example.com;proxy5.example.co
```

Ejemplo para varios servers:

```
ServerActive=127.0.0.1:20051,zabbix.domain,[::1]:30051,::1,[12fc::1]
```

Ejemplo para alta disponibilidad:

```
ServerActive=zabbix.cluster.node1;zabbix.cluster.node2:20051;zabbix.cluster.node3
```

Ejemplo para alta disponibilidad con dos clústeres y un server:

```
ServerActive=zabbix.cluster.node1;zabbix.cluster.node2:20051,zabbix.cluster2.node1;zabbix.cluster2.node2,z
```

SourceIP

Especifique la dirección IP de origen para:

- conexiones salientes a Zabbix server o Zabbix proxy;
- realizar conexiones al ejecutar algunos items (web.page.get, net.tcp.port, etc.).

StartAgents

Establece el número de instancias pre-fork de zabbix_agentd para procesar comprobaciones pasivas.
 Si se establece en 0, las comprobaciones pasivas se deshabilitan y el agent no escuchará en ningún puerto TCP.

Predeterminado: 10
 Rango: 0-100

Tiempo de espera

Especifique cuánto tiempo esperar (en segundos) para establecer la conexión e intercambiar datos con el proxy o server de Zabbix.

Este parámetro define la duración de varias operaciones de comunicación, incluidas:

- esperar una respuesta del server de Zabbix;
- enviar solicitudes al server de Zabbix, incluidas las solicitudes de configuración de item y los datos de item en active checks;
- recuperar datos de registro a través de logfile;
- enviar mensajes de latido;
- duración máxima para comprobaciones `vfs.*`;
- ser utilizado por los módulos del agent de Zabbix;
- servir como valor de reserva en escenarios en los que un server o proxy anterior a la versión 7.0 envía comprobaciones sin tiempos de espera.

Este tiempo de espera **no** se utilizará para aquellas comprobaciones del agent que tienen ajustes de tiempo de espera configurables en el frontend (a nivel global, de proxy o por item).

Predeterminado: 3
 Rango: 1-30

TLSAccept

Especifique el nivel de cifrado requerido para las conexiones entrantes.
 Se utiliza para comprobaciones pasivas.
 Se pueden especificar varios valores, separados por comas: - *unencrypted* - aceptar conexiones sin cifrado (predeterminado); - *psk* - aceptar conexiones con TLS y una clave precompartida (PSK); - *cert* - aceptar conexiones con TLS y un certificado.

Obligatorio: sí, si se definen parámetros de certificado TLS o PSK (incluso para una conexión *unencrypted*); en caso contrario, no

TLSCAFile

Especifique la ruta completa del archivo que contiene los certificados de la CA de nivel superior para la verificación del certificado del par, utilizados para las comunicaciones cifradas entre los componentes de Zabbix.

TLSCertFile

Especifique la ruta completa del archivo que contiene el certificado del agent o la cadena de certificados, utilizado para las comunicaciones cifradas con los componentes de Zabbix.

TLSCipherAll

Especifique la cadena de prioridad de GnuTLS o la cadena de cifrado de OpenSSL (TLS 1.2). Anule los criterios de selección de conjuntos de cifrado predeterminados para el cifrado basado en certificados y PSK.

Ejemplo:

```
TLS_AES_256_GCM_SHA384:TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
```

TLSCipherAll13

Especifique la cadena de cifrado para OpenSSL 1.1.1 o posterior en TLS 1.3. Anule los criterios de selección predeterminados de la suite de cifrado para el cifrado basado en certificados y PSK.

Ejemplo para GnuTLS:

```
NONE:+VERS-TLS1.2:+ECDHE-RSA:+RSA:+ECDHE-PSK:+PSK:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL
```

Ejemplo para OpenSSL:

```
EECDH+aRSA+AES128:RSA+aRSA+AES128:kECDHEPSK+AES128:kPSK+AES128
```

TLSCipherCert

Especifique la cadena de prioridad de GnuTLS o la cadena de cifrado de OpenSSL (TLS 1.2). Anule los criterios predeterminados de selección de la suite de cifrado para el cifrado basado en certificados.

Ejemplo para GnuTLS:

```
NONE:+VERS-TLS1.2:+ECDHE-RSA:+RSA:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL:+COMP-NULL:+SIG
```

Ejemplo para OpenSSL:

```
EECDH+aRSA+AES128:RSA+aRSA+AES128
```

TLSCipherCert13

Especifique la cadena de cifrado para OpenSSL 1.1.1 o una versión posterior en TLS 1.3. Anule los criterios de selección predeterminados de la suite de cifrado para el cifrado basado en certificados.

TLSCipherPSK

Especifique la cadena de prioridad de GnuTLS o la cadena de cifrado de OpenSSL (TLS 1.2). Anule los criterios predeterminados de selección de la suite de cifrado para el cifrado basado en PSK.

Ejemplo para GnuTLS:

```
NONE:+VERS-TLS1.2:+ECDHE-PSK:+PSK:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL:+COMP-NULL:+SIG
```

Ejemplo para OpenSSL:

```
kECDHEPSK+AES128:kPSK+AES128
```

TLSCipherPSK13

Especifique la cadena de cifrado para OpenSSL 1.1.1 o una versión posterior en TLS 1.3. Anule los criterios de selección predeterminados de la suite de cifrado para el cifrado basado en PSK.

Ejemplo:

```
TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
```

TLSConnect

Especifique el nivel de cifrado requerido para las conexiones del agent a Zabbix server o proxy.
 Se usa para comprobaciones activas.
 Solo se puede especificar un valor: - *unencrypted* - conectar sin cifrado (predeterminado); - *psk* - conectar usando TLS y una clave precompartida (PSK); - *cert* - conectar usando TLS y un certificado.

Obligatorio: sí, si se definen parámetros de certificado TLS o PSK (incluso para una conexión *unencrypted*); en caso contrario, no

TLSCRLFile

Especifique la ruta completa del archivo que contiene los certificados revocados. Este parámetro se utiliza para las comunicaciones cifradas entre los componentes de Zabbix.

TLSKeyFile

Especifique la ruta completa del archivo que contiene la clave privada del agent, utilizada para las comunicaciones cifradas entre los componentes de Zabbix.

TLSPSKFile

Especifique la ruta completa del archivo que contiene la clave precompartida del agent, utilizada para las comunicaciones cifradas con el server o proxy de Zabbix.

TLSPSKIdentity

Especifique la cadena de identidad de la clave precompartida, utilizada para comunicaciones cifradas con Zabbix server o proxy.

TLSServerCertIssuer

Especifique el emisor del certificado del server (proxy) permitido.

TLSServerCertSubject

Especifica el subject del certificado del server (proxy) permitido.

UnsafeUserParameters

Permite que todos los caracteres se pasen como argumentos a parámetros definidos por el usuario. Los siguientes caracteres no están permitidos: \ ' " ` * ? [ ] { } ~ $! & ; ( ) < > | # @ %
Además, no se permiten caracteres de nueva línea.

Predeterminado: 0
 Valores: 0 - no permitir, 1 - permitir

Usuario

Reduce los privilegios a un usuario específico y existente en el sistema.
Solo tiene efecto si se ejecuta como `root` y `AllowRoot` está deshabilitado.

Predeterminado: `zabbix`

UserParameter

Especifique un parámetro definido por el usuario para supervisar. Puede haber varios parámetros definidos por el usuario.
Formato: `UserParameter=<key>,<shell command><br>`Tenga en cuenta que el comando de shell no debe devolver una cadena vacía ni solo EOL. Los comandos de shell pueden tener rutas relativas si se especifica el parámetro `UserParameterDir`.

Ejemplo:

```
UserParameter=system.test,who|wc -l
UserParameter=check_cpu,./custom_script.sh
```

UserParameterDir

Especifique la ruta de búsqueda predeterminada para los comandos `UserParameter`. Si se utiliza, el agent cambiará su directorio de trabajo al especificado aquí antes de ejecutar un comando. De este modo, los comandos `UserParameter` pueden tener un prefijo relativo `./` en lugar de una ruta completa.
Solo se permite una entrada.

Ejemplo:

```
UserParameterDir=/opt/myscripts
```

Véase también

1. Diferencias en la configuración del agent de Zabbix para comprobaciones activas y pasivas a partir de la versión 2.0.0

Parámetros de configuración: Zabbix agent (Windows)

Descripción general

Los parámetros de configuración permiten personalizar el comportamiento del Zabbix agent en Windows.

Especifique los valores de los parámetros en el archivo de configuración del agent de Zabbix (`zabbix_agentd.conf`). Todos los parámetros son **opcionales**, a menos que se indique explícitamente que el parámetro es obligatorio.

Esta página está estructurada de la siguiente manera:

- Descripción general de los parámetros (para consultar todos los detalles de un parámetro, haga clic en su nombre).
- Detalles de los parámetros.

Parámetro	Descripción
Alias	Establece un alias para una clave de item.
AllowKey	Permite la ejecución de las claves de item que coincidan con un patrón.
AllowKeyRegexp	Permite la ejecución de las claves de item que coincidan con un patrón basado en una expresión regular.
BufferSend	Establece la frecuencia con la que se envían los valores del búfer al Zabbix server/proxy.
BufferSize	Establece el número máximo de valores en el búfer de memoria.
DebugLevel	Establece el nivel de depuración.
DenyKey	Deniega la ejecución de las claves de item que coincidan con un patrón.
DenyKeyRegexp	Deniega la ejecución de las claves de item que coincidan con un patrón basado en una expresión regular.
EnableRemoteCommands	Permite comandos remotos del Zabbix server. Este parámetro está obsoleto; utilice `AllowKey`, `AllowKeyRegexp`, `DenyKey` o `DenyKeyRegexp` en su lugar.
HeartbeatFrequency	Establece la frecuencia de los mensajes de latido en segundos.
HostInterface	Especifica la interfaz del host.
HostInterfaceItem	Especifica un item para obtener la interfaz del host.
HostMetadata	Especifica los metadatos del host.
HostMetadataItem	Especifica un item del Zabbix agent para obtener los metadatos del host.
Hostname	Especifica el nombre del host.
HostnameItem	Especifica un item del Zabbix agent para obtener el nombre del host.
Include	Especifica un directorio o archivos individuales que se incluirán en el archivo de configuración.
ListenBacklog	Establece el número máximo de conexiones pendientes en la cola TCP.
ListenIP	Especifica la dirección IP de escucha del agent para las conexiones del server/proxy.
ListenPort	Especifica el puerto de escucha del agent.
LogFile	Especifica el archivo de registro del agent.
LogFileSize	Establece el tamaño máximo del archivo de registro.
LogRemoteCommands	Habilita el registro de los comandos de shell ejecutados como advertencias.
LogType	Especifica el tipo de salida del registro.
MaxLinesPerSecond	Establece el número máximo de líneas nuevas por segundo que se enviarán al Zabbix server/proxy al procesar comprobaciones activas `log`, `logrt` o `eventlog`.

Parámetro	Descripción
PerfCounter	Define un nuevo parámetro <parameter_name>, que corresponde al valor promedio del contador de rendimiento del sistema <perf_counter_path> durante el período especificado <period> (en segundos).
PerfCounterEn	Define un nuevo parámetro <parameter_name>, que corresponde al valor promedio del contador de rendimiento del sistema <perf_counter_path> durante el período especificado <period> (en segundos). A diferencia de PerfCounter, las rutas de los contadores de rendimiento deben estar en inglés.
RefreshActiveChecks	Especifica la frecuencia de actualización de la lista de comprobaciones activas.
Server	Especifica las direcciones IP o los nombres DNS de los Zabbix servers/proxies.
ServerActive	Especifica la dirección del Zabbix server/proxy o la configuración del clúster desde la que se obtendrán las comprobaciones activas.
SourceIP	Especifica la dirección IP de origen.
StartAgents	Establece el número de instancias prefork de `zabbix_agentd` para procesar comprobaciones pasivas.
Timeout	Especifica cuánto tiempo esperar (en segundos) para establecer la conexión e intercambiar datos con el Zabbix server o proxy.
TLSAccept	Especifica el nivel de cifrado requerido para las conexiones entrantes.
TLSCAFile	Especifica la ruta completa del archivo que contiene los certificados de las CA de nivel superior para la verificación del certificado del par, utilizados para las comunicaciones cifradas entre los componentes de Zabbix.
TLSCertFile	Especifica la ruta completa del archivo que contiene el certificado o la cadena de certificados del agent, utilizados para las comunicaciones cifradas entre los componentes de Zabbix.
TLSConnect	Especifica el nivel de cifrado requerido para las conexiones al Zabbix server o proxy.
TLSCRLFile	Especifica la ruta completa del archivo que contiene los certificados revocados. Este parámetro se utiliza para las comunicaciones cifradas entre los componentes de Zabbix.
TLSKeyFile	Especifica la ruta completa del archivo que contiene la clave privada del agent, utilizada para las comunicaciones cifradas entre los componentes de Zabbix.
TLSPSKFile	Especifica la ruta completa del archivo que contiene la clave precompartida del agent, utilizada para las comunicaciones cifradas con el Zabbix server o proxy.
TLSPSKIdentity	Especifica la cadena de identidad de la clave precompartida, utilizada para las comunicaciones cifradas con el Zabbix server o proxy.
TLSServerCertIssuer	Especifica el emisor permitido del certificado del server (proxy).
TLSServerCertSubject	Especifica el sujeto permitido del certificado del server (proxy).
UnsafeUserParameters	Permite pasar todos los caracteres en los argumentos de los parámetros definidos por el usuario.
UserParameter	Especifica un parámetro definido por el usuario que se supervisará.
UserParameterDir	Especifica la ruta de búsqueda predeterminada para los comandos UserParameter.

Tenga en cuenta lo siguiente:

- Los valores predeterminados reflejan los valores predeterminados del daemon, no los valores de los archivos de configuración distribuidos.
- Los valores admiten variables de entorno.
- Zabbix admite archivos de configuración únicamente con codificación UTF-8 y sin BOM.
- Los comentarios que comienzan con # solo se admiten al principio de la línea.

Detalles del parámetro

Alias

Establezca un alias para una clave de item. Puede utilizarse para sustituir una clave de item larga y compleja por otra más corta y sencilla.
 Puede haber varios parámetros `Alias`. No se permiten varios parámetros con la misma clave `Alias`.
 Distintas claves `Alias` pueden hacer referencia a la misma clave de item.
 Los alias pueden utilizarse en `HostMetadataItem`, pero no en los parámetros `HostnameItem` ni `PerfCounter`.

Ejemplo 1: Recuperación del uso del archivo de paginación en porcentaje desde el server.

```
Alias=pg_usage:perf_counter[\Paging File(_Total)\% Usage]
```

Ahora se puede utilizar la clave abreviada `pg_usage` para recuperar datos.

Ejemplo 2: Obtención de la carga de la CPU con parámetros predeterminados y personalizados.

```
Alias=cpu.load:system.cpu.load
Alias=cpu.load[*]:system.cpu.load[*]
```

Esto permite utilizar la clave `cpu.load` para obtener la carga de la CPU con los parámetros predeterminados, así como utilizar `cpu.load[percpu,avg15]` para obtener datos específicos sobre la carga de la CPU.

Ejemplo 3: Ejecución de varias reglas de descubrimiento de bajo nivel que procesan los mismos items de descubrimiento.

```
Alias=vfs.fs.discovery[*]:vfs.fs.discovery
```

Ahora es posible configurar varias reglas de descubrimiento utilizando `vfs.fs.discovery` con distintos parámetros para cada regla, por ejemplo, `vfs.fs.discovery[foo]`, `vfs.fs.discovery[bar]`, etc.

AllowKey

Permite la ejecución de aquellas claves de item que coincidan con un patrón. El patrón de clave es una expresión comodín que admite el carácter * para coincidir con cualquier número de caracteres.
 Se pueden definir varias reglas de coincidencia de claves en combinación con `DenyKey`, `AllowKeyRegexp` y `DenyKeyRegexp`. Los parámetros se procesan uno por uno según el orden en que aparecen. El orden determina qué regla coincidente tendrá prioridad. Consulte también: Restricting agent checks.

AllowKeyRegexp

Permite la ejecución de los item keys que coincidan con un patrón.
El patrón de la key es una expresión regular.
 Se pueden definir varias reglas de coincidencia de keys en combinación con `DenyKeyRegexp`, `AllowKey` y `DenyKey`. Los parámetros se procesan uno por uno según el orden en que aparecen. El orden determina qué regla coincidente tendrá prioridad. Consulte también: Restricción de las comprobaciones del agent.

Ejemplo:

```
AllowKeyRegexp=^system\.run\[(?:/usr)?/bin/hostname(?: -[fsd])?\]$
```

BufferSend

Establezca la frecuencia de envío de valores desde el búfer al Zabbix server/proxy (en segundos).

Predeterminado: 5
 Rango: 1-3600

BufferSize

Establece el número máximo de valores en el búfer de memoria.
 El agent enviará todos los datos recopilados al server o proxy de Zabbix si el búfer está lleno.

Predeterminado: 100
 Rango: 2-65535

DebugLevel

Establezca el nivel de depuración: - *0* - información básica sobre el inicio y la detención de los procesos de Zabbix; - *1* - información crítica; - *2* - información de error; - *3* - advertencias; - *4* - para depuración (genera mucha información); - *5* - depuración ampliada (genera aún más información).

Predeterminado: 3
 Rango: 0-5

DenyKey

Deniega la ejecución de los item keys que coincidan con un patrón. El patrón de la key es una expresión comodín que admite el carácter * para coincidir con cualquier cantidad de caracteres.
 Se pueden definir varias reglas de coincidencia de keys en combinación con `AllowKey`, `AllowKeyRegexp` y `DenyKeyRegexp`. Los parámetros se procesan uno por uno según el orden en que aparecen. El orden determina qué regla coincidente tendrá prioridad. Véase también: Restricción de las comprobaciones del agent.

Predeterminado: `DenyKey=system.run[*]`

DenyKeyRegexp

Deniega la ejecución de los item keys que coincidan con un patrón.
El patrón de la key es una expresión regular.
 Se pueden definir varias reglas de coincidencia de keys en combinación con `AllowKeyRegexp`, `AllowKey` y `DenyKey`. Los parámetros se procesan uno por uno según el orden en que aparecen. El orden determina qué regla de coincidencia tendrá prioridad.
 Todos los items `system.run[]` se deniegan de forma predeterminada (`DenyKey=system.run[*]`), a menos que se especifique otra regla `system.run[*]`.

Véase también: Restricción de las comprobaciones del agent.

EnableRemoteCommands

Permitir comandos remotos desde el Zabbix server.
 Este parámetro está **obsoleto**; utilice en su lugar `AllowKey=system.run[*]`, `AllowKeyRegexp=^system\.run\[.*\]$`, `DenyKey=system.run[*]` o `DenyKeyRegexp=^system\.run\[.*\]$`.
Es un alias interno de los parámetros `AllowKey`, `AllowKeyRegexp`, `DenyKey` y `DenyKeyRegexp`, según el valor: - `0` - `DenyKey=system.run[*]` (`DenyKeyRegexp=^system\.run\[.*\]$`) - `1` - `AllowKey=system.run[*]` (`AllowKeyRegexp=^system\.`

Predeterminado: 0
 Valores: 0 - no permitir, 1 - permitir

HeartbeatFrequency

Establece la frecuencia de los mensajes de heartbeat en segundos. Se usa para supervisar la disponibilidad de active checks.
0 - mensajes de heartbeat deshabilitados.

Predeterminado: 60
 Rango: 0-3600

HostInterface

Especifique la interfaz del host (dirección IP o nombre DNS) utilizada durante el autoregistro del host. Este valor se usará para rellenar la interfaz en el host recién creado y permite configurar explícitamente una dirección IP o DNS. Para más detalles, consulte Using DNS as default interface.

Si no se define, el valor se obtendrá de HostInterfaceItem.

El agent mostrará un error y no se iniciará si el valor supera el límite de 255 caracteres.

Rango: 0-255 caracteres

HostInterfaceItem

Especifique un item para determinar la interfaz del host (dirección IP o nombre DNS) durante el registro automático del host.
Este valor se utiliza únicamente si `HostInterface` no está definido.
Para obtener más información, consulte Uso de DNS como interfaz predeterminada.

Durante una solicitud de registro automático, el agent registrará un mensaje de advertencia si el valor devuelto por el item especificado supera el límite de 255 caracteres.

El item `system.run[]` es compatible independientemente de la configuración de `AllowKey`, `AllowKeyRegexp`, `DenyKey` o `DenyKeyRegexp`.

HostMetadata

Especifique los metadatos utilizados para identificar o distinguir el host durante la autoregistración del host (agent activo).
`HostMetadata` permite distinguir entre hosts más allá del nombre del host.

Si no se define, el valor se obtendrá de `HostMetadataItem`.

El agent generará un error y no se iniciará si el valor especificado supera el límite de 2034 bytes o si es una cadena que no está codificada en UTF-8.
Cuando un parámetro espera una dirección IP o un nombre DNS, también se rechazarán y se notificarán como no válidos los valores que sean UTF-8 válido, pero no sean IP válidas ni nombres DNS válidos.

No se admiten metadatos de varias líneas: la salida se truncará en el primer salto de línea.

Rango: 0-2034 bytes

HostMetadataItem

Especifique un item del Zabbix agent para obtener los metadatos del host. Esta opción solo se utiliza cuando `HostMetadata` no está definido.

El valor de `HostMetadataItem` se obtiene en cada intento de registro automático y se utiliza únicamente durante el proceso de registro automático del host (agent activo). `HostMetadataItem` permite distinguir entre hosts más allá del nombre del host.

Se admiten parámetros de usuario, contadores de rendimiento y alias. El item `system.run[]` se admite independientemente de la configuración de `AllowKey`, `AllowKeyRegexp`, `DenyKey` o `DenyKeyRegexp`.

Durante una solicitud de registro automático, el agent registrará un mensaje de advertencia si el valor devuelto por el item especificado supera el límite de 65535 puntos de código UTF-8. El valor devuelto por el item debe ser una cadena UTF-8; de lo contrario, se ignorará. Cuando un parámetro espera una dirección IP o un nombre DNS, también se rechazarán y se notificarán como no válidos los valores que sean UTF-8 válido, pero no sean direcciones IP ni nombres DNS válidos.

No se admiten metadatos de varias líneas: la salida se truncará en el primer salto de línea.

Nombre de host

Especifique nombres de host únicos y sensibles a mayúsculas y minúsculas. Una lista delimitada por comas.
 Requerido para las comprobaciones activas y debe coincidir con los nombres de host configurados en el server. El valor se obtiene de `HostnameItem` si no está definido.
Caracteres permitidos: alfanuméricos, '.', ' ', '_' y '-'. Longitud máxima: 128 caracteres por nombre de host, 2048 caracteres para toda la línea.

Predeterminado: establecido por HostnameItem

HostnameItem

Especifique un item del agent de Zabbix para obtener el nombre del host.
 Esta opción solo se utiliza cuando Hostname no está definido.
 No se admiten parámetros de usuario, contadores de rendimiento ni alias, pero el item system.run[] se admite independientemente de los valores de AllowKey, AllowKeyRegexp, DenyKey o DenyKeyRegexp.
Consulte también una descripción más detallada.

Valor predeterminado: system.hostname

Incluir

Especifique un directorio o archivos individuales que se incluirán en el archivo de configuración (ubicado en C:\Program Files\Zabbix Agent de forma predeterminada si Zabbix agent se instala mediante paquetes de instalación MSI de Windows; ubicado en la carpeta especificada durante la instalación si Zabbix agent se instala como un archivo zip).

Reglas generales para el parámetro Include:

- La ruta puede ser relativa a la ubicación del archivo zabbix_agentd.conf.
- Puede especificar varios parámetros Include en un mismo archivo de configuración.
- Zabbix procesa el parámetro Include en cuanto lo lee, por lo que se recomienda colocarlo al final del archivo de configuración.

Reglas para los archivos incluidos:

- Cada archivo debe poder leerse.
- Cada archivo debe tener una sintaxis correcta. De lo contrario, el componente no se iniciará.

Reglas para incluir directorios o patrones de archivos:

- No dé por supuesto ningún orden de inclusión (por ejemplo, los archivos no se incluyen en orden alfabético).
- No defina el mismo parámetro en más de un archivo incluido (por ejemplo, para sobrescribir una configuración general con una específica), ya que no puede predecir qué valor se utilizará.
- Preste atención a las copias de seguridad que algunos editores de texto crean automáticamente (por ejemplo, al editar include/my_specific.conf se puede crear include/my_specific.conf.BAK), ya que ambos archivos se incluirán.

Para incluir únicamente los archivos relevantes de un directorio, utilice el comodín (*) para la coincidencia de patrones.

Ejemplo:

```
Include=C:\Program Files\Zabbix Agent\zabbix_agentd.d\*.conf
```

ListenBacklog

Establece el número máximo de conexiones pendientes en la cola TCP.
 El valor predeterminado es una constante codificada, que depende del sistema.
 El valor máximo admitido también depende del sistema; los valores demasiado altos pueden truncarse silenciosamente al ”implementation-specified maximum”.

Predeterminado: SOMAXCONN
 Rango: 0 - INT_MAX

ListenIP

Especifique la dirección IP de escucha del agent para conexiones desde el server/proxy. Una lista delimitada por comas.

Predeterminado: 0.0.0.0

ListenPort

Especifique el puerto de escucha del agent para conexiones desde el server/proxy.

Predeterminado: 10050
 Rango: 1024-32767

LogFile

Especifique el archivo de registro del agent.

Predeterminado: c:\zabbix_agentd.log
 Obligatorio: Sí, si LogType está establecido en file; en caso contrario, no

LogFileSize

Establece el tamaño máximo del archivo de registro (en MB).
0 - deshabilita la rotación automática de registros.
Nota: Si se alcanza el límite de tamaño del archivo de registro y la rotación del archivo falla, por cualquier motivo, el archivo de registro existente se trunca y se inicia uno nuevo.

Predeterminado: 1
 Rango: 0-1024

LogRemoteCommands

Habilita el registro de los comandos de shell ejecutados como advertencias.
 Los comandos se registrarán solo si se ejecutan de forma remota.
 No se crearán entradas de registro si `system.run[]` se inicia localmente mediante los parámetros `HostMetadataItem`, `HostInterfaceItem` o `HostnameItem`.

Predeterminado: 0
 Valores: 0 - deshabilitado, 1 - habilitado

LogType

Especifique el tipo de salida del registro: - *file* - escribir el registro en el archivo especificado por el parámetro `LogFile`; - *system* - escribir el registro en el Registro de eventos de Windows; - *console* - escribir el registro en la salida estándar.

Predeterminado: `file`

MaxLinesPerSecond

Establece el número máximo de nuevas líneas por segundo para el envío a Zabbix server/proxy al procesar comprobaciones activas `log`, `logrt` o `eventlog`.
 El valor proporcionado se sobrescribirá con el parámetro `maxlines`, incluido en la clave del item `log`, `logrt` o `eventlog`.
Nota: Zabbix procesará 10 veces más nuevas líneas que las establecidas en `MaxLinesPerSecond` para buscar la cadena requerida en los items de registro.

Predeterminado: 20
 Rango: 1-1000

PerfCounter

Defina un nuevo parámetro <parameter_name> que sea el valor promedio del contador de rendimiento del sistema <perf_counter_path> para el período de tiempo especificado <period> (en segundos).
Sintaxis: <parameter_name>,”<perf_counter_pat

Por ejemplo, si desea recibir el número promedio de interrupciones del procesador por segundo durante el último minuto, puede definir un nuevo parámetro ”interrupts” de la siguiente manera:

```
PerfCounter = interrupts,"\Processor(0)\Interrupts/sec",60
```

Tenga en cuenta las comillas dobles alrededor de la ruta del contador de rendimiento. El nombre del parámetro (interrupts) se usará como la clave del item al crear un item. Las muestras para calcular el valor promedio se tomarán cada segundo.
Puede ejecutar `typeperf -qx` para obtener la lista de todos los contadores de rendimiento disponibles en Windows.

PerfCounterEn

Define un nuevo parámetro <parameter_name> que es el valor promedio del contador de rendimiento del sistema <perf_counter_path> para el período de tiempo especificado <period> (en segundos). En comparación con PerfCounter, las rutas de perfcounter deben estar en inglés. Solo compatible con **Windows Server 2008/Vista** y versiones posteriores.
Sintaxis: <parameter_name>,”<perf_counter_path>”,<period>

Por ejemplo, si desea recibir el número promedio de interrupciones del procesador por segundo durante el último minuto, puede definir un nuevo parámetro ”interrupts” de la siguiente manera:

```
PerfCounterEn = interrupts,"\Processor(0)\Interrupts/sec",60
```

Tenga en cuenta las comillas dobles alrededor de la ruta del contador de rendimiento. El nombre del parámetro (interrupts) se usará como la clave del item al crear un item. Las muestras para calcular el valor promedio se tomarán cada segundo.
Puede encontrar la lista de cadenas en inglés consultando la siguiente clave del registro: `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Perflib\009`.

RefreshActiveChecks

Especifique la frecuencia de actualización de la lista de comprobaciones activas, en segundos.
 Tenga en cuenta que, después de que falle la actualización de las comprobaciones activas, el siguiente intento se realizará en 60 segundos.

Predeterminado: 5
 Rango: 1-86400

Server

Especifique las direcciones IP, opcionalmente en notación CIDR, o los nombres DNS de los servers/proxies de Zabbix. Una lista delimitada por comas.
 Las conexiones entrantes solo se aceptarán desde los hosts enumerados aquí. Si la compatibilidad con IPv6 está habilitada, entonces `127.0.0.1`, `::127.0.0.1`, `::ffff:127.0.0.1` se tratan por igual y `::/0` permitirá cualquier dirección IPv4 o IPv6. `0.0.0.0/0` puede usarse para permitir cualquier dirección IPv4. Tenga en cuenta que las ”direcciones IPv6 compatibles con IPv4” (`0000::/96 prefix`) son compatibles, pero están obsoletas según RFC4291. Se permiten espacios.

Ejemplo:

```
Server=127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.example.com
```

Obligatorio: sí, si StartAgents no está establecido explícitamente en 0

ServerActive

Especifique la dirección del server/proxy de Zabbix o la configuración del clúster para obtener las comprobaciones activas.
 La dirección del server/proxy es una dirección IP o un nombre DNS y un puerto opcional separados por dos puntos.
La configuración del clúster es una o más direcciones de miembros de un grupo de server o proxy separadas por punto y coma. Se pueden especificar varios servers/clusters de Zabbix y proxies de Zabbix, separados por coma. A menos que se usen grupos de proxy, no se debe especificar más de un proxy de Zabbix para cada server/clúster de Zabbix. Si se especifica un proxy de Zabbix, entonces no se debe especificar el server/clúster de Zabbix para ese proxy.
Se pueden proporcionar varias direcciones delimitadas por comas para usar varios servers de Zabbix independientes en paralelo. Se permiten espacios.
Si no se especifica el puerto, se usa el puerto predeterminado.
Las direcciones IPv6 deben ir entre corchetes si se especifica el puerto para ese host. Si no se especifica el puerto, los corchetes para direcciones IPv6 son opcionales.
Si no se especifica este parámetro, las comprobaciones activas se deshabilitan.

Ejemplo para proxy de Zabbix:

```
ServerActive=127.0.0.1:10051
```

Ejemplo para grupo de proxy de Zabbix:

```
ServerActive=proxy1.example.com;proxy2.example.com;proxy3.example.com;proxy4.example.com;proxy5.example.co
```

Ejemplo para varios servers:

```
ServerActive=127.0.0.1:20051,zabbix.domain,[::1]:30051,::1,[12fc::1]
```

Ejemplo para alta disponibilidad:

```
ServerActive=zabbix.cluster.node1;zabbix.cluster.node2:20051;zabbix.cluster.node3
```

Ejemplo para alta disponibilidad con dos clústeres y un server:

```
ServerActive=zabbix.cluster.node1;zabbix.cluster.node2:20051,zabbix.cluster2.node1;zabbix.cluster2.node2,z
```

Rango: El número de servers activos enumerados en `ServerActive` más el número de instancias pre-fork para comprobaciones pasivas especificadas en `StartAgents` debe ser menor que 64.

SourceIP

Especifique la dirección IP de origen para:

- conexiones salientes a Zabbix server o Zabbix proxy;
- realizar conexiones al ejecutar algunos items (`web.page.get`, `net.tcp.port`, etc.).

StartAgents

Establece el número de instancias pre-fork de `zabbix_agentd` para procesar comprobaciones pasivas.
 Si se establece en 0, las comprobaciones pasivas se deshabilitan y el agent no escuchará en ningún puerto TCP.

Predeterminado: 10
 Rango: 0-100 (*)

Timeout

Especifique cuánto tiempo esperar (en segundos) para establecer la conexión e intercambiar datos con Zabbix proxy o server.

Este parámetro define la duración de varias operaciones de comunicación, entre ellas:

- esperar una respuesta del Zabbix server;
- enviar solicitudes al Zabbix server, incluidas las solicitudes de configuración de item y los datos de item en active checks;
- recuperar datos de registro mediante la supervisión de logfile o del registro de eventos de Windows;
- enviar mensajes de latido;
- duración máxima para comprobaciones `vfs.*`;
- utilizado por los módulos de Zabbix agent;
- servir como alternativa en escenarios en los que un server o proxy anterior a la versión 7.0 envía comprobaciones sin timeouts.

Este timeout **no** se utilizará para aquellas comprobaciones de agent que tienen configuraciones de timeout ajustables en el frontend (a nivel global, de proxy o por item).

Predeterminado: 3
 Rango: 1-30

TLSAccept

Especifique el nivel de cifrado requerido para las conexiones entrantes.
 Se usa para comprobaciones pasivas.
 Se pueden especificar varios valores, separados por comas: - *unencrypted* - aceptar conexiones sin cifrado (predeterminado); - *psk* - aceptar conexiones con TLS y una clave previamente compartida (PSK); - *cert* - aceptar conexiones con TLS y un certificado.

Obligatorio: sí, si se definen parámetros de certificado TLS o PSK (incluso para una conexión *unencrypted*); en caso contrario, no

TLSCAFile

Especifique la ruta completa del archivo que contiene los certificados de la CA de nivel superior para la verificación del certificado del par, utilizados para las comunicaciones cifradas entre los componentes de Zabbix.

TLSCertFile

Especifique la ruta completa del archivo que contiene el certificado del agent o la cadena de certificados, utilizado para las comunicaciones cifradas con los componentes de Zabbix.

TLSConnect

Especifique el nivel de cifrado requerido para las conexiones del agent con Zabbix server o proxy.
 Se usa para comprobaciones activas.
 Solo se puede especificar un valor: - *unencrypted* - conectar sin cifrado (predeterminado); - *psk* - conectar usando TLS y una clave precompartida (PSK); - *cert* - conectar usando TLS y un certificado.

Obligatorio: sí, si se definen parámetros de certificado TLS o PSK (incluso para una conexión *unencrypted*); en caso contrario, no

TLSCRLFile

Especifique la ruta completa del archivo que contiene los certificados revocados. Este parámetro se utiliza para las comunicaciones cifradas entre los componentes de Zabbix.

TLSKeyFile

Especifique la ruta completa del archivo que contiene la clave privada del agent, utilizada para las comunicaciones cifradas entre los componentes de Zabbix.

TLSPSKFile

Especifique la ruta completa del archivo que contiene la clave precompartida del agent, utilizada para comunicaciones cifradas con el server o proxy de Zabbix.

TLSPSKIdentity

Especifique la cadena de identidad de la clave precompartida, utilizada para comunicaciones cifradas con Zabbix server o proxy.

TLSServerCertIssuer

Especifique el emisor del certificado del server (proxy) permitido.

TLSServerCertSubject

Especifique el subject del certificado permitido del server (proxy).

UnsafeUserParameters

Permite que todos los caracteres se pasen como argumentos a parámetros definidos por el usuario. No se permiten los siguientes caracteres: \ ' " ` * ? [ ] { } ~ $! & ; ( ) < > | # @ %
Además, no se permiten caracteres de nueva línea.

Predeterminado: 0
 Valores: 0 - no permitir, 1 - permitir

UserParameter

Especifique un parámetro definido por el usuario para supervisar. Puede haber varios parámetros definidos por el usuario.
Formato: `UserParameter=<key>,<shell command>`
Tenga en cuenta que el comando de shell no debe devolver una cadena vacía ni solo EOL. Los comandos de shell pueden tener rutas relativas si se especifica el parámetro `UserParameterDir`.

Ejemplo:

```
UserParameter=system.test,dir /b | find /c /v ""
UserParameter=check_cpu,.\custom\_script.bat
```

UserParameterDir

Especifica la ruta de búsqueda predeterminada para los comandos `UserParameter`. Si se utiliza, el agent cambiará su directorio de trabajo al especificado aquí antes de ejecutar un comando. De este modo, los comandos `UserParameter` pueden usar un prefijo relativo `./` en lugar de una ruta completa. Solo se permite una entrada.

Ejemplo:

```
UserParameterDir=C:\opt\myscripts
```

Zabbix agent en Microsoft Windows

Configuración del agent

Ambas generaciones de agents de Zabbix se ejecutan como un servicio de Windows.
Para Zabbix agent 2, sustituya `agentd` por `agent2` en las instrucciones siguientes.

Puede ejecutar una única instancia de Zabbix agent o varias instancias del agent en un host de Microsoft Windows.
Una única instancia puede utilizar una de las siguientes opciones:

- el archivo de configuración predeterminado, ubicado en el mismo directorio que el binario del agent;
- un archivo de configuración especificado en la línea de comandos.

En el caso de varias instancias, cada instancia del agent debe tener su propio archivo de configuración (una de las instancias puede utilizar el archivo de configuración predeterminado).

Hay un archivo de configuración de ejemplo disponible en el archivo de código fuente de Zabbix como:

- `conf/zabbix_agentd.conf` para Zabbix agent;
- `conf/zabbix_agent2.conf` para Zabbix agent2.

Si desea instalar Zabbix agent/agent 2 para Windows como un servicio desde un archivo sin especificar explícitamente el archivo de configuración, antes de instalar el agent:

- `conf/zabbix_agentd.conf` debe copiarse manualmente en el directorio donde se instalará `zabbix_agentd.exe`;
- `conf/zabbix_agent2.conf` y el directorio `conf/zabbix_agent2.d` deben copiarse manualmente en el directorio donde se instalará `zabbix_agent2.exe`.

Consulte las opciones del archivo de configuración del agent de Zabbix o del archivo de configuración de Zabbix agent 2 para obtener más información sobre la configuración del agent de Zabbix para Windows.

Parámetro Hostname

Para realizar comprobaciones activas en un host, el Zabbix agent debe tener definido el nombre del host. Además, el valor del nombre del host establecido en el lado del agent debe coincidir exactamente con el *Nombre del host* configurado para el host en el frontend.

El valor del nombre del host en el lado del agent se puede definir mediante el parámetro `Hostname` o `HostnameItem` en el archivo de configuración del agent; si no se especifica ninguno de estos parámetros, se utilizan los valores predeterminados.

El valor predeterminado del parámetro `HostnameItem` es el valor devuelto por la clave de agent `system.hostname`. En Windows, esta clave devuelve el resultado de la función `gethostname()`, que consulta los proveedores de espacio de nombres para determinar el nombre del host local. Si ningún proveedor de espacio de nombres responde, se devuelve el nombre NetBIOS.

El valor predeterminado de `Hostname` es el valor devuelto por el parámetro `HostnameItem`. Por lo tanto, si no se especifica ninguno de estos parámetros, el nombre real del host será el nombre NetBIOS del host; el Zabbix agent utilizará el nombre NetBIOS del host para recuperar la lista de comprobaciones activas del Zabbix server y enviarle los resultados.

La clave `system.hostname` admite dos parámetros opcionales: *type* y *transform*.

Type determina el tipo de nombre que debe devolver el item:

- *netbios* (predeterminado): devuelve el nombre NetBIOS del host, limitado a 15 caracteres y compuesto únicamente por letras MAYÚSCULAS;
- *host*: distingue entre mayúsculas y minúsculas; devuelve el nombre completo y real del host de Windows (sin el dominio);
- *shorthost*: devuelve la parte del nombre del host anterior al primer punto. Devuelve la cadena completa si el nombre no contiene ningún punto.
- *fqdn*: devuelve el nombre de dominio completo (Fully Qualified Domain Name), sin el punto final.

Transform permite especificar una regla de transformación adicional para el nombre del host:

- *none* (predeterminado): utiliza el uso original de mayúsculas y minúsculas;
- *lower*: convierte el texto a minúsculas.

Por lo tanto, para simplificar la configuración del archivo `zabbix_agentd.conf` y hacerla uniforme, se pueden utilizar distintos enfoques:

1. Dejar sin definir los parámetros `Hostname` o `HostnameItem`; el Zabbix agent utilizará el nombre NetBIOS del host como nombre del host.
2. Dejar sin definir el parámetro `Hostname` y definir `HostnameItem` de la siguiente manera:
   - `HostnameItem=system.hostname[host]`: para que el Zabbix agent utilice el nombre completo y real del host de Windows, distinguiendo entre mayúsculas y minúsculas, como nombre del host.

- `HostnameItem=system.hostname[shorthost,lower]`: para que el Zabbix agent utilice únicamente la parte del nombre del host anterior al primer punto, convertida a minúsculas.
- `HostnameItem=system.hostname[fqdn]`: para que el Zabbix agent utilice el nombre de dominio completo (Fully Qualified Domain Name) como nombre del host.

El nombre del host también se utiliza como parte del nombre del servicio de Windows, que se emplea para instalar, iniciar, detener y desinstalar el servicio de Windows. Por ejemplo, si el archivo de configuración del Zabbix agent especifica `Hostname=Windows_db_server`, el agent se instalará como el servicio de Windows `Zabbix Agent [Windows_db_server]`. Por lo tanto, para tener un nombre de servicio de Windows diferente para cada instancia del Zabbix agent, cada instancia debe utilizar un nombre de host diferente.

Instalación de agent como servicio de Windows

Antes de instalar el agent, copie manualmente `conf/zabbix_agentd.conf` al directorio donde se instalará zabbix_agentd.exe.

Para instalar una única instancia de Zabbix agent con el archivo de configuración predeterminado:

```
zabbix_agentd.exe --install
```

Attention:
En un sistema de 64 bits, se requiere una versión de 64 bits de Zabbix agent para que todas las comprobaciones relacionadas con la ejecución de procesos de 64 bits funcionen correctamente.

Si desea usar un archivo de configuración distinto del predeterminado, debe usar el siguiente comando para instalar el servicio:

```
zabbix_agentd.exe --config <your_configuration_file> --install
```

Debe especificarse una ruta completa al archivo de configuración.

Se pueden instalar varias instancias de Zabbix agent como servicios de esta manera:

```
zabbix_agentd.exe --config <configuration_file_for_instance_1> --install --multiple-agents
zabbix_agentd.exe --config <configuration_file_for_instance_2> --install --multiple-agents
...
zabbix_agentd.exe --config <configuration_file_for_instance_N> --install --multiple-agents
```

El servicio instalado ahora debería ser visible en el Panel de control.

Inicio de agent

Para iniciar el servicio de agent, puede usar el Panel de control o hacerlo desde la línea de comandos.

Para iniciar una única instancia de Zabbix agent con el archivo de configuración predeterminado:

```
zabbix_agentd.exe --start
```

Para iniciar una única instancia de Zabbix agent con otro archivo de configuración:

```
zabbix_agentd.exe --config <your_configuration_file> --start
```

Para iniciar una de varias instancias de Zabbix agent:

```
zabbix_agentd.exe --config <configuration_file_for_this_instance> --start --multiple-agents
```

Detener agent

Para detener el servicio de agent, puede usar el Panel de control o hacerlo desde la línea de comandos.

Para detener una única instancia de Zabbix agent iniciada con el archivo de configuración predeterminado:

```
zabbix_agentd.exe --stop
```

Para detener una única instancia de Zabbix agent iniciada con otro archivo de configuración:

```
zabbix_agentd.exe --config <your_configuration_file> --stop
```

Para detener una de varias instancias de Zabbix agent:

```
zabbix_agentd.exe --config <configuration_file_for_this_instance> --stop --multiple-agents
```

Desinstalación del servicio de Windows de agent

Para desinstalar una única instancia de Zabbix agent usando el archivo de configuración predeterminado:

```
zabbix_agentd.exe --uninstall
```

Para desinstalar una única instancia de Zabbix agent usando un archivo de configuración no predeterminado:

```
zabbix_agentd.exe --config <your_configuration_file> --uninstall
```

Para desinstalar varias instancias de Zabbix agent de los servicios de Windows:

```
zabbix_agentd.exe --config <configuration_file_for_instance_1> --uninstall --multiple-agents
zabbix_agentd.exe --config <configuration_file_for_instance_2> --uninstall --multiple-agents
...
zabbix_agentd.exe --config <configuration_file_for_instance_N> --uninstall --multiple-agents
```

Limitaciones

Zabbix agent para Windows no admite configuraciones de Windows no estándar en las que las CPU se distribuyen de forma no uniforme entre nodos NUMA. Si las CPU lógicas se distribuyen de forma no uniforme, es posible que las métricas de rendimiento de la CPU no estén disponibles para algunas CPU. Por ejemplo, si hay 72 CPU lógicas con 2 nodos NUMA, ambos nodos deben tener 36 CPU cada uno.

Zabbix agent en IBM AIX

Descripción general

Los binarios precompilados del Zabbix agent para IBM AIX están disponibles para descarga como un archivo `.tar.gz`. Esta página describe cómo instalar el agent desde el archivo, ejecutarlo manualmente o con el System Resource Controller (SRC), iniciarlo durante el arranque del sistema y eliminarlo de SRC.

Instalación del agent

El comando `tar` de AIX no admite la opción `-z`, por lo que debe descomprimir el archivo con `gunzip`. Utilice la opción `-o` de `tar` para que los archivos extraídos pertenezcan al usuario que los extrae, y no a los ID numéricos de usuario y grupo almacenados en el archivo. Por ejemplo, para colocar el agent en `/opt/zabbix` (ejecute como root; `/opt/zabbix` no debe existir todavía):

```
gunzip -c zabbix_agent-7.4.7-aix-7.2-powerpc-openssl.tar.gz | tar -xof -
mv zabbix_agent-7.4.7-aix-7.2-powerpc-openssl /opt/zabbix
```

En el archivo de AIX, el binario del agent se encuentra en `usr/sbin/` y el archivo de configuración, en `etc/zabbix/`. Antes de iniciar el agent, ajuste `/opt/zabbix/etc/zabbix/zabbix_agentd.conf` según sea necesario (como mínimo, los parámetros `Server`, `ServerActive` y `Hostname`).

Si se inicia como root, el agent cambia al usuario `zabbix` (o al usuario especificado por el parámetro `User`), a menos que se establezca `AllowRoot=1` en el archivo de configuración. Este usuario debe existir en el sistema; de lo contrario, el agent no se inicia. Si el usuario `zabbix` aún no existe, créelo junto con un grupo del mismo nombre:

```
mkgroup zabbix
mkuser pgrp=zabbix login=false rlogin=false zabbix
```

Los atributos `login=false` y `rlogin=false` impiden que la cuenta se utilice para iniciar sesión; el agent aún puede ejecutarse como este usuario.

Iniciar el agent manualmente

El agent se puede iniciar manualmente:

```
/opt/zabbix/usr/sbin/zabbix_agentd -c /opt/zabbix/etc/zabbix/zabbix_agentd.conf
```

Ejecución del agent con SRC

Registro del agent

El agent también se puede registrar con el System Resource Controller (SRC). Cuando se inicia mediante SRC, el agent debe ejecutarse en modo de primer plano (la opción `-f`). De lo contrario, SRC pierde el seguimiento del proceso del agent y no puede detenerlo. Por ejemplo (ejecutado como root):

```
mkssys \
  -s zabbix_agentd \
  -p /opt/zabbix/usr/sbin/zabbix_agentd \
  -a "-c /opt/zabbix/etc/zabbix/zabbix_agentd.conf -f" \
  -u zabbix \
  -G tcpip \
  -S -n 15 -f 9 \
  -R
```

En este ejemplo:

- `-s` establece el nombre del subsistema utilizado con los comandos de SRC;
- `-p` establece la ruta completa al binario del agent;
- `-a` establece los argumentos de la línea de comandos del agent como una única cadena entre comillas; SRC divide esta cadena en argumentos independientes cuando inicia el agent (solo se acepta una opción `-a`);
- `-u` establece el usuario con el que se ejecuta el agent; el nombre de usuario se convierte en un ID de usuario numérico cuando se crea el subsistema, por lo que el usuario ya debe existir;
- `-G` añade el agent al grupo de subsistemas `tcpip`, de modo que los comandos del grupo también se aplican a él (por ejemplo, `stopsrc -g tcpip` también detiene el agent);
- `-S -n 15 -f 9` hace que SRC controle el agent mediante señales: `stopsrc` envía `SIGTERM` (15) para una detención normal y `SIGKILL` (9) para una detención forzada (`stopsrc -f`);
- `-R` hace que SRC reinicie el agent si termina de forma anómala (por ejemplo, si uno de sus procesos se bloquea); el agent no se reinicia después de `stopsrc`.

Dado que el ID de usuario se almacena en la definición del subsistema, actualice la definición si el usuario `zabbix` se vuelve a crear con un ID de usuario diferente:

```
chssys -s zabbix_agentd -u zabbix
```

Inicio y detención del agent

El agent se puede iniciar, consultar y detener con `startsrc`, `lssrc` y `stopsrc`:

```
startsrc -s zabbix_agentd
lssrc -s zabbix_agentd
stopsrc -s zabbix_agentd
```

Evite la detención forzada (`stopsrc -f`). Esta solo finaliza el proceso principal del agent; los demás procesos del agent siguen ejecutándose y mantienen abierto el puerto de escucha, por lo que el agent no se puede iniciar de nuevo hasta que se finalicen manualmente (por ejemplo, con `kill -9`).

Iniciar el agent durante el arranque del sistema

Registrar el agent con SRC no hace que se inicie durante el arranque del sistema. Para iniciar el agent automáticamente durante el arranque, añada una entrada a `/etc/inittab`:

```
mkitab "zabbix_agentd:2:once:/usr/bin/startsrc -s zabbix_agentd >/dev/console 2>&1"
```

Eliminación del agent de SRC

Para eliminar el agent del inicio del sistema y de SRC:

```
stopsrc -s zabbix_agentd
rmitab zabbix_agentd
rmssys -s zabbix_agentd
```

Agent 2

Resumen Zabbix agent 2 es una nueva generación de Zabbix agent, escrita en Go (con algo de código C reutilizado de Zabbix agent). Está diseñado para:

- Reducir el número de conexiones TCP;
- Proporcionar una mejor concurrencia de comprobaciones;
- Ampliarse fácilmente con plugins que proporcionan comprobaciones simples con un código mínimo y admiten comprobaciones complejas compuestas por scripts de larga duración y recopilación de datos independiente con informes periódicos;
- Funcionar como reemplazo de Zabbix agent, admitiendo todas las funciones anteriores.

Todas las métricas en Zabbix agent 2 son recopiladas por plugins.

Estas funciones son específicas de los plugins:

- Intervalos programados y flexibles para comprobaciones pasivas y activas
- Gestión de la cola de tareas con respecto a la programación y la concurrencia de tareas
- Tiempos de espera a nivel de plugin
- Comprobación de compatibilidad de Zabbix agent 2 y sus plugins al inicio

Existen dos tipos de plugins:

- Plugins integrados

- Plugins cargables

También puede crear sus propios plugins.

Los plugins creados de forma personalizada no necesitan estar compilados dentro de agent 2. Puede agregarlos como plugins cargables, simplificando así la creación de plugins para nuevas métricas.

Los plugins cargables pueden añadirse en tiempo de ejecución sin recompilar Zabbix agent 2.

Véase también:

- Comparación entre Agent y agent 2
- Parámetros de configuración: Zabbix agent 2
- Parámetros de configuración: Zabbix agent 2 (Windows)
- Parámetros de configuración: Plugins
- Protocolo de plugins de Zabbix agent 2

Comprobaciones pasivas y activas

Zabbix agent 2 admite comprobaciones pasivas y activas, al igual que Zabbix agent. Además, las comprobaciones activas de Zabbix agent 2 admiten intervalos flexibles/programados y concurrencia de comprobaciones dentro de un mismo server activo.

Note:
De forma predeterminada, después de un reinicio, Zabbix agent 2 programará la primera recopilación de datos para las comprobaciones activas en un momento aleatorio condicionado dentro del intervalo de actualización del item, con el fin de evitar picos en el uso de recursos. Para realizar inmediatamente después del reinicio del agent las comprobaciones activas que no tengan un intervalo de actualización de *Programación*, establezca el parámetro `ForceActiveChecksOnStart` (nivel global) o el parámetro del plugin `Plugins.<Plugin name>.System.ForceActiveChecksOnStart` (que afecta únicamente a las comprobaciones de un plugin específico) en el archivo de configuración. Si se establece el parámetro a nivel de plugin, este tendrá prioridad sobre el parámetro global.

Concurrencia de comprobaciones

Las comprobaciones de diferentes plugins pueden ejecutarse de forma concurrente. El número de comprobaciones concurrentes dentro de un mismo plugin está limitado por la configuración de capacidad del plugin. Cada plugin puede tener una configuración de capacidad codificada de forma fija (1000 como valor predeterminado) que puede reducirse mediante la configuración `Plugins.<PluginName>.System.Capacity=N` en el parámetro de configuración `Plugins`.

Plataformas compatibles Para conocer las plataformas compatibles, consulte la página Requirements.

Agent 2 en sistemas tipo Unix Zabbix agent 2 en sistemas tipo Unix se ejecuta en el host que se está supervisando.

Instalación

Zabbix agent 2 se puede instalar en sistemas basados en Linux utilizando uno de los siguientes métodos:

- Paquetes de Zabbix - seleccione el componente Agent 2 (después de seleccionar su versión de Zabbix, la distribución del SO y la versión del SO) y siga las instrucciones;
- Fuentes de Zabbix - descargue los archivos fuente y compile el agent mediante la configuración con la opción `--enable-agent2`.

Note:
Las capacidades de monitorización de Zabbix agent 2 se pueden ampliar con plugins cargables, que están disponibles por separado. Para más detalles, consulte Plugins cargables.

Si está instalado como paquete

Zabbix agent 2 se ejecuta como un proceso en primer plano y depende de un gestor de servicios externo (por ejemplo, systemd) para su ejecución en segundo plano. Zabbix agent 2 no tiene compatibilidad integrada con la daemonización en Linux.

El agent se puede iniciar ejecutando:

```
systemctl start zabbix-agent2
```

Para detener, reiniciar o comprobar el estado de Zabbix agent 2, use los siguientes comandos:

```
systemctl stop zabbix-agent2
systemctl restart zabbix-agent2
systemctl status zabbix-agent2
```

Iniciar manualmente

Puede iniciar Zabbix agent localizando el binario `zabbix_agent2` y ejecutándolo directamente; por ejemplo:

```
zabbix_agent2
```

Agent 2 en sistemas Windows Zabbix agent 2 se ejecuta como un proceso independiente; sin embargo, también puede ejecutarse como un servicio de Windows.

Instalación

Zabbix agent 2 se puede instalar en Windows utilizando uno de los siguientes métodos:

- Binarios precompilados de Zabbix agent - descargue el paquete instalador MSI del agent y siga las instrucciones de la página Instalación de Zabbix agent en Windows desde MSI;
- Fuentes de Zabbix - descargue los archivos fuente y siga las instrucciones de la página Compilación de Zabbix agent 2 en Windows.

Note:
Las capacidades de monitoreo de Zabbix agent 2 se pueden ampliar con plugins cargables, que están disponibles por separado. Para más detalles, consulte Plugins cargables.

Consulte la página Zabbix agent en Microsoft Windows para obtener detalles adicionales sobre la instalación de Zabbix agent 2 (desde un archivo ZIP) como servicio de Windows.

Opciones Los siguientes parámetros de línea de comandos se pueden utilizar con Zabbix agent 2:

Parámetro	Descripción
agent para UNIX y Windows	
`-c --config <config-file>`	Ruta al archivo de configuración. Puede utilizar esta opción para especificar un archivo de configuración distinto del predeterminado.
`-f --foreground`	Ejecutar Zabbix agent en primer plano (valor predeterminado: true).
`-p --print`	Mostrar los items conocidos y salir. Tenga en cuenta que, para devolver también los resultados de los parámetros de usuario, debe especificar el archivo de configuración (si no se encuentra en la ubicación predeterminada).
`-t --test <item key>`	Probar el item especificado y salir. Tenga en cuenta que, para devolver también los resultados de los parámetros de usuario, debe especificar el archivo de configuración (si no se encuentra en la ubicación predeterminada).
`-T --test-config`	Validar el archivo de configuración y salir.
`-h --help`	Mostrar la información de ayuda y salir.
`-v --verbose`	Mostrar información de depuración. Utilice esta opción con las opciones `-p` y `-t`.
`-V --version`	Mostrar la versión del agent y la información de licencia.
`-R --runtime-control <option>`	Realizar funciones administrativas. Consulte control en tiempo de ejecución.
Solo agent para Windows	
`-m --multiple-agents`	Utilizar varias instancias del agent (con las opciones `-i`, `-d`, `-s`, `-x`). Para distinguir los nombres de servicio de las instancias, cada nombre de servicio incluirá el valor Hostname del archivo de configuración especificado.
`-S --startup-type <value>`	Establecer el tipo de inicio del servicio de Zabbix agent para Windows. Valores permitidos: `automatic` - *(predeterminado)* iniciar el servicio automáticamente al iniciar Windows; `delayed` - retrasar el inicio del servicio hasta que los servicios iniciados automáticamente hayan completado su inicio; `manual` - iniciar el servicio manualmente (por un usuario o una aplicación); `disabled` - deshabilitar el servicio para que no pueda ser iniciado por un usuario o una aplicación. Puede utilizar esta opción junto con la opción `-i` o por separado para modificar el tipo de inicio de un servicio ya instalado.
`-i --install`	Instalar Zabbix agent para Windows como servicio.
`-d --uninstall`	Desinstalar el servicio de Zabbix agent para Windows.
`-s --start`	Iniciar el servicio de Zabbix agent para Windows.

Parámetro	Descripción
`-x --stop`	Detener el servicio de Zabbix agent para Windows.

Ejemplos específicos del uso de parámetros de línea de comandos:

- Mostrar todos los items integrados del agent con sus valores.
- Probar un parámetro de usuario con la clave `mysql.ping` definida en el archivo de configuración especificado.
- Instalar un servicio de Zabbix Agent para Windows utilizando la ruta predeterminada al archivo de configuración.
- Modificar el tipo de inicio de un servicio de Zabbix Agent para Windows instalado utilizando el archivo de configuración ubicado en la misma carpeta que el ejecutable del agent.

```
zabbix_agent2 --print
zabbix_agent2 -t "mysql.ping" -c /etc/zabbix/zabbix_agent2.conf
zabbix_agent2.exe -i
zabbix_agent2.exe -c zabbix_agent2.conf -S delayed
```

Control en tiempo de ejecución

El control en tiempo de ejecución proporciona algunas opciones para el control remoto.

Opción	Descripción
`log_level_increase`	Aumentar el nivel de registro.
`log_level_decrease`	Disminuir el nivel de registro.
`metrics`	Enumerar las métricas disponibles.
`periodic_prof_enable`	Habilitar la creación de perfiles periódica. Todas las métricas recopilan datos en el momento y los vuelcan, excepto los datos de CPU. Los datos periódicos de CPU se recopilan a lo largo del tiempo en un archivo temporal oculto. Cuando finaliza el intervalo, el archivo se publica como un perfil de CPU y se conserva el número configurado de archivos completados. El archivo temporal no cuenta para `ProfilerMaxFilesPerProfile`.
`periodic_prof_disable`	Deshabilitar la creación de perfiles periódica.
`periodic_prof_execute`	Escribir perfiles después de un muestreo de 5 segundos. Cuando la creación de perfiles periódica está habilitada, este comando finaliza el intervalo de CPU actual, escribe un perfil de CPU independiente de 5 segundos e inicia un nuevo archivo de CPU periódico. Los perfiles solicitados explícitamente se rotan según `ProfilerMaxFilesPerProfile` cuando finaliza el comando. Mientras este comando recopila perfiles, todos los comandos de tiempo de ejecución del profiler se rechazan con "profiler is busy".
`periodic_prof_set_interval`	Establecer el intervalo de creación de perfiles.
`version`	Mostrar la versión del agent.
`userparameter_reload`	Volver a cargar los valores de las opciones `UserParameter` e `Include` desde el archivo de configuración actual.
`help`	Mostrar información de ayuda sobre el control en tiempo de ejecución.

Ejemplos:

- Aumentar el nivel de registro del agent 2.
- Imprimir las opciones de control en tiempo de ejecución.

```
zabbix_agent2 -R log_level_increase
zabbix_agent2 -R help
```

Archivo de configuración La ruta predeterminada al archivo de configuración depende del sistema operativo y del método de instalación de Zabbix:

- En UNIX, la ruta predeterminada es:
  - Para la instalación desde las fuentes: `/usr/local/etc/zabbix_agent2.conf`, o la definida mediante las variables de tiempo de compilación `--sysconfdir` o `--prefix`.
  - Para la instalación desde paquetes: `/etc/zabbix/zabbix_agent2.conf`.
- En Windows, la ruta predeterminada es `C:\Program Files\Zabbix Agent 2\zabbix_agent2.conf`.

Si se está ejecutando una única instancia del agent en un host, puede utilizar el archivo de configuración predeterminado o un archivo de configuración especificado en la línea de comandos. En caso de varias instancias, cada instancia del agent debe tener su propio archivo de configuración (una de las instancias puede utilizar el archivo de configuración predeterminado).

Los parámetros de configuración de Zabbix agent 2 coinciden en su mayoría con los de Zabbix agent, con algunas excepciones:

Parameters		Description
Parámetros de configuración específicos de Zabbix agent 2		
	`ControlSocket`	La ruta del socket de control en tiempo de ejecución. Zabbix agent 2 utiliza un socket de control para los comandos en tiempo de ejecución.
	`EnablePersistentBuffer`, `PersistentBufferFile`, `PersistentBufferPeriod`	Estos parámetros se utilizan para configurar el almacenamiento persistente en Zabbix agent 2 para los items activos.
	`ForceActiveChecksOnStart`	Determina si el agent debe realizar comprobaciones activas inmediatamente después de reiniciarse o distribuirlas uniformemente a lo largo del tiempo.
	`Plugins`	Los plugins pueden tener sus propios parámetros, con el formato `Plugins.<Plugin name>.<Parameter>=<value>`. Un parámetro común de los plugins es `System.Capacity`, que establece el límite de comprobaciones que se pueden ejecutar simultáneamente.
	`StatusPort`	El puerto en el que Zabbix agent 2 escuchará las solicitudes de estado HTTP y mostrará una lista de plugins configurados y algunos parámetros internos.
Parámetros de configuración específicos de Zabbix agent no compatibles con Zabbix agent 2		
	`AllowRoot`, `User`	No son compatibles con Zabbix agent 2 porque no admite la ejecución como daemon.
	`EnableRemoteCommands`	Obsoleto; en ambos agents se utilizan en su lugar los parámetros `AllowKey`, `AllowKeyRegexp`, `DenyKey` y `DenyKeyRegexp`.

Parameters	Description
ListenBacklog	Se utiliza en Zabbix agent para definir el número máximo de conexiones pendientes en la cola TCP. No es compatible con Zabbix agent 2.
LoadModule, LoadModulePath	Los módulos cargables no son compatibles con Zabbix agent 2.
MaxLinesPerSecond, LogRemoteCommands	En Zabbix agent 2 se utilizan en su lugar Plugins.Log.MaxLinesPerSecond y Plugins.SystemRun.LogRemoteCommands.
StartAgents	Este parámetro se utiliza en Zabbix agent para aumentar la concurrencia de las comprobaciones pasivas o deshabilitarlas. En Zabbix agent 2, la concurrencia se configura en el nivel del plugin y puede limitarse mediante un ajuste de capacidad. Las comprobaciones pasivas están deshabilitadas en Zabbix Agent 2 si no se especifica el parámetro Server.

Para obtener más información, consulte las opciones del archivo de configuración de Zabbix agent2.

Códigos de salida Zabbix agent 2 también puede compilarse con versiones anteriores de OpenSSL (1.0.1, 1.0.2).

En este caso, Zabbix proporciona mutexes para el bloqueo en OpenSSL. Si falla el bloqueo o desbloqueo de un mutex, se imprime un mensaje de error en el flujo de error estándar (STDERR) y Agent 2 finaliza con el código de retorno 2 o 3, respectivamente.

Comparación entre Agent y agent 2

Esta sección describe las diferencias entre Zabbix agent y Zabbix agent 2.

Parameter	Zabbix agent	Zabbix agent 2
Lenguaje de programación	C	Go con algunas partes en C
Daemonización	sí	solo mediante systemd (sí en Windows)
Extensiones compatibles	Módulos cargables personalizados loadable modules en C.	plugins personalizados en Go.
Requisitos		
Plataformas compatibles	Linux, IBM AIX, FreeBSD, NetBSD, OpenBSD, HP-UX, Mac OS X, Solaris: 9, 10, 11, Windows: todas las versiones de escritorio y server desde XP	Linux, Windows: todas las versiones de escritorio y server, en las que se pueda instalar una versión compatible de Go.
Bibliotecas criptográficas compatibles	GnuTLS 3.1.18 y posteriores. OpenSSL 1.0.1, 1.0.2, 1.1.0, 1.1.1, 3.0.x. LibreSSL - probado con las versiones 2.7.4, 2.8.2 (se aplican ciertas limitaciones; consulte la página Encryption para obtener más detalles).	Linux: OpenSSL 1.0.1 y posteriores. MS Windows: OpenSSL 1.1.1 o posterior. La biblioteca OpenSSL debe tener habilitado el soporte PSK. LibreSSL no es compatible.
Procesos de monitoreo		
Procesos	Un proceso de active check independiente para cada registro de server/proxy.	Un único proceso con hilos creados automáticamente. El número máximo de hilos lo determina la variable de entorno GOMAXPROCS.

Parameter	Zabbix agent	Zabbix agent 2
Métricas	**UNIX**: consulte una lista de items compatibles. **Windows**: consulte una lista de items adicionales específicos de Windows.	**UNIX**: Todas las métricas compatibles con Zabbix agent. Además, agent 2 proporciona una solución de monitoreo nativa de Zabbix para: Docker, Memcached, MySQL, PostgreSQL, Redis, systemd y otros objetivos de monitoreo; consulte una lista completa de items específicos de agent 2. **Windows**: Todas las métricas compatibles con Zabbix agent, y también comprobaciones `net.tcp.service*` de HTTPS y LDAP. Además, agent 2 proporciona una solución de monitoreo nativa de Zabbix para: PostgreSQL, Redis.
Concurrencia	Las active checks para un solo server se ejecutan secuencialmente.	Las comprobaciones de diferentes plugins o varias comprobaciones dentro de un mismo plugin pueden ejecutarse de forma concurrente.
Traps de terceros	no	sí
Funciones adicionales		
Almacenamiento persistente	no	sí
Archivos persistentes para métricas `log*[]`	sí (solo en Unix)	no
Carga de datos de log	Puede realizarse durante la recopilación de logs para liberar el búfer.	La recopilación de logs se detiene cuando el búfer está lleno; por lo tanto, el parámetro `BufferSize` debe ser al menos `MaxLinesPerSecond` x 2.
Cambia de usuario en tiempo de ejecución	sí (solo en sistemas tipo Unix)	no (controlado por systemd)
Cifrados configurables por el usuario	sí	no

Véase también:

- *Descripción de procesos de Zabbix*: Zabbix agent, Zabbix agent 2
- *Parámetros de configuración*: Zabbix agent UNIX / Windows, Zabbix agent 2 UNIX / Windows

Plugins integrados

Estos plugins están integrados en Zabbix agent 2 y están disponibles de forma inmediata. Haga clic en el nombre del plugin para ir al repositorio del plugin con información adicional.

Nombre del plugin	Descripción	Claves de item compatibles	Comentarios
Agent	Métricas del Zabbix agent que se está utilizando.	agent.hostname, agent.ping, agent.version	Las claves compatibles tienen los mismos parámetros que las claves de Zabbix agent.
CPU	Supervisión de la CPU del sistema (número de CPU/núcleos de CPU, CPU detectadas, porcentaje de utilización).	system.cpu.discovery, system.cpu.num, system.cpu.util	Las claves compatibles tienen los mismos parámetros que las claves de Zabbix agent.

Nombre del plugin	Descripción	Claves de item compatibles	Comentarios
Docker	Supervisión de contenedores Docker.	docker.container_info, docker.container_stats, docker.containers, docker.containers.discovery, docker.data_usage, docker.images, docker.images.discovery, docker.info, docker.ping	Véase también: Parámetros de configuración
File	Recopilación de métricas de archivos.	vfs.file.cksum, vfs.file.contents, vfs.file.exists, vfs.file.md5sum, vfs.file.regexp, vfs.file.regmatch, vfs.file.size, vfs.file.time	Las claves compatibles tienen los mismos parámetros que las claves de Zabbix agent.
Kernel	Supervisión del kernel.	kernel.maxfiles, kernel.maxproc	Las claves compatibles tienen los mismos parámetros que las claves de Zabbix agent.
Log	Supervisión de archivos de registro.	log, log.count, logrt, logrt.count	Las claves compatibles tienen los mismos parámetros que las claves de Zabbix agent. Véase también: Parámetros de configuración del plugin (Unix/Windows)
Memcached	Supervisión del server Memcached.	memcached.ping, memcached.stats	
Modbus	Lee datos Modbus.	modbus.get	Las claves compatibles tienen los mismos parámetros que las claves de Zabbix agent.
MQTT	Recibe los valores publicados de los temas MQTT.	mqtt.get	Para configurar una conexión cifrada con el broker MQTT, especifique los parámetros TLS en el archivo de configuración del agent como una sesión con nombre o como parámetros predeterminados. Actualmente, los parámetros TLS no se pueden pasar como parámetros de la clave del item.
MySQL	Supervisión de MySQL y sus derivados.	mysql.custom.query, mysql.db.discovery, mysql.db.size, mysql.get_status_variables, mysql.ping, mysql.replication.discovery, mysql.replication.get_slave_status, mysql.version	Para configurar una conexión cifrada con la base de datos, especifique los parámetros TLS en el archivo de configuración del agent como una sesión con nombre o como parámetros predeterminados. Actualmente, los parámetros TLS no se pueden pasar como parámetros de la clave del item.
NetIf	Supervisión de interfaces de red.	net.if.collisions, net.if.discovery, net.if.get, net.if.in, net.if.out, net.if.total	Las claves compatibles tienen los mismos parámetros que las claves de Zabbix agent.

Nombre del plugin	Descripción	Claves de item compatibles	Comentarios
Oracle	Supervisión de Oracle Database.	oracle.diskgroups.stats, oracle.diskgroups.discovery, oracle.archive.info, oracle.archive.discovery, oracle.cdb.info, oracle.custom.query, oracle.datafiles.stats, oracle.db.discovery, oracle.fra.stats, oracle.instance.info, oracle.pdb.info, oracle.pdb.discovery, oracle.pga.stats, oracle.ping, oracle.proc.stats, oracle.redolog.info, oracle.sga.stats, oracle.sessions.stats, oracle.sys.metrics, oracle.sys.params, oracle.ts.stats, oracle.ts.discovery, oracle.user.info, oracle.version	Instale Oracle Instant Client antes de usar el plugin.
Proc	Porcentaje de utilización de CPU de un proceso.	proc.cpu.util	La clave compatible tiene los mismos parámetros que la clave de Zabbix agent.
Redis	Supervisión del server Redis.	redis.config, redis.info, redis.ping, redis.slowlog.count	Para configurar una conexión cifrada con Redis, especifique los parámetros TLS en el archivo de configuración del agent como una sesión con nombre o como parámetros predeterminados. Los parámetros TLS no se pueden pasar como parámetros de la clave del item. Tenga en cuenta que una configuración TLS incorrecta o no válida puede impedir que Zabbix agent 2 se inicie, por lo que debe verificar los archivos de certificados, los permisos y las rutas antes de habilitar TLS.

Smart	Supervisión S.M.A.R.T.	smart.attribute.discovery, smart.disk.discovery, smart.disk.get	La versión mínima requerida de smartctl es 7.1.
			Se requieren privilegios de sudo/root sobre smartctl para el usuario que ejecuta Zabbix agent 2. El plugin solo utiliza los siguientes comandos:
			`/usr/sbin/smartctl -a *`
			`/usr/sbin/smartctl --scan *`
			`/usr/sbin/smartctl -j -V`
			
			Las claves compatibles solo se pueden usar con Zabbix agent 2 en Linux/Windows, tanto como comprobación pasiva como activa.
			Véase también:
			Parámetros de configuración

SW	Listado de paquetes instalados.	system.sw.packages, system.sw.packages.get	Las claves compatibles tienen los mismos parámetros que la clave de Zabbix agent.
Swap	Tamaño del espacio de swap en bytes/porcentaje.	system.swap.size	La clave compatible tiene los mismos parámetros que la clave de Zabbix agent.

Nombre del plugin	Descripción	Claves de item compatibles	Comentarios
SystemRun	Ejecuta el comando especificado.	system.run	La clave compatible tiene los mismos parámetros que la clave de Zabbix agent. Véase también: Parámetros de configuración del plugin (Unix/Windows)
Systemd	Supervisión de servicios systemd.	systemd.unit.discovery, systemd.unit.get, systemd.unit.info	
TCP	Comprobación de disponibilidad de conexiones TCP.	net.tcp.port	La clave compatible tiene los mismos parámetros que la clave de Zabbix agent.
UDP	Supervisión de la disponibilidad y el rendimiento de los servicios UDP.	net.udp.service, net.udp.service.perf	Las claves compatibles tienen los mismos parámetros que las claves de Zabbix agent.
Uname	Obtención de información sobre el sistema.	system.hostname, system.sw.arch, system.uname	Las claves compatibles tienen los mismos parámetros que las claves de Zabbix agent.
Uptime	Recopilación de métricas de tiempo de actividad del sistema.	system.uptime	La clave compatible tiene los mismos parámetros que la clave de Zabbix agent.
VFSDev	Recopilación de métricas VFS.	vfs.dev.discovery, vfs.dev.get, vfs.dev.read, vfs.dev.write	Las claves compatibles tienen los mismos parámetros que las claves de Zabbix agent.
WebCertificate	Supervisión de certificados TLS/SSL de sitios web.	web.certificate.get	
WebPage	Supervisión de páginas web.	web.page.get, web.page.perf, web.page.regexp	Las claves compatibles tienen los mismos parámetros que las claves de Zabbix agent.
ZabbixAsync	Recopilación asíncrona de métricas.	net.tcp.listen, net.udp.listen, sensor, system.boottime, system.cpu.intr, system.cpu.load, system.cpu.switches, system.hw.cpu, system.hw.macaddr, system.localtime, system.sw.os, system.swap.in, system.swap.out, vfs.fs.discovery	Las claves compatibles tienen los mismos parámetros que las claves de Zabbix agent.
ZabbixStats	Métricas internas de Zabbix server/proxy o número de items retrasados en una cola.	zabbix.stats	Las claves compatibles tienen los mismos parámetros que las claves de Zabbix agent.

Nombre del plugin	Descripción	Claves de item compatibles	Comentarios
ZabbixSync	Recopilación síncrona de métricas.	net.dns, net.dns.record, net.tcp.service, net.tcp.service.perf, proc.mem, proc.num, system.hw.chassis, system.hw.devices, system.sw.packages, system.users.num, vfs.dir.count, vfs.dir.size, vfs.fs.get, vfs.fs.inode, vfs.fs.size, vm.memory.size.	Las claves compatibles tienen los mismos parámetros que las claves de Zabbix agent.

Plugins cargables

Los plugins cargables no están disponibles de forma predeterminada para Zabbix agent 2 y deben instalarse por separado:

- En Linux, puede utilizar paquetes (por ejemplo, instalar Ember+ en Ubuntu con `apt install zabbix-agent2-plugin-ember-plus`) o compilar plugins
- En Windows, puede instalarlos desde MSI o compilar plugins

Attention:
Antes de instalar un plugin, consulte su archivo README. Puede contener requisitos específicos e instrucciones de instalación.

Haga clic en el nombre del plugin para ir al repositorio del plugin, que contiene el archivo README del plugin con información adicional.

Nombre del plugin	Descripción	Claves de item compatibles	Comentarios
Ceph	Monitorización de Ceph.	ceph.df.details, ceph.osd.stats, ceph.osd.discovery, ceph.osd.dump, ceph.ping, ceph.pool.discovery, ceph.status	Consulte también Parámetros de configuración del plugin de Ceph.
Ember+	Monitorización de Ember+.	ember.get	Consulte también Parámetros de configuración del plugin de Ember+.

Nombre del plugin	Descripción	Claves de item compatibles	Comentarios
MongoDB	Monitorización de servidores y clústeres MongoDB (base de datos distribuida basada en documentos).	mongodb.cfg.discovery, mongodb.collection.stats, mongodb.collections.discovery, mongodb.collections.usage, mongodb.connpool.stats, mongodb.db.stats, mongodb.db.discovery, mongodb.jumbo_chunks.count, mongodb.oplog.stats, mongodb.ping, mongodb.rs.config, mongodb.rs.status, mongodb.server.status, mongodb.sh.discovery, mongodb.version	Para configurar conexiones cifradas a la base de datos, especifique los parámetros TLS en el archivo de configuración del agent como parámetros de una sesión con nombre. Actualmente, los parámetros TLS no se pueden pasar como parámetros de la clave del item. Consulte también Parámetros de configuración del plugin de MongoDB.
MSSQL	Monitorización de la base de datos MSSQL.	mssql.availability.group.get, mssql.custom.query, mssql.db.get, mssql.job.status.get, mssql.last.backup.get, mssql.local.db.get, mssql.mirroring.get, mssql.nonlocal.db.get, mssql.perfcounter.get, mssql.ping, mssql.quorum.get, mssql.quorum.member.get, mssql.replica.get, mssql.version	Para configurar una conexión cifrada a la base de datos, especifique los parámetros TLS en el archivo de configuración del agent como parámetros de una sesión con nombre o predeterminados. Actualmente, los parámetros TLS no se pueden pasar como parámetros de la clave del item. Consulte también Parámetros de configuración del plugin de MSSQL.
NVIDIA GPU	Monitorización de GPU NVIDIA.	nvml.device.count, nvml.device.decoder.utilization, nvml.device.ecc.mode, nvml.device.encoder.stats.get, nvml.device.encoder.utilization, nvml.device.energy.consumption, nvml.device.errors.memory, nvml.device.errors.register, nvml.device.fan.speed.avg, nvml.device.get, nvml.device.graphics.frequency, nvml.device.memory.bar1.get, nvml.device.memory.fb.get, nvml.device.memory.frequency, nvml.device.pci.utilization, nvml.device.performance.state, nvml.device.power.limit, nvml.device.power.usage, nvml.device.serial, nvml.device.sm.frequency, nvml.device.temperature, nvml.device.utilization, nvml.device.video.frequency, nvml.system.driver.version, nvml.version	Consulte también Parámetros de configuración del plugin de NVIDIA GPU.

Nombre del plugin	Descripción	Claves de item compatibles	Comentarios
PostgreSQL	Monitorización de PostgreSQL y sus derivados.	pgsql.autovacuum.count, pgsql.archive, pgsql.bgwriter, pgsql.cache.hit, pgsql.connections, pgsql.custom.query, pgsql.dbstat, pgsql.dbstat.sum, pgsql.db.age, pgsql.db.bloating_tables, pgsql.db.discovery, pgsql.db.size, pgsql.locks, pgsql.oldest.xid, pgsql.ping, pgsql.queries, pgsql.replication.count, pgsql.replication.process, pgsql.replication.process.discovery, pgsql.replication.recovery_role, pgsql.replication.status, pgsql.replication_lag.b, pgsql.replication_lag.sec, pgsql.uptime, pgsql.version, pgsql.wal.stat	Para configurar conexiones cifradas a la base de datos, especifique los parámetros TLS en el archivo de configuración del agent como parámetros de una sesión con nombre o predeterminados. Actualmente, los parámetros TLS no se pueden pasar como parámetros de la clave del item. Consulte también Parámetros de configuración del plugin de PostgreSQL.

Note:
Plugins cargables, al iniciarse con:
`-V --version`: muestran la versión del plugin y la información de licencia;
`-h --help`: muestran información de ayuda.
`-t, --test <item key>`: inician el plugin para realizar pruebas (se ignora la configuración del plugin).

Parámetros de configuración: Zabbix agent 2

Descripción general

Los parámetros de configuración permiten personalizar el comportamiento de Zabbix agent 2.

Zabbix agent 2 es una nueva generación de Zabbix agent y puede utilizarse en lugar de Zabbix agent.

Especifique los valores de los parámetros en el archivo de configuración de agent 2 de Zabbix (`zabbix_agent2.conf`). Todos los parámetros son **opcionales**, a menos que se indique explícitamente que el parámetro es obligatorio.

Esta página está estructurada de la siguiente manera:

- Descripción general de los parámetros (para consultar todos los detalles de un parámetro, haga clic en su nombre)
- Detalles de los parámetros

Parámetro	Descripción
Alias	Establece un alias para una clave de item.
AllowKey	Permite la ejecución de las claves de item que coincidan con un patrón.
AllowKeyRegexp	Permite la ejecución de las claves de item que coincidan con un patrón basado en una expresión regular.
BufferSend	Establece la frecuencia de envío de valores del búfer al server/proxy de Zabbix.
BufferSize	Establece el número máximo de valores en el búfer de memoria.
ControlSocket	Especifica el socket de control para enviar comandos en tiempo de ejecución con la opción '-R'.
DebugLevel	Establece el nivel de depuración.
DenyKey	Deniega la ejecución de las claves de item que coincidan con un patrón.
DenyKeyRegexp	Deniega la ejecución de las claves de item que coincidan con un patrón basado en una expresión regular.

Parámetro	Descripción
EnablePersistentBuffer	Habilita el uso de almacenamiento persistente local para los items activos.
EnableProfiler	Habilita la creación periódica de perfiles de Go al iniciar el agent.
ForceActiveChecksOnStart	Realiza comprobaciones activas inmediatamente después del reinicio para la primera configuración recibida.
HeartbeatFrequency	Establece la frecuencia de los mensajes de latido en segundos.
HostInterface	Especifica la interfaz del host.
HostInterfaceItem	Especifica un item para obtener la interfaz del host.
HostMetadata	Especifica los metadatos del host.
HostMetadataItem	Especifica un item de Zabbix agent para obtener los metadatos del host.
Hostname	Especifica el nombre del host.
HostnameItem	Especifica un item de Zabbix agent para obtener el nombre del host.
Include	Especifica un directorio o archivos individuales que se incluirán en el archivo de configuración.
ListenIP	Especifica la dirección IP de escucha del agent para las conexiones desde el server/proxy.
ListenPort	Especifica el puerto de escucha del agent.
LogFile	Especifica el archivo de registro del agent.
LogFileSize	Establece el tamaño máximo del archivo de registro.
LogType	Especifica el tipo de salida del registro.
PersistentBufferFile	Especifica el archivo donde se conservará la base de datos SQLite.
PersistentBufferPeriod	Establece el período durante el cual se almacenarán los datos cuando no haya conexión con el server o proxy de Zabbix.
PidFile	Especifica el archivo PID.
Plugins.<PluginName>.System.Capacity	Establece el límite de comprobaciones por plugin que pueden ejecutarse simultáneamente.
Plugins.Log.MaxLinesPerSecond	Establece el número máximo de líneas nuevas por segundo que se enviarán al server/proxy de Zabbix al procesar comprobaciones activas `log` o `logrt`.
Plugins.SystemRun.LogRemoteCommands	Habilita el registro de los comandos de shell ejecutados como advertencias.
PluginSocket	Especifica la ruta al socket UNIX para las comunicaciones con plugins cargables.
PluginTimeout	Establece el tiempo de espera para las conexiones con plugins cargables.
ProfilerDir	El directorio donde se escribirán los datos de creación de perfiles.
ProfilerMaxFilesPerProfile	El número máximo de archivos que se conservarán para cada tipo de perfil.
ProfilerInterval	El intervalo de escritura de los datos de creación de perfiles.
RefreshActiveChecks	Especifica la frecuencia de actualización de la lista de comprobaciones activas.
Server	Especifica las direcciones IP o los nombres DNS de los servers/proxies de Zabbix.
ServerActive	Especifica la dirección del server/proxy de Zabbix o la configuración del clúster desde la que se obtendrán las comprobaciones activas.
SourceIP	Especifica la dirección IP de origen.
StatusPort	Especifica el puerto de escucha de las solicitudes de estado HTTP (`http://localhost:<port>/status`).
Timeout	Especifica cuánto tiempo se debe esperar (en segundos) para establecer la conexión e intercambiar datos con el server o proxy de Zabbix.
TLSAccept	Especifica el nivel de cifrado requerido para las conexiones entrantes.
TLSCAFile	Especifica la ruta completa del archivo que contiene los certificados de las CA de nivel superior para la verificación del certificado del par, utilizados para las comunicaciones cifradas entre los componentes de Zabbix.
TLSCertFile	Especifica la ruta completa del archivo que contiene el certificado o la cadena de certificados del agent, utilizados para las comunicaciones cifradas entre los componentes de Zabbix.
TLSCipherAll	Especifica la cadena de cifrado de OpenSSL (TLS 1.2). Anula los criterios de selección predeterminados del conjunto de cifrado para el cifrado basado en certificados y PSK.
TLSCipherAll13	Especifica la cadena de cifrado de OpenSSL (TLS 1.3) para OpenSSL 1.1.1 o posterior. Anula los criterios de selección predeterminados del conjunto de cifrado para el cifrado basado en certificados y PSK.
TLSCipherCert	Especifica la cadena de cifrado de OpenSSL (TLS 1.2). Anula los criterios de selección predeterminados del conjunto de cifrado para el cifrado basado en certificados.
TLSCipherCert13	Especifica la cadena de cifrado de OpenSSL (TLS 1.3) para OpenSSL 1.1.1 o posterior. Anula los criterios de selección predeterminados del conjunto de cifrado para el cifrado basado en certificados.
TLSCipherPSK	Especifica la cadena de cifrado de OpenSSL (TLS 1.2). Anula los criterios de selección predeterminados del conjunto de cifrado para el cifrado basado en PSK.
TLSCipherPSK13	Especifica la cadena de cifrado de OpenSSL (TLS 1.3) para OpenSSL 1.1.1 o posterior. Anula los criterios de selección predeterminados del conjunto de cifrado para el cifrado basado en PSK.
TLSConnect	Especifica el nivel de cifrado requerido para las conexiones con el server o proxy de Zabbix.
TLSCRLFile	Especifica la ruta completa del archivo que contiene los certificados revocados. Este parámetro se utiliza para las comunicaciones cifradas entre los componentes de Zabbix.

Parámetro	Descripción
TLSKeyFile	Especifica la ruta completa del archivo que contiene la clave privada del agent, utilizada para las comunicaciones cifradas entre los componentes de Zabbix.
TLSPSKFile	Especifica la ruta completa del archivo que contiene la clave precompartida del agent, utilizada para las comunicaciones cifradas con el server o proxy de Zabbix.
TLSPSKIdentity	Especifica la cadena de identidad de la clave precompartida, utilizada para las comunicaciones cifradas con el server o proxy de Zabbix.
TLSServerCertIssuer	Especifica el emisor permitido del certificado del server (proxy).
TLSServerCertSubject	Especifica el sujeto permitido del certificado del server (proxy).
UnsafeUserParameters	Permite pasar todos los caracteres en los argumentos de los parámetros definidos por el usuario.
UserParameter	Especifica un parámetro definido por el usuario que se supervisará.
UserParameterDir	Especifica la ruta de búsqueda predeterminada para los comandos UserParameter.

Tenga en cuenta lo siguiente:

- Los valores predeterminados reflejan los valores predeterminados del proceso, no los valores de los archivos de configuración distribuidos.
- Los valores admiten variables de entorno.
- Zabbix admite archivos de configuración únicamente con codificación UTF-8 y sin BOM.
- Los comentarios que comienzan con # solo se admiten al principio de la línea.

Detalles del parámetro

Alias

Establezca un alias para una clave de item. Puede utilizarse para sustituir una clave de item larga y compleja por otra más corta y sencilla.
 Puede haber varios parámetros `Alias`. No se permiten varios parámetros con la misma clave `Alias`.
 Diferentes claves `Alias` pueden hacer referencia a la misma clave de item.
 Los alias pueden utilizarse en el parámetro `HostMetadataItem`, pero no en `HostnameItem`.

Ejemplo 1: Obtener el ID del usuario `zabbix`.

```
Alias=zabbix.userid:vfs.file.regexp[/etc/passwd,"^zabbix:.:([0-9]+)",,,,\1]
```

Ahora se puede utilizar la clave abreviada `zabbix.userid` para obtener datos.

Ejemplo 2: Obtener el uso de la CPU con parámetros predeterminados y personalizados.

```
Alias=cpu.util:system.cpu.util
Alias=cpu.util[*]:system.cpu.util[*]
```

Esto permite utilizar la clave `cpu.util` para obtener el porcentaje de uso de la CPU con los parámetros predeterminados, así como utilizar `cpu.util[all, idle, avg15]` para obtener datos específicos sobre el uso de la CPU.

Ejemplo 3: Ejecutar varias reglas de descubrimiento de bajo nivel que procesan los mismos items de descubrimiento.

```
Alias=vfs.fs.discovery[*]:vfs.fs.discovery
```

Ahora es posible configurar varias reglas de descubrimiento utilizando `vfs.fs.discovery` con diferentes parámetros para cada regla, por ejemplo, `vfs.fs.discovery[foo]`, `vfs.fs.discovery[bar]`, etc.

AllowKey

Permite la ejecución de aquellas claves de item que coinciden con un patrón. El patrón de clave es una expresión comodín que admite el carácter * para coincidir con cualquier número de caracteres.
Se pueden definir varias reglas de coincidencia de claves en combinación con `DenyKey`. Se pueden definir varias reglas de coincidencia de claves en combinación con `DenyKey`, `AllowKeyRegexp` y `DenyKeyRegexp`. Los parámetros se procesan uno por uno según el orden en que aparecen. El orden determina qué regla coincidente tendrá prioridad. Véase también: Restricting agent checks.

AllowKeyRegexp

Permite la ejecución de los item keys que coincidan con un patrón.
El patrón de la key es una expresión regular.
 Se pueden definir varias reglas de coincidencia de keys en combinación con `DenyKeyRegexp`, `AllowKey` y `DenyKey`. Los parámetros se procesan uno por uno según el orden en que aparecen. El orden determina qué regla coincidente tendrá prioridad. Consulte también: Restricción de las comprobaciones del agent.

Ejemplo:

```
AllowKeyRegexp=^system\.run\[(?:/usr)?/bin/hostname(?: -[fsd])?\]$
```

BufferSend

Establece la frecuencia de envío de valores desde el búfer al Zabbix server/proxy (en segundos).
 Tenga en cuenta que, si el búfer está lleno, los datos se enviarán antes.

Predeterminado: 5
 Rango: 1-3600

BufferSize

Establece el número máximo de valores en el búfer de memoria.
 El agent enviará todos los datos recopilados al server de Zabbix o al proxy si el búfer está lleno. Este parámetro solo debe usarse si el búfer persistente está deshabilitado (`EnablePersistentBuffer=0`).

Predeterminado: 1000
 Rango: 2-65535

ControlSocket

Especifica el socket de control para enviar comandos en tiempo de ejecución con la opción `-R`.

Predeterminado: `/tmp/agent.sock`

DebugLevel

Establezca el nivel de depuración: - *0* - información básica sobre el inicio y la detención de los procesos de Zabbix; - *1* - información crítica; - *2* - información de error; - *3* - advertencias; - *4* - para depuración (genera mucha información); - *5* - depuración ampliada (genera aún más información).

Predeterminado: 3
 Rango: 0-5

DenyKey

Deniega la ejecución de los item keys que coincidan con un patrón.
 El patrón de la key es una expresión comodín que admite el carácter `*` para coincidir con cualquier cantidad de caracteres.
Se pueden definir varias reglas de coincidencia de keys en combinación con `AllowKey`. Se pueden definir varias reglas de coincidencia de keys en combinación con `AllowKey`, `AllowKeyRegexp` y `DenyKeyRegexp`. Los parámetros se procesan uno por uno según el orden en que aparecen. El orden determina qué regla de coincidencia tendrá prioridad. Véase también: Restricción de las comprobaciones del agent.

Predeterminado: `DenyKey=system.run[*]`

DenyKeyRegexp

Deniega la ejecución de los item keys que coincidan con un patrón.
El patrón de la key es una expresión regular.
 Se pueden definir varias reglas de coincidencia de keys en combinación con `AllowKeyRegexp`, `AllowKey` y `DenyKey`. Los parámetros se procesan uno por uno según el orden en que aparecen. El orden determina qué regla coincidente tendrá prioridad.
 Todos los items `system.run[]` se deniegan de forma predeterminada (`DenyKey=system.run[*]`), a menos que se especifique otra regla `system.run[*]`.

Consulte también: Restringir las comprobaciones del agent.

Ejemplo:

```
DenyKeyRegexp=^system\.run\[.*(?:rm -rf|poweroff|reboot).*\]$
```

EnablePersistentBuffer

Habilita el uso de almacenamiento local persistente para los items activos.
 Si el almacenamiento persistente está deshabilitado, se usará el búfer de memoria.

Predeterminado: 0
 Valores: 0 - deshabilitado, 1 - habilitado

EnableProfiler

Habilita el perfilado periódico de Go al iniciar el agent. El perfilado también se puede habilitar o deshabilitar durante la ejecución mediante la opción `-R`. Mientras `periodic_prof_execute` recopila perfiles, todos los comandos de ejecución del profiler se rechazan porque está ocupado.

Valor predeterminado: 0
 Valores: 0 - deshabilitado, 1 - habilitado

ForceActiveChecksOnStart

Realiza comprobaciones activas inmediatamente después del reinicio para la primera configuración recibida.
 También está disponible como parámetro de configuración por plugin, por ejemplo: `Plugins.Uptime.System.ForceActiveChecksOnStart=1`

Predeterminado: 0
 Valores: 0 - deshabilitado, 1 - habilitado

HeartbeatFrequency

Establece la frecuencia de los mensajes de heartbeat en segundos.
 Se utiliza para supervisar la disponibilidad de las comprobaciones activas.
0 - mensajes de heartbeat deshabilitados.

Predeterminado: 60
 Rango: 0-3600

HostInterface

Especifique la interfaz del host (dirección IP o nombre DNS) utilizada durante el autoregistro del host.
 Este valor se utilizará para rellenar la interfaz en el host recién creado y permite configurar explícitamente una dirección IP o DNS.
 Para más detalles, consulte Using DNS as default interface.

Si no se define, el valor se obtendrá de `HostInterfaceItem`.

El agent mostrará un error y no se iniciará si el valor supera el límite de 255 caracteres.

Rango: 0-255 caracteres

HostInterfaceItem

Especifique un item para determinar la interfaz del host (dirección IP o nombre DNS) durante el registro automático del host.
 Este valor se utiliza únicamente si `HostInterface` no está definido.
 Para obtener más información, consulte Uso de DNS como interfaz predeterminada.

Durante una solicitud de registro automático, el agent registrará un mensaje de advertencia si el valor devuelto por el item especificado supera el límite de 255 caracteres.

El item `system.run[]` es compatible independientemente de la configuración de `AllowKey`, `AllowKeyRegexp`, `DenyKey` o `DenyKeyRegexp`.

HostMetadata

Especifique los metadatos utilizados para identificar o distinguir el host durante la autoregistración del host.
 `HostMetadata` permite distinguir entre hosts más allá del nombre del host.

Si no se define, el valor se obtendrá de `HostMetadataItem`.

El agent generará un error y no se iniciará si el valor especificado supera el límite de 2034 bytes o es una cadena que no está codificada en UTF-8. Cuando un parámetro espera una dirección IP o un nombre DNS, los valores que son UTF-8 válido, pero no son direcciones IP ni nombres DNS válidos, también se rechazarán y se notificarán como no válidos.

No se admiten metadatos multilínea: la salida se truncará en el primer salto de línea.

Rango: 0-2034 bytes

HostMetadataItem

Especifique un item para obtener los metadatos del host.
 Esta opción solo se utiliza cuando `HostMetadata` no está definido.

El valor de `HostMetadataItem` se obtiene en cada intento de autoregistro y se utiliza únicamente durante el proceso de autoregistro del host. `HostMetadataItem` permite distinguir entre hosts más allá del nombre del host.

Se admiten parámetros de usuario y alias. El item `system.run[]` se admite independientemente de la configuración de `AllowKey`, `AllowKeyRegexp`, `DenyKey` o `DenyKeyRegexp`.

Durante una solicitud de autoregistro, el agent registrará un mensaje de advertencia si el valor devuelto por el item especificado supera el límite de 65535 puntos de código UTF-8. El valor devuelto por el item debe ser una cadena UTF-8; de lo contrario, se ignorará. Cuando un parámetro espera una dirección IP o un nombre DNS, también se rechazarán y se notificarán como no válidos los valores que sean UTF-8 válido, pero no sean IP ni nombres DNS válidos.

No se admiten metadatos multilínea: la salida se truncará en el primer salto de línea.

Nombre de host

Especifique nombres de host únicos y sensibles a mayúsculas y minúsculas. Una lista delimitada por comas.
 Requerido para comprobaciones activas y debe coincidir con los nombres de host configurados en el server.
 El valor se obtiene de `HostnameItem` si no está definido.
Caracteres permitidos: alfanuméricos, '.', ' ', '_' y '-'. Longitud máxima: 128 caracteres por nombre de host, 2048 caracteres para toda la línea.

Predeterminado: establecido por HostnameItem

HostnameItem

Especifique un item para obtener el nombre del host.
 Esta opción solo se utiliza cuando `Hostname` no está definido.
 No se admiten parámetros de usuario ni alias, pero el item `system.run[]` es compatible independientemente de los valores de `AllowKey`, `AllowKeyRegexp`, `DenyKey` o `DenyKeyRegexp`.

Valor predeterminado: `system.hostname`

Include

Especifique un directorio o archivos individuales que se incluirán en el archivo de configuración.
 Durante la instalación de Zabbix, este parámetro se establece en `/usr/local/etc`, a menos que cambie esto durante el tiempo de compilación.

Reglas generales para el parámetro `Include`:

- La ruta puede ser relativa a la ubicación del archivo `zabbix_agent2.conf`.
- Puede especificar varios parámetros `Include` en un solo archivo de configuración.
- Zabbix procesa el parámetro `Include` en cuanto lo lee, así que considere colocarlo al final del archivo de configuración.

Reglas para los archivos incluidos:

- Cada archivo debe ser legible.
- Cada archivo debe tener una sintaxis correcta. De lo contrario, el componente no se iniciará.

Reglas para incluir directorios o patrones de archivo:

- No asuma ningún orden de inclusión (por ejemplo, los archivos no se incluyen en orden alfabético).
- No defina el mismo parámetro en más de un archivo incluido (por ejemplo, para sobrescribir una configuración general con una específica), ya que no puede predecir qué valor se utilizará.
- Tenga cuidado con las copias de seguridad que algunos editores de texto crean automáticamente (por ejemplo, editar `include/my_specific.conf` puede crear `include/my_specific.conf.BAK`), ya que ambos archivos se incluirán.

Para incluir solo los archivos relevantes de un directorio, use el comodín (`*`) para la coincidencia de patrones.

Ejemplo:

```
Include=/absolute/path/to/config/files/*.conf
```

ListenIP

Especifique la dirección IP de escucha del agent para conexiones desde el server/proxy. Una lista delimitada por comas.
 La primera dirección IP se envía al server de Zabbix, si se conecta a él, para recuperar la lista de comprobaciones activas.

Predeterminado: `0.0.0.0`

ListenPort

Especifique el puerto de escucha del agent para conexiones desde el server/proxy.

Predeterminado: 10050
 Rango: 1024-32767

LogFile

Especifique el archivo de registro del agent.

Predeterminado: `/tmp/zabbix_agent2.log`
 Obligatorio: Sí, si `LogType` está establecido en `file`; de lo contrario, no

LogFileSize

Establezca el tamaño máximo del archivo de registro (en MB).
0 - deshabilita la rotación automática de registros.
Nota: Si se alcanza el límite de tamaño del archivo de registro y la rotación del archivo falla, por cualquier motivo, el archivo de registro existente se trunca y se inicia uno nuevo.

Predeterminado: 1
 Rango: 0-1024

LogType

Especifique el tipo de salida del registro: - *file* - escribe el registro en el archivo especificado por el parámetro `LogFile`; - *system* - escribe el registro en syslog; - *console* - escribe el registro en la salida estándar.

Predeterminado: `file`

PersistentBufferFile

Especifique el archivo para conservar la base de datos SQLite. Debe ser un nombre de archivo completo.
 Este parámetro solo se usa si el búfer persistente está habilitado (`EnablePersistentBuffer=1`).

PersistentBufferPeriod

Establece el período de tiempo para almacenar datos cuando no hay conexión con el server o proxy de Zabbix. Los datos más antiguos se perderán.
 Los datos de registro se conservarán. Este parámetro solo se usa si el búfer persistente está habilitado (`EnablePersistentBuffer=1`).

Predeterminado: 1h
 Rango: 1m-365d

PidFile

Especifique el archivo PID.

Predeterminado: `/tmp/zabbix_agent2.pid`

Plugins.<PluginName>.System.Capacity

Establece el límite de comprobaciones por el plugin `<PluginName>` que pueden ejecutarse al mismo tiempo.

Predeterminado: 1000 Rango: 1-1000

Plugins.Log.MaxLinesPerSecond

Establece el número máximo de nuevas líneas por segundo para enviar al server/proxy de Zabbix al procesar comprobaciones activas `log` o `logrt`.
 El valor proporcionado se sobrescribirá con el parámetro `maxlines`, incluido en la clave del item `log` y `logrt`.
Nota: Zabbix procesará 10 veces más nuevas líneas que las establecidas en `MaxLinesPerSecond` para buscar la cadena requerida en los items de log.

Predeterminado: 20
 Rango: 1-1000

Plugins.SystemRun.LogRemoteCommands

Habilita el registro de los comandos de shell ejecutados como advertencias.
 Los comandos se registrarán solo si se ejecutan de forma remota.
 No se crearán entradas de registro si `system.run[]` se inicia localmente mediante los parámetros `HostMetadataItem`, `HostInterfaceItem` o `HostnameItem`.

Predeterminado: 0
 Valores: 0 - deshabilitado, 1 - habilitado

PluginSocket

Especifique la ruta al socket UNIX para las comunicaciones del plugin cargable.

Predeterminado: `/tmp/agent.plugin.sock`

PluginTimeout

Especifica el tiempo de espera para las conexiones a plugins cargables, en segundos.

Predeterminado: `Timeout`
 Rango: 1-30

ProfilerDir

El directorio donde se escribirán los datos de perfilado.

Valor predeterminado: `/tmp/zabbix_agent2_profiles`

ProfilerMaxFilesPerProfile

El número máximo de archivos que se conservarán para cada tipo de perfil. El límite se aplica a los archivos completados después de cada volcado del perfil. El perfil de CPU activo se recopila en un archivo temporal oculto y no cuenta para el límite. Los archivos creados por `periodic_prof_execute` también están sujetos al límite.

Valor predeterminado: 10
 Rango: 1-100

ProfilerInterval

El intervalo de escritura de los datos de perfil, en segundos.

Valor predeterminado: 3600
 Rango: 1-86400

RefreshActiveChecks

Especifique la frecuencia de actualización de la lista de comprobaciones activas, en segundos.
 Tenga en cuenta que, después de fallar la actualización de las comprobaciones activas, el siguiente intento se realizará en 60 segundos.

Predeterminado: 5
 Rango: 1-86400

Server

Especifique las direcciones IP, opcionalmente en notación CIDR, o los nombres DNS de los servers/proxies de Zabbix. Una lista delimitada por comas.
 Las conexiones entrantes solo se aceptarán desde los hosts enumerados aquí. Si la compatibilidad con IPv6 está habilitada, entonces `127.0.0.1`, `::127.0.0.1`, `::ffff:127.0.0.1` se tratan por igual y `::/0` permitirá cualquier dirección IPv4 o IPv6. `0.0.0.0/0` se puede usar para permitir cualquier dirección IPv4. Se permiten espacios. Si este parámetro no se especifica, deshabilita las comprobaciones pasivas y el agent no escuchará en ningún puerto TCP.

Ejemplo:

```
Server=127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.example.com
```

ServerActive

Especifique la dirección del server/proxy de Zabbix o la configuración del clúster para obtener comprobaciones activas.
 La dirección del server/proxy es una dirección IP o un nombre DNS y, opcionalmente, un puerto separado por dos puntos.
La configuración del clúster es una o más direcciones de miembros de un grupo de server o proxy separadas por punto y coma. Se pueden especificar varios servers/clústeres de Zabbix y proxies de Zabbix, separados por comas. A menos que se utilicen grupos de proxy, no se debe especificar más de un proxy de Zabbix para cada server/clúster de Zabbix. Si se especifica un proxy de Zabbix, entonces no se debe especificar el server/clúster de Zabbix para ese proxy.
Se pueden proporcionar varias direcciones separadas por comas para usar varios servers de Zabbix independientes en paralelo. Se permiten espacios.
Si no se especifica el puerto, se usa el puerto predeterminado.
Las direcciones IPv6 deben ir entre corchetes si se especifica el puerto para ese host. Si no se especifica el puerto, los corchetes para direcciones IPv6 son opcionales.
Si no se especifica este parámetro, las comprobaciones activas se deshabilitan.

Ejemplo para proxy de Zabbix:

```
ServerActive=127.0.0.1:10051
```

Ejemplo para grupo de proxy de Zabbix:

```
ServerActive=proxy1.example.com;proxy2.example.com;proxy3.example.com;proxy4.example.com;proxy5.example.co
```

Ejemplo para varios servers:

```
ServerActive=127.0.0.1:20051,zabbix.domain,[::1]:30051,::1,[12fc::1]
```

Ejemplo para alta disponibilidad:

```
ServerActive=zabbix.cluster.node1;zabbix.cluster.node2:20051;zabbix.cluster.node3
```

Ejemplo para alta disponibilidad con dos clústeres y un server:

```
ServerActive=zabbix.cluster.node1;zabbix.cluster.node2:20051,zabbix.cluster2.node1;zabbix.cluster2.node2,z
```

SourceIP

Especifique la dirección IP de origen para:

- conexiones salientes a Zabbix server o Zabbix proxy.
- realizar conexiones al ejecutar algunos items (`web.page.get`, `net.tcp.port`, etc.).

StatusPort

Especifique el puerto para escuchar solicitudes de estado HTTP (`http://localhost:<port>/status`).

Rango: 1024-32767

Tiempo de espera

Especifique cuánto tiempo esperar (en segundos) para establecer la conexión e intercambiar datos con Zabbix proxy o server.

Este parámetro define la duración de varias operaciones de comunicación, incluidas:

- esperar una respuesta del Zabbix server;
- enviar solicitudes a Zabbix server, incluidas las solicitudes de configuración de item y los datos de item en active checks;
- recuperar datos de registro a través de logfile;
- enviar mensajes de latido;
- duración máxima para comprobaciones `vfs.*`;
- servir como valor de reserva en escenarios en los que un server o proxy anterior a la versión 7.0 envía comprobaciones sin tiempos de espera.

Este tiempo de espera **no** se utilizará para aquellas comprobaciones de agent que tengan configuraciones de tiempo de espera configurables en el frontend (a nivel global, de proxy o por item).

Predeterminado: 3
 Rango: 1-30

TLSAccept

Especifique el nivel de cifrado requerido para las conexiones entrantes.
 Se usa para comprobaciones pasivas.
 Se pueden especificar varios valores, separados por comas: - *unencrypted* - aceptar conexiones sin cifrado (predeterminado); - *psk* - aceptar conexiones con TLS y una clave precompartida (PSK); - *cert* - aceptar conexiones con TLS y un certificado.

Obligatorio: sí, si se definen parámetros de certificado TLS o PSK (incluso para una conexión *unencrypted*); en caso contrario, no

TLSCAFile

Especifique la ruta completa del archivo que contiene los certificados de la CA de nivel superior para la verificación del certificado del par, utilizado para las comunicaciones cifradas entre los componentes de Zabbix.

TLSCertFile

Especifique la ruta completa del archivo que contiene el certificado del agent o la cadena de certificados, utilizado para comunicaciones cifradas con los componentes de Zabbix.

TLSCipherAll

Especifique la cadena de cifrado de OpenSSL (TLS 1.2). Anule los criterios de selección predeterminados del conjunto de cifrado para el cifrado basado en certificados y PSK.

Ejemplo:

```
EECDH+aRSA+AES128:RSA+aRSA+AES128:kECDHEPSK+AES128:kPSK+AES128
```

TLSCipherAll13

Especifique la cadena de cifrado de OpenSSL (TLS 1.3) para OpenSSL 1.1.1 o posterior. Anule los criterios de selección predeterminados de la suite de cifrado para el cifrado basado en certificados y PSK.

Ejemplo:

```
TLS_AES_256_GCM_SHA384:TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
```

TLSCipherCert

Especifique la cadena de cifrado de OpenSSL (TLS 1.2). Anule los criterios de selección predeterminados del conjunto de cifrados para el cifrado basado en certificados.

Ejemplo:

```
EECDH+aRSA+AES128:RSA+aRSA+AES128
```

Tenga en cuenta que este parámetro no puede usarse junto con `TLSAccept=cert,psk`; para una conexión con certificado (`TLSConnect=cert`), use `TLSCipherAll` en su lugar.

TLSCipherCert13

Especifique la cadena de cifrado de OpenSSL (TLS 1.3) para OpenSSL 1.1.1 o una versión posterior. Anule los criterios predeterminados de selección de la suite de cifrado para el cifrado basado en certificados.

Tenga en cuenta que este parámetro no puede usarse junto con `TLSAccept=cert,psk`; para una conexión con certificado (`TLSConnect=cert`), use en su lugar `TLSCipherAll13`.

TLSCipherPSK

Especifique la cadena de cifrado de OpenSSL (TLS 1.2). Anule los criterios de selección predeterminados de la suite de cifrado para el cifrado basado en PSK.

Ejemplo:

```
kECDHEPSK+AES128:kPSK+AES128
```

TLSCipherPSK13

Especifique la cadena de cifrado de OpenSSL (TLS 1.3) para OpenSSL 1.1.1 o una versión posterior. Anule los criterios de selección predeterminados de la suite de cifrado para el cifrado basado en PSK.

Ejemplo:

```
TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
```

TLSConnect

Especifique el nivel de cifrado requerido para las conexiones del agent con Zabbix server o proxy.
 Se usa para comprobaciones activas.
 Solo se puede especificar un valor: - *unencrypted* - conectarse sin cifrado (predeterminado); - *psk* - conectarse usando TLS y una clave precompartida (PSK); - *cert* - conectarse usando TLS y un certificado.

Obligatorio: sí, si se definen parámetros de certificado TLS o PSK (incluso para una conexión *unencrypted*); en caso contrario, no

TLSCRLFile

Especifique la ruta completa del archivo que contiene los certificados revocados. Este parámetro se utiliza para las comunicaciones cifradas entre los componentes de Zabbix.

TLSKeyFile

Especifique la ruta completa del archivo que contiene la clave privada del agent, utilizada para las comunicaciones cifradas entre los componentes de Zabbix.

TLSPSKFile

Especifique la ruta completa del archivo que contiene la clave precompartida del agent, utilizada para las comunicaciones cifradas con el server o proxy de Zabbix.

TLSPSKIdentity

Especifique la cadena de identidad de la clave precompartida, utilizada para comunicaciones cifradas con Zabbix server o proxy.

TLSServerCertIssuer

Especifique el emisor del certificado del server (proxy) permitido.

TLSServerCertSubject

Especifique el subject del certificado permitido del server (proxy).

UnsafeUserParameters

Permite que todos los caracteres se pasen como argumentos a parámetros definidos por el usuario. Los siguientes caracteres no están permitidos: \ ' " ` * ? [ ] { } ~ $! & ; ( ) < > | # @ %
Además, no se permiten caracteres de nueva línea.

Predeterminado: 0
 Valores: 0 - no permitir, 1 - permitir

UserParameter

Especifique un parámetro definido por el usuario para supervisar. Puede haber varios parámetros definidos por el usuario.
Formato: `UserParameter=<key>,<shell command>`
Tenga en cuenta que el comando de shell no debe devolver una cadena vacía ni solo EOL. Los comandos de shell pueden tener rutas relativas si se especifica el parámetro `UserParameterDir`.

Ejemplo:

```
UserParameter=system.test,who|wc -l
UserParameter=check_cpu,./custom_script.sh
```

UserParameterDir

Especifica la ruta de búsqueda predeterminada para los comandos `UserParameter`. Si se utiliza, el agent cambiará su directorio de trabajo al especificado aquí antes de ejecutar un comando. De este modo, los comandos UserParameter pueden usar un prefijo relativo `./` en lugar de una ruta completa.
Solo se permite una entrada.

Ejemplo:

```
UserParameterDir=/opt/myscripts
```

Parámetros de configuración: Zabbix agent 2 (Windows)

Descripción general

Los parámetros de configuración permiten personalizar el comportamiento de Zabbix agent 2 en Windows.

Especifique los valores de los parámetros en el archivo de configuración de Zabbix agent 2 (`zabbix_agent2.conf`). Todos los parámetros son **opcionales**, a menos que se indique explícitamente que el parámetro es obligatorio.

Esta página está estructurada de la siguiente manera:

- Descripción general de los parámetros (para consultar todos los detalles de un parámetro, haga clic en su nombre)
- Detalles de los parámetros

Parámetro	Descripción
Alias	Establece un alias para una clave de item.
AllowKey	Permite la ejecución de las claves de item que coincidan con un patrón.
AllowKeyRegexp	Permite la ejecución de las claves de item que coincidan con un patrón basado en una expresión regular.
BufferSend	Establece la frecuencia de envío de valores del búfer al Zabbix server.
BufferSize	Establece el número máximo de valores en el búfer de memoria.
ControlSocket	Especifica el socket de control utilizado para enviar comandos en tiempo de ejecución con la opción `-R`.
DebugLevel	Establece el nivel de depuración.
DenyKey	Deniega la ejecución de las claves de item que coincidan con un patrón.

Parámetro	Descripción
DenyKeyRegexp	Deniega la ejecución de las claves de item que coincidan con un patrón basado en una expresión regular.
EnablePersistentBuffer	Habilita el uso de almacenamiento persistente local para los items activos.
EnableProfiler	Habilita la creación periódica de perfiles de Go al iniciar el agent.
ForceActiveChecksOnStart	Realiza comprobaciones activas inmediatamente después del reinicio para la primera configuración recibida.
HeartbeatFrequency	Establece la frecuencia de los mensajes de latido en segundos.
HostInterface	Especifica la interfaz del host.
HostInterfaceItem	Especifica un item para obtener la interfaz del host.
HostMetadata	Especifica los metadatos del host.
HostMetadataItem	Especifica un item de Zabbix agent utilizado para obtener los metadatos del host.
Hostname	Especifica el nombre del host.
HostnameItem	Especifica un item de Zabbix agent para obtener el nombre del host.
Include	Especifica un directorio o archivos individuales que se incluirán en el archivo de configuración.
ListenIP	Especifica la dirección IP de escucha del agent para las conexiones desde el server/proxy.
ListenPort	Especifica el puerto de escucha del agent.
LogFile	Especifica el archivo de registro del agent.
LogFileSize	Establece el tamaño máximo del archivo de registro.
LogType	Especifica el tipo de salida del registro.
PerfCounter	Especifica un nuevo parámetro <parameter_name>, que corresponde al valor promedio del contador de rendimiento del sistema <perf_counter_path> durante el período especificado <period> (en segundos).
PerfCounterEn	Especifica un nuevo parámetro <parameter_name>, que corresponde al valor promedio del contador de rendimiento del sistema <perf_counter_path> durante el período especificado <period> (en segundos). A diferencia de PerfCounter, las rutas de los contadores de rendimiento deben estar en inglés.
PersistentBufferFile	Especifica el archivo donde Zabbix agent 2 debe mantener la base de datos SQLite.
PersistentBufferPeriod	Establece la duración del almacenamiento de datos cuando no hay conexión con el server o el proxy.
Plugins.<PluginName>.SystemCapacity	Establece el límite de comprobaciones por plugin que pueden ejecutarse simultáneamente.
Plugins.Log.MaxLinesPerSecond	Establece el número máximo de líneas nuevas que el agent enviará por segundo al Zabbix server o proxy al procesar comprobaciones activas `log`, `logrt` y `eventlog`.
Plugins.SystemRun.LogRemoteCommands	Habilita el registro de los comandos de shell ejecutados como advertencias.
PluginSocket	Especifica la ruta a la canalización con nombre de Windows para las comunicaciones con plugins cargables.
PluginTimeout	Establece el tiempo de espera para las conexiones con plugins cargables, en segundos.
ProfilerDir	El directorio donde se escribirán los datos de creación de perfiles.
ProfilerMaxFilesPerProfile	El número máximo de archivos que se conservarán para cada tipo de perfil.
ProfilerInterval	El intervalo de escritura de los datos de creación de perfiles.
RefreshActiveChecks	Especifica la frecuencia de actualización de la lista de comprobaciones activas.
Server	Especifica las direcciones IP o los nombres DNS de los Zabbix servers/proxies.
ServerActive	Especifica la dirección del Zabbix server/proxy o la configuración del clúster desde la que se obtendrán las comprobaciones activas.
SourceIP	Especifica la dirección IP de origen.
StatusPort	Establece el puerto en el que el agent escucha las solicitudes de estado HTTP (`http://localhost:<port>/status`).
Timeout	Especifica cuánto tiempo se debe esperar (en segundos) para establecer la conexión e intercambiar datos con el Zabbix server o proxy.
TLSAccept	Especifica el nivel de cifrado requerido para las conexiones entrantes.
TLSCAFile	Especifica la ruta completa del archivo que contiene los certificados de las CA de nivel superior para la verificación del certificado del par, utilizados para las comunicaciones cifradas entre componentes de Zabbix.
TLSCertFile	Especifica la ruta completa del archivo que contiene el certificado o la cadena de certificados del agent, utilizados para las comunicaciones cifradas entre componentes de Zabbix.
TLSCipherAll13	Especifica la cadena de cifrado de OpenSSL (TLS 1.3). Anula los criterios de selección predeterminados del conjunto de cifrados para el cifrado basado en certificados y PSK.
TLSCipherCert13	Especifica la cadena de cifrado de OpenSSL (TLS 1.3). Anula los criterios de selección predeterminados del conjunto de cifrados para el cifrado basado en certificados.
TLSCipherPSK13	Especifica la cadena de cifrado de OpenSSL (TLS 1.3). Anula los criterios de selección predeterminados del conjunto de cifrados para el cifrado basado en PSK.
TLSConnect	Especifica el nivel de cifrado requerido para las conexiones con el Zabbix server o proxy.

Parámetro	Descripción
TLSCRLFile	Especifica la ruta completa del archivo que contiene los certificados revocados. Este parámetro se utiliza para las comunicaciones cifradas entre componentes de Zabbix.
TLSKeyFile	Especifica la ruta completa del archivo que contiene la clave privada del agent, utilizada para las comunicaciones cifradas entre componentes de Zabbix.
TLSPSKFile	Especifica la ruta completa del archivo que contiene la clave precompartida del agent, utilizada para las comunicaciones cifradas con el Zabbix server o proxy.
TLSPSKIdentity	Especifica la cadena de identidad de la clave precompartida, utilizada para las comunicaciones cifradas con el Zabbix server o proxy.
TLSServerCertIssuer	Especifica el emisor de certificado permitido del server (proxy).
TLSServerCertSubject	Especifica el asunto de certificado permitido del server (proxy).
UnsafeUserParameters	Permite pasar todos los caracteres en los argumentos de los parámetros definidos por el usuario.
UserParameter	Especifica un parámetro definido por el usuario que se supervisará.
UserParameterDir	Especifica la ruta de búsqueda predeterminada para los comandos `UserParameter`.

Tenga en cuenta lo siguiente:

- Los valores predeterminados reflejan los valores predeterminados del proceso, no los valores de los archivos de configuración distribuidos;
- Los valores admiten variables de entorno;
- Zabbix admite archivos de configuración únicamente con codificación UTF-8 y sin BOM;
- Los comentarios que comienzan con # solo se admiten al principio de la línea.

Detalles del parámetro

Alias

Establezca un alias para una clave de item. Puede utilizarse para sustituir una clave de item larga y compleja por otra más corta y sencilla.
 Puede haber varios parámetros `Alias`. No se permiten varios parámetros con la misma clave `Alias`.
 Distintas claves `Alias` pueden hacer referencia a la misma clave de item.
 Los alias pueden utilizarse en `HostMetadataItem`, pero no en el parámetro `HostnameItem`.

Ejemplo 1: Recuperación del uso del archivo de paginación en porcentaje desde el server.

```
Alias=pg_usage:perf_counter[\Paging File(_Total)\% Usage]
```

Ahora puede utilizarse la clave abreviada `pg_usage` para recuperar datos.

Ejemplo 2: Obtención de la carga de la CPU con parámetros predeterminados y personalizados.

```
Alias=cpu.load:system.cpu.load
Alias=cpu.load[*]:system.cpu.load[*]
```

Esto permite utilizar la clave `cpu.load` para obtener la carga de la CPU con los parámetros predeterminados, así como utilizar `cpu.load[percpu,avg15]` para obtener datos específicos sobre la carga de la CPU.

Ejemplo 3: Ejecución de varias reglas de detección de bajo nivel que procesan los mismos items de detección.

```
Alias=vfs.fs.discovery[*]:vfs.fs.discovery
```

Ahora es posible configurar varias reglas de detección utilizando `vfs.fs.discovery` con distintos parámetros para cada regla, por ejemplo, `vfs.fs.discovery[foo]`, `vfs.fs.discovery[bar]`, etc.

AllowKey

Permite la ejecución de aquellas claves de item que coincidan con un patrón. El patrón de clave es una expresión comodín que admite el carácter * para coincidir con cualquier número de caracteres.
 Se pueden definir varias reglas de coincidencia de claves en combinación con `DenyKey`, `AllowKeyRegexp` y `DenyKeyRegexp`. Los parámetros se procesan uno por uno según el orden en que aparecen. El orden determina qué regla coincidente tendrá prioridad. Consulte también: Restricting agent checks.

AllowKeyRegexp

Permite la ejecución de los item keys que coincidan con un patrón.
El patrón de la key es una expresión regular.
 Se pueden definir varias reglas de coincidencia de keys en combinación con `DenyKeyRegexp`, `AllowKey` y `DenyKey`. Los parámetros se procesan uno por uno según el orden en que aparecen. El orden determina qué regla coincidente tendrá prioridad. Consulte también: Restricción de las comprobaciones del agent.

Ejemplo:

```
AllowKeyRegexp=^system\.run\[(?:/usr)?/bin/hostname(?: -[fsd])?\]$
```

BufferSend

Establece la frecuencia de envío de valores desde el buffer al Zabbix server/proxy (en segundos).
Tenga en cuenta que, si el buffer está lleno, los datos se enviarán antes.

Predeterminado: 5
 Rango: 1-3600

BufferSize

Establece el número máximo de valores en el búfer de memoria.
 El agent enviará todos los datos recopilados al server o proxy de Zabbix si el búfer está lleno.
Este parámetro solo debe usarse si el búfer persistente está deshabilitado (`EnablePersistentBuffer=0`).

Predeterminado: 1000
 Rango: 2-65535

ControlSocket

Especifique el socket de control, utilizado para enviar comandos en tiempo de ejecución con la opción `-R`.

Predeterminado: `\\.\pipe\agent.sock`

DebugLevel

Establezca el nivel de depuración: - *0* - información básica sobre el inicio y la detención de los procesos de Zabbix; - *1* - información crítica; - *2* - información de error; - *3* - advertencias; - *4* - para depuración (genera mucha información); - *5* - depuración ampliada (genera aún más información).

Predeterminado: 3
 Rango: 0-5

DenyKey

Deniega la ejecución de los item keys que coincidan con un patrón. El patrón de la key es una expresión comodín que admite el carácter `*` para coincidir con cualquier cantidad de caracteres.
 Se pueden definir varias reglas de coincidencia de keys en combinación con `AllowKey`, `AllowKeyRegexp` y `DenyKeyRegexp`. Los parámetros se procesan uno por uno según el orden en que aparecen. El orden determina qué regla coincidente tendrá prioridad.

Véase también: Restricción de las comprobaciones del agent.

Valor predeterminado: `DenyKey=system.run[*]`

DenyKeyRegexp

Deniega la ejecución de los item keys que coincidan con un patrón.
El patrón de la key es una expresión regular.
 Se pueden definir varias reglas de coincidencia de keys en combinación con `AllowKeyRegexp`, `AllowKey` y `DenyKey`. Los parámetros se procesan uno por uno según el orden en que aparecen. El orden determina qué regla de coincidencia tendrá prioridad.
 Todos los items `system.run[]` se deniegan de forma predeterminada (`DenyKey=system.run[*]`), a menos que se especifique otra regla `system.run[*]`.

Consulte también: Restricción de las comprobaciones del agent.

EnablePersistentBuffer

Habilita el uso de almacenamiento persistente local para los items activos. Si el almacenamiento persistente está deshabilitado, se usará el búfer de memoria.

Predeterminado: 0
 Valores: 0 - deshabilitado, 1 - habilitado

EnableProfiler

Habilita el perfilado periódico de Go al iniciar el agent. El perfilado también se puede habilitar o deshabilitar en tiempo de ejecución mediante la opción `-R`. Mientras `periodic_prof_execute` recopila perfiles, todos los comandos de tiempo de ejecución del perfilador se rechazan porque está ocupado.

Valor predeterminado: 0
 Valores: 0 - deshabilitado, 1 - habilitado

ForceActiveChecksOnStart

Realiza comprobaciones activas inmediatamente después del reinicio para la primera configuración recibida. También está disponible como un parámetro de configuración por plugin, por ejemplo: `Plugins.Uptime.System.ForceActiveChecksOnStart=1`

Predeterminado: 0
 Valores: 0 - deshabilitado, 1 - habilitado

HeartbeatFrequency

Establece la frecuencia de los mensajes de heartbeat en segundos. Se utiliza para supervisar la disponibilidad de las comprobaciones activas.
0 - mensajes de heartbeat deshabilitados.

Predeterminado: 60
 Rango: 0-3600

HostInterface

Especifique la interfaz del host (dirección IP o nombre DNS) utilizada durante el autoregistro del host. Este valor se utilizará para rellenar la interfaz en el host recién creado y permite configurar explícitamente una dirección IP o DNS. Para más detalles, consulte Uso de DNS como interfaz predeterminada.

Si no se define, el valor se obtendrá de HostInterfaceItem.

El agent mostrará un error y no se iniciará si el valor supera el límite de 255 caracteres.

Rango: 0-255 caracteres

HostInterfaceItem

Especifique un item para determinar la interfaz del host (dirección IP o nombre DNS) durante el registro automático del host. Este valor se utiliza únicamente si `HostInterface` no está definido. Para obtener más información, consulte Uso de DNS como interfaz predeterminada.

Durante una solicitud de registro automático, el agent registrará un mensaje de advertencia si el valor devuelto por el item especificado supera el límite de 255 caracteres.

El item `system.run[]` es compatible independientemente de la configuración de `AllowKey`, `AllowKeyRegexp`, `DenyKey` o `DenyKeyRegexp`.

HostMetadata

Especifique los metadatos utilizados para identificar o distinguir el host durante la autoregistración del host (agent activo). `HostMetadata` permite distinguir entre hosts más allá del nombre del host.

Si no se define, el valor se obtendrá de `HostMetadataItem`.

El agent generará un error y no se iniciará si el valor especificado supera el límite de 2034 bytes o si es una cadena que no está codificada en UTF-8.
Cuando un parámetro espera una dirección IP o un nombre DNS, también se rechazarán y se notificarán como no válidos los valores que sean UTF-8 válido, pero no sean direcciones IP ni nombres DNS válidos.

No se admiten metadatos de varias líneas: la salida se truncará en el primer salto de línea.

Rango: 0-2034 bytes

HostMetadataItem

Especifique un item del Zabbix agent para obtener los metadatos del host. Esta opción solo se utiliza cuando `HostMetadata` no está definido.

El valor de `HostMetadataItem` se obtiene en cada intento de registro automático y se utiliza únicamente durante el proceso de registro automático del host (agent activo). `HostMetadataItem` permite distinguir entre hosts más allá del nombre del host.

Se admiten los parámetros de usuario y los alias. El item `system.run[]` se admite independientemente de la configuración de `AllowKey`, `AllowKeyRegexp`, `DenyKey` o `DenyKeyRegexp`.

Durante una solicitud de registro automático, el agent registrará un mensaje de advertencia si el valor devuelto por el item especificado supera el límite de 65535 puntos de código UTF-8. El valor devuelto por el item debe ser una cadena UTF-8; de lo contrario, se ignorará. Cuando un parámetro espera una dirección IP o un nombre DNS, también se rechazarán y se notificarán como no válidos los valores que sean UTF-8 válido, pero no sean direcciones IP ni nombres DNS válidos.

No se admiten metadatos multilínea: la salida se truncará en el primer salto de línea.

Nombre de host

Especifique nombres de host únicos y sensibles a mayúsculas y minúsculas. Una lista delimitada por comas.
 Obligatorio para las comprobaciones activas y debe coincidir con los nombres de host configurados en el server. El valor se obtiene de HostnameItem si no está definido.
 Caracteres permitidos: alfanuméricos, '.', ' ', '_' y '-'. Longitud máxima: 128 caracteres por nombre de host, 2048 caracteres para toda la línea.

Predeterminado: establecido por HostnameItem

HostnameItem

Especifique un item para obtener el nombre del host.
Esta opción solo se utiliza cuando `Hostname` no está definido.

No se admiten parámetros de usuario ni alias, pero el item `system.run[]` es compatible independientemente de los valores de `AllowKey`, `AllowKeyRegexp`, `DenyKey` o `DenyKeyRegexp`.

Valor predeterminado: `system.hostname`

Include

Especifique un directorio o archivos individuales que se incluirán en el archivo de configuración (ubicado en `C:\Program Files\Zabbix Agent 2` de forma predeterminada si Zabbix agent se instala mediante los paquetes del instalador MSI de Windows; ubicado en la carpeta especificada durante la instalación si Zabbix agent se instala como un archivo zip). Durante la instalación de Zabbix, este parámetro se establece en `/usr/local/etc`, a menos que lo cambie durante la compilación.

Reglas generales para el parámetro `Include`:

- La ruta puede ser relativa a la ubicación del archivo `zabbix_agent2.conf` (por ejemplo, `Include=.\zabbix_agent2.d\plugins.c`
- Puede especificar varios parámetros `Include` en un mismo archivo de configuración.
- Zabbix procesa el parámetro `Include` en cuanto lo lee, por lo que se recomienda colocarlo al final del archivo de configuración.

Reglas para los archivos incluidos:

- Cada archivo debe poder leerse.
- Cada archivo debe tener una sintaxis correcta. De lo contrario, el componente no se iniciará.

Reglas para incluir directorios o patrones de archivos:

- No dé por supuesto ningún orden de inclusión (por ejemplo, los archivos no se incluyen en orden alfabético).
- No defina el mismo parámetro en más de un archivo incluido (por ejemplo, para sobrescribir una configuración general con una específica), ya que no puede predecir qué valor se utilizará.
- Preste atención a las copias de seguridad que algunos editores de texto crean automáticamente (por ejemplo, al editar `include/my_specific.conf` se puede crear `include/my_specific.conf.BAK`), ya que ambos archivos se incluirán.

Para incluir únicamente los archivos relevantes de un directorio, utilice el comodín (`*`) para la coincidencia de patrones.

Ejemplo:

```
Include=C:\Program Files\Zabbix Agent2\zabbix_agent2.d\*.conf
```

ListenIP

Especifique la dirección IP de escucha del agent para conexiones desde el server/proxy. Una lista delimitada por comas.
 La primera dirección IP se envía al server de Zabbix, si se conecta a él, para recuperar la lista de comprobaciones activas.

Predeterminado: `0.0.0.0`

ListenPort

Especifique el puerto de escucha del agent para conexiones desde el server/proxy.

Predeterminado: 10050
 Rango: 1024-32767

LogFile

Especifique el archivo de registro del agent.

Predeterminado: `c:\zabbix_agent2.log`
 Obligatorio: Sí, si LogType está establecido en `file`; de lo contrario, no

LogFileSize

Establece el tamaño máximo del archivo de registro (en MB).
0 - deshabilita la rotación automática de registros.
Nota: Si se alcanza el límite de tamaño del archivo de registro y la rotación del archivo falla, por cualquier motivo, el archivo de registro existente se trunca y se inicia uno nuevo.

Predeterminado: 1
 Rango: 0-1024

LogType

Especifique el tipo de salida del registro: - *file* - escribe el registro en el archivo especificado por el parámetro `LogFile`; - *console* - escribe el registro en la salida estándar.

Predeterminado: `file`

PerfCounter

Defina un nuevo parámetro <parameter_name> que sea el valor promedio del contador de rendimiento del sistema <perf_counter_path> para el período de tiempo especificado <period> (en segundos).
Sintaxis: <parameter_name>,"<perf_counter_pat

Por ejemplo, si desea recibir el número promedio de interrupciones del procesador por segundo durante el último minuto, puede definir un nuevo parámetro ”interrupts” de la siguiente manera:

```
PerfCounter = interrupts,"\Processor(0)\Interrupts/sec",60
```

Tenga en cuenta las comillas dobles alrededor de la ruta del contador de rendimiento. El nombre del parámetro (interrupts) se utilizará como la clave del item al crear un item. Las muestras para calcular el valor promedio se tomarán cada segundo.
Puede ejecutar `typeperf -qx` para obtener la lista de todos los contadores de rendimiento disponibles en Windows.

PerfCounterEn

Defina un nuevo parámetro <parameter_name> que sea el valor promedio del contador de rendimiento del sistema <perf_counter_path> para el período de tiempo especificado <period> (en segundos). En comparación con PerfCounter, las rutas de perfcounter deben estar en inglés. Solo es compatible con **Windows Server 2008/Vista** y versiones posteriores.
Sintaxis: <parameter_name>,”<perf_counter_path>”,<period>

Por ejemplo, si desea recibir el número promedio de interrupciones del procesador por segundo durante el último minuto, puede definir un nuevo parámetro ”interrupts” de la siguiente manera:

```
PerfCounterEn = interrupts,"\Processor(0)\Interrupts/sec",60
```

Tenga en cuenta las comillas dobles alrededor de la ruta del contador de rendimiento. El nombre del parámetro (interrupts) se utilizará como la clave del item al crear un item. Las muestras para calcular el valor promedio se tomarán cada segundo.
Puede encontrar la lista de cadenas en inglés consultando la siguiente clave del registro: `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Perflib\009`.

PersistentBufferFile

Especifique el archivo donde Zabbix agent 2 debe guardar la base de datos SQLite. Debe ser un nombre de archivo completo.
 Este parámetro solo se usa si el búfer persistente está habilitado (`EnablePersistentBuffer=1`).

PersistentBufferPeriod

Establece la duración del almacenamiento de datos cuando no hay conexión con el server o proxy. Los datos más antiguos se perderán. Los datos de registro se conservarán.
 Este parámetro solo se usa si el búfer persistente está habilitado (`EnablePersistentBuffer=1`).

Predeterminado: `1h`
 Rango: 1m-365d

Plugins.<PluginName>.System.Capacity

Establece el límite de comprobaciones por plugin <`PluginName`> que pueden ejecutarse al mismo tiempo.

Predeterminado: 1000 Rango: 1-1000

Plugins.Log.MaxLinesPerSecond

Establece el número máximo de nuevas líneas que el agent enviará por segundo a Zabbix server o proxy al procesar comprobaciones activas `log`, `logrt` y `eventlog`. El valor proporcionado se sobrescribirá mediante el parámetro `maxlines`, especificado en la clave del item `log`, `logrt` o `eventlog`.
Nota: Zabbix procesará 10 veces más nuevas líneas que las establecidas en `MaxLinesPerSecond` para buscar la cadena requerida en los items de log.

Predeterminado: 20
 Rango: 1-1000

Plugins.SystemRun.LogRemoteCommands

Habilita el registro de los comandos de shell ejecutados como advertencias. Los comandos se registrarán solo si se ejecutan de forma remota. No se crearán entradas de registro si `system.run[]` se inicia localmente mediante los parámetros `HostMetadataItem`, `HostInterfaceItem` o `HostnameItem`.

Predeterminado: 0
 Valores: 0 - deshabilitado, 1 - habilitado

PluginSocket

Especifique la ruta de la canalización con nombre de Windows para las comunicaciones del plugin cargable.

Predeterminado: `\\.\pipe\agent.plugin.sock`

PluginTimeout

Establece el tiempo de espera para las conexiones con plugins cargables, en segundos.

Predeterminado: `Timeout`
 Rango: 1-30

ProfilerDir

El directorio donde se escribirán los datos de perfilado.

Valor predeterminado: `c:\zabbix_pprof<br>`

ProfilerMaxFilesPerProfile

El número máximo de archivos que se conservarán para cada tipo de perfil. El límite se aplica a los archivos completados después de cada volcado de perfil. El perfil de CPU activo se recopila en un archivo temporal oculto y no cuenta para el límite. Los archivos creados por `periodic_prof_execute` también están sujetos al límite.

Valor predeterminado: 10
 Rango: 1-100

ProfilerInterval

El intervalo de escritura de los datos de perfilado, en segundos.

Valor predeterminado: 3600
 Rango: 1-86400

RefreshActiveChecks

Especifique la frecuencia de actualización de la lista de comprobaciones activas, en segundos.
 Tenga en cuenta que, si falla la actualización de las comprobaciones activas, el siguiente intento se realizará en 60 segundos.

Predeterminado: 5
 Rango: 1-86400

Server

Especifique las direcciones IP, opcionalmente en notación CIDR, o los nombres DNS de los servers/proxies de Zabbix. Una lista delimitada por comas.
 Si no se especifica el parámetro, las comprobaciones pasivas se deshabilitan y el agent no escuchará en ningún puerto TCP.
 Las conexiones entrantes solo se aceptarán desde los hosts enumerados aquí. Si la compatibilidad con IPv6 está habilitada, entonces `127.0.0.1`, `::127.0.0.1`, `::ffff:127.0.0.1` se tratan por igual y `::/0` permitirá cualquier dirección IPv4 o IPv6. `0.0.0.0/0` se puede usar para permitir cualquier dirección IPv4. Se permiten espacios.

Ejemplo:

```
Server=127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.example.com
```

ServerActive

Especifique la dirección del server/proxy de Zabbix o la configuración del clúster para obtener los checks activos.
 Si no se especifica el parámetro, los checks activos se deshabilitan.

La dirección del server/proxy es una dirección IP o un nombre DNS y un puerto opcional, separados por dos puntos. Si no se especifica el puerto, se usa el puerto predeterminado.
 Las direcciones IPv6 deben ir entre corchetes si se especifica el puerto para ese host. Si no se especifica el puerto, los corchetes para las direcciones IPv6 son opcionales.

La configuración del clúster es una o más direcciones de miembros del grupo de server o proxy separadas por punto y coma.

Se pueden especificar varios Zabbix servers/clusters y Zabbix proxies, separados por comas.

A menos que se usen grupos de proxy, no se debe especificar más de un Zabbix proxy para cada Zabbix server/cluster. Si se especifica un Zabbix proxy, entonces no se debe especificar el Zabbix server/cluster para ese proxy.

Se pueden proporcionar varias direcciones delimitadas por comas para usar varios Zabbix servers independientes en paralelo. Se permiten espacios.

Ejemplo para Zabbix proxy:

```
ServerActive=127.0.0.1:10051
```

Ejemplo para grupo de Zabbix proxy:

```
ServerActive=proxy1.example.com;proxy2.example.com;proxy3.example.com;proxy4.example.com;proxy5.example.co
```

Ejemplo para varios servers:

```
ServerActive=127.0.0.1:20051,zabbix.domain,\[::1\]:30051,::1,\[12fc::1\]
```

Ejemplo para alta disponibilidad:

```
ServerActive=zabbix.cluster.node1;zabbix.cluster.node2:20051;zabbix.cluster.node3
```

Ejemplo para alta disponibilidad con dos clústeres y un server:

```
ServerActive=zabbix.cluster.node1;zabbix.cluster.node2:20051,zabbix.cluster2.node1;zabbix.cluster2.node2,z
```

SourceIP

Especifique la dirección IP de origen para:

- conexiones salientes a Zabbix server o Zabbix proxy;

- realizar conexiones al ejecutar algunos items (`web.page.get`, `net.tcp.port`, etc.).

StatusPort

Establezca el puerto en el que el agent escucha las solicitudes de estado HTTP (`http://localhost:<port>/status`).

Rango: 1024-32767

Tiempo de espera

Especifique cuánto tiempo esperar (en segundos) para establecer la conexión e intercambiar datos con Zabbix proxy o server.

Este parámetro define la duración de varias operaciones de comunicación, incluidas:

- esperar una respuesta del Zabbix server;
- enviar solicitudes a Zabbix server, incluidas las solicitudes de configuración de item y los datos de item en active checks;
- recuperar datos de registro mediante la supervisión de logfile o del registro de eventos de Windows;
- enviar mensajes de latido;
- duración máxima para comprobaciones `vfs.*`;
- servir como valor de reserva en escenarios en los que un server o proxy anterior a la versión 7.0 envía comprobaciones sin tiempos de espera.

Este tiempo de espera **no** se utilizará para aquellas comprobaciones de agent que tienen configuraciones de tiempo de espera ajustables en el frontend (a nivel global, de proxy o por item).

Predeterminado: 3
 Rango: 1-30

TLSAccept

Especifique el nivel de cifrado requerido para las conexiones entrantes.
 Se utiliza para comprobaciones pasivas.

Se pueden especificar varios valores, separados por comas:

- *unencrypted* - aceptar conexiones sin cifrado (predeterminado);
- *psk* - aceptar conexiones con TLS y una clave precompartida (PSK);
- *cert* - aceptar conexiones con TLS y un certificado.

Obligatorio: sí, si se definen parámetros de certificado TLS o PSK (incluso para una conexión *unencrypted*); en caso contrario, no

TLSCAFile

Especifique la ruta completa del archivo que contiene los certificados de la CA de nivel superior para la verificación del certificado del par, utilizados para las comunicaciones cifradas entre los componentes de Zabbix.

TLSCertFile

Especifique la ruta completa del archivo que contiene el certificado del agent o la cadena de certificados, utilizado para comunicaciones cifradas con los componentes de Zabbix.

TLSCipherAll13

Especifique la cadena de cifrado de OpenSSL (TLS 1.3). Anule los criterios de selección predeterminados de la suite de cifrado para el cifrado basado en certificados y PSK.

Ejemplo:

```
TLS_AES_256_GCM_SHA384:TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
```

TLSCipherCert13

Especifique la cadena de cifrado de OpenSSL (TLS 1.3). Anule los criterios de selección predeterminados del conjunto de cifrado para el cifrado basado en certificados.

Tenga en cuenta que este parámetro no puede usarse junto con `TLSAccept=cert,psk`; para una conexión con certificado (`TLSConnect=cert`), use `TLSCipherAll13` en su lugar.

TLSCipherPSK13

Especifique la cadena de cifrado de OpenSSL (TLS 1.3). Anule los criterios de selección predeterminados del conjunto de cifrado para el cifrado basado en PSK.

Ejemplo:

```
TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
```

TLSConnect

Especifique el nivel de cifrado requerido para las conexiones del agent con Zabbix server o proxy.
 Se utiliza para comprobaciones activas.
 Solo se puede especificar un valor: - *unencrypted* - conectar sin cifrado (predeterminado); - *psk* - conectar usando TLS y una clave precompartida (PSK); - *cert* - conectar usando TLS y un certificado.

Obligatorio: sí, si se definen parámetros de certificado TLS o PSK (incluso para una conexión *unencrypted*); de lo contrario, no

TLSCRLFile

Especifique la ruta completa del archivo que contiene los certificados revocados. Este parámetro se utiliza para las comunicaciones cifradas entre los componentes de Zabbix.

TLSKeyFile

Especifique la ruta completa del archivo que contiene la clave privada del agent, utilizada para las comunicaciones cifradas entre los componentes de Zabbix.

TLSPSKFile

Especifique la ruta completa del archivo que contiene la clave precompartida del agent, utilizada para las comunicaciones cifradas con Zabbix server o proxy.

TLSPSKIdentity

Especifique la cadena de identidad de clave precompartida, utilizada para comunicaciones cifradas con Zabbix server o proxy.

TLSServerCertIssuer

Especifique el emisor del certificado del server (proxy) permitido.

TLSServerCertSubject

Especifica el subject del certificado del server permitido (proxy).

UnsafeUserParameters

Permite que todos los caracteres se pasen como argumentos a parámetros definidos por el usuario. Los siguientes caracteres no están permitidos: \ ' " ` * ? [ ] { } ~ $! & ; ( ) < > | # @ %
Además, no se permiten caracteres de nueva línea.

Predeterminado: 0
 Valores: 0 - no permitir, 1 - permitir

UserParameter

Especifique un parámetro definido por el usuario para supervisar. Puede haber varios parámetros definidos por el usuario.
Formato: `UserParameter=<key>,<shell command>`
Tenga en cuenta que el comando de shell no debe devolver una cadena vacía ni solo EOL. Los comandos de shell pueden tener rutas relativas si se especifica el parámetro `UserParameterDir`.

Ejemplo:

```
UserParameter=system.test,dir /b | find /c /v ""
UserParameter=check_cpu,.\custom\_script.bat
```

UserParameterDir

Especifique la ruta de búsqueda predeterminada para los comandos `UserParameter`. Si se utiliza, el agent cambiará su directorio de trabajo al especificado aquí antes de ejecutar un comando. De este modo, los comandos UserParameter pueden usar un prefijo relativo `.\` en lugar de una ruta completa.
Solo se permite una entrada.

Ejemplo:

```
UserParameterDir=C:\opt\myscripts
```

Parámetros de configuración: Plugins

Consulte los parámetros de configuración de los plugins de Zabbix agent 2:

- plugin de Ceph
- plugin de Docker
- plugin de Ember+
- plugin de Memcached
- plugin de Modbus
- plugin de MongoDB
- plugin de MQTT
- plugin de MSSQL
- plugin de MySQL

- plugin de GPU NVIDIA
- plugin de Oracle
- plugin de PostgreSQL
- plugin de Redis
- plugin S.M.A.R.T.

Principios de configuración de los plugins

Esta página describe los principios de configuración de los plugins y las prácticas recomendadas.

Todos los plugins se configuran mediante el parámetro `Plugins.*`, que puede formar parte del archivo de configuración del agent 2 de Zabbix o del archivo de configuración de un plugin. Si un plugin utiliza un archivo de configuración independiente, la ruta a este archivo debe especificarse en el parámetro `Include` del archivo de configuración del agent 2 de Zabbix.

Un parámetro típico de un plugin tiene la siguiente estructura:

`Plugins.<PluginName>.<Parameter>=<Value>`

Además, existen dos grupos específicos de parámetros:

- `Plugins.<PluginName>.Default.<Parameter>=<Value>` se utiliza para definir valores de parámetros predeterminados.
- `Plugins.<PluginName>.<SessionName>.<Parameter>=<Value>` se utiliza para definir conjuntos independientes de parámetros para distintos objetivos de monitorización mediante sesiones con nombre.

Todos los nombres de parámetros deben cumplir los siguientes requisitos:

- Se recomienda escribir con mayúscula inicial los nombres de los plugins.
- El parámetro debe escribirse con mayúscula inicial.
- No se permiten caracteres especiales.
- El anidamiento no está limitado por un nivel máximo.
- El número de parámetros no está limitado.

Por ejemplo, para realizar comprobaciones activas que no tengan un *intervalo de actualización* de Scheduling inmediatamente después del reinicio del agent y solo para el plugin Uptime, establezca `Plugins.Uptime.System.ForceActiveChecksOnStart=1` en el archivo de configuración. Del mismo modo, para establecer un límite personalizado para las comprobaciones simultáneas del plugin CPU, establezca `Plugins.CPU.System.Capacity=N` en el archivo de configuración.

Valores predeterminados

Puede establecer valores predeterminados para los parámetros relacionados con la conexión (URI, nombre de usuario, contraseña, etc.) en el archivo de configuración con el formato:

```
Plugins.<PluginName>.Default.<Parameter>=<Value>
```

Por ejemplo, `Plugins.Mysql.Default.Username=zabbix`, `Plugins.MongoDB.Default.Uri=tcp://127.0.0.1:27017`, etc.

Si no se proporciona un valor para dicho parámetro en una clave de item o en los parámetros de la sesión con nombre, el plugin utilizará el valor predeterminado. Si un parámetro predeterminado tampoco está definido, se utilizarán los valores predeterminados codificados.

> **Note:**
> Si una clave de item no tiene ningún parámetro, Zabbix agent 2 intentará recopilar la métrica utilizando los valores definidos en la sección de parámetros predeterminados.

Sesiones con nombre

Las sesiones con nombre representan un nivel adicional de parámetros del plugin y se pueden usar para especificar conjuntos separados de parámetros de autenticación para cada una de las instancias supervisadas. Cada parámetro de sesión con nombre debe tener la siguiente estructura:

```
Plugins.<PluginName>.Sessions.<SessionName>.<Parameter>=<Value>
```

Un nombre de sesión se puede usar como parámetro de clave de elemento connString en lugar de especificar una URI, nombre de usuario y/o contraseña por separado.

En las claves de elemento, el primer parámetro puede ser connString o una URI. Si el primer parámetro de la clave no coincide con ningún nombre de sesión, se tratará como una URI. Tenga en cuenta que no se admite pasar credenciales de URI incrustadas en la clave de elemento; en su lugar, use parámetros de sesión con nombre.

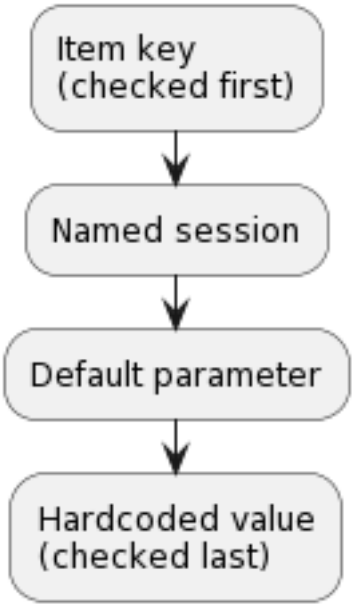
La lista de parámetros de sesión con nombre disponibles depende del plugin.

Es posible sobrescribir los parámetros de sesión especificando nuevos valores en los parámetros de la clave de elemento (consulte el ejemplo).

Si un parámetro no está definido para la sesión con nombre, Zabbix agent 2 usará el valor definido en el parámetro predeterminado del plugin.

Prioridad de parámetros

Los plugins de Zabbix agent 2 buscan los valores de los parámetros relacionados con la conexión en el siguiente orden:



1. El primer parámetro de la clave del item se compara con los nombres de sesión. Si no se encuentra ninguna coincidencia, se trata como un valor real; en este caso, se omitirá el paso 3. Si se encuentra una coincidencia, el valor del parámetro (normalmente, una URI) debe definirse en la sesión con nombre.
2. Los demás parámetros se tomarán de la clave del item si están definidos.
3. Si un parámetro de la clave del item (por ejemplo, password) está vacío, el plugin buscará el parámetro correspondiente de la sesión con nombre.
4. Si el parámetro de la sesión tampoco está especificado, se usará el valor definido en el parámetro predeterminado correspondiente.
5. Si todo lo demás falla, el plugin usará el valor predeterminado codificado.

Ejemplo 1

Monitorización de dos instancias "MySQL1" y "MySQL2".

Parámetros de configuración:

```
Plugins.Mysql.Sessions.MySQL1.Uri=tcp://127.0.0.1:3306
Plugins.Mysql.Sessions.MySQL1.User=mysql1_user
Plugins.Mysql.Sessions.MySQL1.Password=unique_password
Plugins.Mysql.Sessions.MySQL2.Uri=tcp://192.0.2.1:3306
Plugins.Mysql.Sessions.MySQL2.User=mysql2_user
Plugins.Mysql.Sessions.MySQL2.Password=different_password
```

Como resultado de esta configuración, cada nombre de sesión puede usarse como connString en una item key, por ejemplo, `mysql.ping[MySQL1]` o `mysql.ping[MySQL2]`.

Ejemplo 2

Proporcionar algunos de los parámetros en la clave del item.

Parámetros de configuración:

```
Plugins.PostgreSQL.Sessions.Session1.Uri=tcp://192.0.2.234:5432
Plugins.PostgreSQL.Sessions.Session1.User=old_username
Plugins.PostgreSQL.Sessions.Session1.Password=session_password
```

Clave del item: `pgsql.ping[session1,new_username,,postgres]`

Como resultado de esta configuración, el agent se conectará a PostgreSQL usando los siguientes parámetros:

- URI del parámetro de sesión: `192.0.2.234:5432`
- Nombre de usuario de la clave del item: `new_username`
- Contraseña del parámetro de sesión (ya que se omite en la clave del item): `session_password`
- Nombre de la base de datos de la clave del item: `postgres`

Ejemplo 3

Recopilación de una métrica usando parámetros de configuración predeterminados.

Parámetros de configuración:

```
Plugins.PostgreSQL.Default.Uri=tcp://192.0.2.234:5432
Plugins.PostgreSQL.Default.User=zabbix
Plugins.PostgreSQL.Default.Password=password
```

Clave del item: `pgsql.ping[,,,postgres]`

Como resultado de esta configuración, el agent se conectará a PostgreSQL usando los parámetros:

- URI predeterminada: `192.0.2.234:5432`
- Nombre de usuario predeterminado: `zabbix`
- Contraseña predeterminada: `password`
- Nombre de la base de datos de la clave del item: `postgres`

Conexiones

Algunos plugins admiten la recopilación de métricas de varias instancias simultáneamente. Se pueden supervisar tanto instancias locales como remotas. Se admiten conexiones TCP y Unix-socket.

Se recomienda configurar los plugins para mantener las conexiones con las instancias en estado abierto. Las ventajas son una menor congestión de red, menor latencia y un menor uso de CPU y memoria debido a la menor cantidad de conexiones. La biblioteca cliente se encarga de esto.

> **Note:**
> El período de tiempo durante el cual las conexiones no utilizadas deben permanecer abiertas puede determinarse mediante el parámetro `Plugins.<PluginName>.KeepAlive`. Ejemplo: `Plugins.Memcached.KeepAlive`

Complemento Ceph

Resumen

Los parámetros de configuración permiten personalizar el comportamiento del plugin Ceph.

Se recomienda especificarlos en su propio archivo de configuración (por ejemplo, `ceph.conf`), en lugar de en el archivo de configuración de agent 2. Luego, use la directiva Include para incluir este archivo en la configuración de Zabbix agent 2.

Todos los parámetros son **opcionales**, salvo que se indique explícitamente que el parámetro es obligatorio.

Esta página está estructurada de la siguiente manera:

- Resumen de parámetros (para ver los detalles completos de un parámetro, haga clic en su nombre)
- Detalles de los parámetros

Parameter	Description
Plugins.Ceph.Default.ApiKey	Especifica la clave API predeterminada para conectarse a Ceph.
Plugins.Ceph.Default.Mode	Especifica el modo predeterminado para conectarse a Ceph.
Plugins.Ceph.Default.URI	Especifica la URI predeterminada para conectarse a Ceph.
Plugins.Ceph.Default.User	Especifica el nombre de usuario predeterminado para conectarse a Ceph.
Plugins.Ceph.InsecureSkipVerify	Especifica si un cliente HTTP debe verificar la cadena de certificados del server y el nombre de host.
Plugins.Ceph.KeepAlive	Establece el tiempo máximo de espera antes de cerrar las conexiones del plugin no utilizadas.
Plugins.Ceph.Sessions.<SessionName>.ApiKey	Especifica la clave API de la sesión con nombre para conectarse a Ceph.
Plugins.Ceph.Sessions.<SessionName>.Mode	Especifica el modo de la sesión con nombre para conectarse a Ceph.
Plugins.Ceph.Sessions.<SessionName>.URI	Especifica la URI de la sesión con nombre para conectarse a Ceph.
Plugins.Ceph.Sessions.<SessionName>.User	Especifica el nombre de usuario de la sesión con nombre para conectarse a Ceph.
Plugins.Ceph.Timeout	Establece el tiempo de espera de ejecución de la solicitud.

Tenga en cuenta que:

- Los valores predeterminados reflejan los valores predeterminados del proceso, no los valores de los archivos de configuración suministrados;
- Los valores admiten variables de entorno;
- Zabbix solo admite archivos de configuración en codificación UTF-8 sin BOM;
- Los comentarios que comienzan con ”#” solo se admiten al principio de la línea.

Detalles del parámetro

Plugins.Ceph.Default.ApiKey

Especifique la clave API predeterminada para conectarse a Ceph.
 Se usa si no se especifica ningún valor en una clave de item o en una sesión con nombre.

Plugins.Ceph.Default.Mode

Especifique el modo predeterminado para conectarse a Ceph.
 El modo nativo solo es compatible en Linux y, a partir de Ceph 16, está disponible. El modo restful no funcionará con la versión 20 de Ceph o posterior.
 Tenga en cuenta que el conjunto de credenciales de usuario difiere para cada modo y que no son compatibles entre sí.

Predeterminado: `restful` (obsoleto)
 Rango: native/restful

Plugins.Ceph.Default.URI

Especifique la URI predeterminada para conectarse a Ceph.
 Se usa si no se especifica ningún valor en una clave de item o en una sesión con nombre.

No debe incluir credenciales incrustadas (se ignorarán).
Debe coincidir con el formato de URI.
Solo se admite el esquema `https`; el esquema puede omitirse.
El puerto puede omitirse (predeterminado=8003).

Predeterminado: `https://localhost:8003`

Ejemplos:

```
https://127.0.0.1:8003
localhost
```

Plugins.Ceph.Default.User

Especifique el nombre de usuario predeterminado para conectarse a Ceph.
 Se utiliza si no se especifica ningún valor en una clave de item o en una sesión con nombre.

Plugins.Ceph.InsecureSkipVerify

Especifique si un cliente HTTP debe verificar la cadena de certificados y el nombre de host del server.
 Si es *true*, TLS acepta cualquier certificado presentado por el server y cualquier nombre de host incluido en ese certificado. En este modo, TLS es susceptible a ataques de intermediario (úselo solo para pruebas).

Predeterminado: `false`
 Rango: false/true

Plugins.Ceph.KeepAlive

Establece el tiempo máximo de espera (en segundos) antes de cerrar las conexiones del plugin no utilizadas.

Predeterminado: 300
 Rango: 60-900

Plugins.Ceph.Sessions.<SessionName>.ApiKey

Especifique la clave API de la sesión con nombre para conectarse a Ceph.
 <SessionName> - defina el nombre de una sesión para usarlo en las claves de item.

Plugins.Ceph.Sessions.<SessionName>.Mode

Especifique el modo de sesión con nombre para conectarse a Ceph.
 <SessionName> - defina el nombre de una sesión para usarlo en las claves de item.
 El modo nativo solo es compatible con Linux y a partir de Ceph 16. El modo restful no funcionará con la versión 20 de Ceph ni con versiones posteriores.
 Tenga en cuenta que el conjunto de credenciales de usuario difiere para cada modo y que no son compatibles entre sí.

Predeterminado: `restful` (obsoleto)
 Rango: native/restful

Plugins.Ceph.Sessions.<SessionName>.URI

Especifique el URI de la sesión con nombre para conectarse a Ceph.
 <SessionName> - defina el nombre de una sesión para usarlo en las claves de item.
No debe incluir credenciales incrustadas (se ignorarán).
Debe coincidir con el formato URI.
 Solo se admite el esquema `https`; el esquema puede omitirse.
El puerto puede omitirse (predeterminado=8003).

Predeterminado: `https://localhost:8003`

Ejemplos:

```
https://127.0.0.1:8003
localhost
```

Plugins.Ceph.Sessions.<SessionName>.User

Especifique el nombre de usuario de la sesión nombrada para conectarse a Ceph.
 <SessionName> - defina el nombre de una sesión para usar en las claves de item.

Plugins.Ceph.Timeout

Establezca el tiempo de espera de ejecución de la solicitud (duración que se espera a que una solicitud se complete antes de cancelarla) en segundos.

Predeterminado: `global timeout`
 Rango: 1-30

Véase también

- Instrucciones para configurar plugins
- Descripción de los parámetros de configuración generales de Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)

Complemento de Docker

Resumen

Los parámetros de configuración permiten personalizar el comportamiento del plugin de Docker.

Se recomienda especificarlos en su propio archivo de configuración (por ejemplo, `docker.conf`), en lugar de en el archivo de configuración del agent 2. Luego, use la directiva Include para incluir este archivo en la configuración de Zabbix agent 2.

Todos los parámetros son **opcionales**, salvo que se indique explícitamente que el parámetro es obligatorio.

Esta página está estructurada de la siguiente manera:

- Resumen de parámetros (para ver los detalles completos de un parámetro, haga clic en su nombre)
- Detalles de los parámetros

Parameter	Description
Plugins.Docker.Endpoint	Especifica la ubicación del socket UNIX del daemon de Docker..
Plugins.Docker.Timeout	Especifica el tiempo de espera para la ejecución de la solicitud.

Tenga en cuenta que:

- Los valores predeterminados reflejan los valores por defecto del proceso, no los valores de los archivos de configuración suministrados;
- Los valores admiten variables de entorno;
- Zabbix solo admite archivos de configuración en codificación UTF-8 sin BOM;
- Los comentarios que comienzan con ”#” solo se admiten al principio de la línea.

Detalles del parámetro

Plugins.Docker.Endpoint

Especifique la ubicación del socket UNIX del daemon de Docker.
 Debe contener un esquema (solo se admite `unix://`).

Predeterminado: `unix:///var/run/docker.sock`

Plugins.Docker.Timeout

Especifica el tiempo de espera para la ejecución de la solicitud (la duración, en segundos, que se espera a que una solicitud se complete antes de interrumpirla).

Predeterminado: `global timeout`
 Rango: 1–30

Véase también

- Descripción de los parámetros de configuración generales de Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Instrucciones para configurar plugins

Complemento de Memcached

Resumen

Los parámetros de configuración permiten personalizar el comportamiento del plugin de Memcached.

Se recomienda especificarlos en su propio archivo de configuración (por ejemplo, `memcached.conf`), en lugar de hacerlo en el archivo de configuración de agent 2. Luego use la directiva Include para incluir este archivo en la configuración de Zabbix agent 2.

Todos los parámetros son **opcionales**, salvo que se indique explícitamente que el parámetro es obligatorio.

Esta página está estructurada de la siguiente manera:

- Resumen de parámetros (para ver los detalles completos de un parámetro, haga clic en su nombre)
- Detalles de los parámetros

Parameter	Description
Plugins.Memcached.Default.Password	Especifica la contraseña predeterminada para conectarse a Memcached.
Plugins.Memcached.Default.Uri	Especifica la URI predeterminada para conectarse a Memcached.
Plugins.Memcached.Default.User	Especifica el nombre de usuario predeterminado para conectarse a Memcached.
Plugins.Memcached.KeepAlive	Establece el tiempo máximo de espera antes de cerrar las conexiones del plugin no utilizadas.
Plugins.Memcached.Sessions.<SessionName>.Password	Especifica la contraseña de la sesión con nombre para conectarse a Memcached.
Plugins.Memcached.Sessions.<SessionName>.Uri	Especifica la URI de la sesión con nombre para conectarse a Memcached.
Plugins.Memcached.Sessions.<SessionName>.User	Especifica el nombre de usuario de la sesión con nombre para conectarse a Memcached.
Plugins.Memcached.Timeout	Establece el tiempo de espera de ejecución de la solicitud.

Tenga en cuenta que:

- Los valores predeterminados reflejan los valores predeterminados del proceso, no los valores de los archivos de configuración suministrados;
- Los valores admiten variables de entorno;
- Zabbix solo admite archivos de configuración en codificación UTF-8 sin BOM;
- Los comentarios que comienzan con ”#” solo se admiten al principio de la línea.

Detalles del parámetro

Plugins.Memcached.Default.Password

Especifique la contraseña predeterminada para conectarse a Memcached.
 Se usa si no se especifica ningún valor en una clave de item o en una sesión con nombre.

Plugins.Memcached.Default.Uri

Especifique el URI predeterminado para conectarse a Memcached.
 Se usa si no se especifica ningún valor en una clave de item o sesión con nombre.
 Las credenciales incrustadas se ignorarán.
 Debe coincidir con el formato URI. Esquemas admitidos: `tcp`, `unix`.
 Se puede omitir un esquema.
 Se puede omitir un puerto (predeterminado=11211).

Predeterminado: `tcp://localhost:11211`

Ejemplos:

```
tcp://localhost:11211
localhost
unix:/var/run/memcached.sock
```

Plugins.Memcached.Default.User

Especifique el nombre de usuario predeterminado para conectarse a Memcached.
 Se usa si no se especifica ningún valor en una clave de item o sesión con nombre.

Plugins.Memcached.KeepAlive

Establece el tiempo máximo de espera (en segundos) antes de que se cierren las conexiones del plugin no utilizadas.

Predeterminado: 300
 Rango: 60–900

Plugins.Memcached.Sessions.<SessionName>.Password

Especifique la contraseña de la sesión con nombre para conectarse a Memcached.
 <SessionName> - defina el nombre de una sesión para usarlo en las claves de item.

Plugins.Memcached.Sessions.<SessionName>.Uri

Especifique el URI de la sesión con nombre para conectarse a Memcached.
 <SessionName> - defina el nombre de una sesión para usarlo en las claves de item.
 Las credenciales incrustadas se ignorarán.
 Debe coincidir con el formato URI. Esquemas admitidos: `tcp`, `unix`.
 Se puede omitir un esquema.
 Se puede omitir un puerto (predeterminado=11211).

Ejemplos:

```
tcp://localhost:11211
localhost
unix:/var/run/memcached.sock
```

Plugins.Memcached.Sessions.<SessionName>.User

Especifique el nombre de usuario de la sesión para conectarse a Memcached.
 <SessionName> - defina el nombre de una sesión para usarlo en las claves de item.

Plugins.Memcached.Timeout

Establece el tiempo de espera de ejecución de la solicitud (la duración, en segundos, que se espera a que una solicitud se complete antes de interrumpirla).

Predeterminado: `global timeout`
 Rango: 1-30

Véase también:

- Descripción de los parámetros de configuración generales de Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Instrucciones para configurar plugins

Complemento de MongoDB

Resumen

Los parámetros de configuración permiten personalizar el comportamiento del plugin de MongoDB.

Se recomienda especificarlos en su propio archivo de configuración (por ejemplo, `mongodb.conf`), en lugar de en el archivo de configuración de agent 2. Luego, use la directiva Include para incluir este archivo en la configuración de Zabbix agent 2.

El plugin de MongoDB es un plugin cargable y está disponible y completamente descrito en el repositorio del plugin de MongoDB.

Todos los parámetros son **opcionales**, salvo que se indique explícitamente que el parámetro es obligatorio.

Al conectarse a MongoDB, el plugin comprueba su configuración en un orden específico, que determina qué valor usar. El orden, de mayor a menor prioridad, es el siguiente:

1. Parámetros especificados en el URI de conexión (por ejemplo, los parámetros de consulta definidos después de `?` en `Plugins.MongoDB.Sessions.<session_name>.Uri` tienen la máxima prioridad).
2. Parámetros en la clave de item de Zabbix agent 2 (por ejemplo, mongodb.collection.stats).
3. Parámetros definidos en una sesión con nombre (por ejemplo, `Plugins.MongoDB.Sessions.<SessionName>.User`, `Plugins.MongoDB.Sessions.<SessionName>.Password`).
4. Valores predeterminados (por ejemplo, `Plugins.MongoDB.Default.User`, `Plugins.MongoDB.Default.Password`).

Por ejemplo:

- Si especifica la base de datos para autenticar una conexión al server de MongoDB en el parámetro `*.<SessionName>.Uri` (por ejemplo, `Plugins.MongoDB.Sessions.<session_name>.Uri=mongodb://user:password@127.0.0.1:27017/?authSo` entonces el plugin considerará que el parámetro `*.<SessionName>.AuthSource` está establecido en `admin`, incluso si el parámetro `*.<SessionName>.AuthSource` está configurado con un valor diferente.
- Si especifica el URI en el parámetro `*.<SessionName>.Uri` (por ejemplo, `Plugins.MongoDB.Sessions.<session_name>.Uri=m` pero proporciona el nombre de usuario y la contraseña en la clave de item de Zabbix agent 2, el plugin usará el nombre de usuario y la contraseña de la clave de item, incluso si `Plugins.MongoDB.Sessions.<SessionName>.User` y `Plugins.MongoDB.Sessions.<SessionName>.Password` están configurados.

> **Attention:**
> Si se usa el esquema `mongodb+srv://`, el URI puede recuperar parámetros de un registro DNS TXT. Los parámetros obtenidos de este registro TXT anulan todas las demás configuraciones, incluidos los parámetros del URI, los parámetros de la clave de item de Zabbix Agent 2 y los parámetros de sesión.
 Solo puede existir un registro TXT para el nombre de host usado en la cadena de conexión.

Esta página está estructurada de la siguiente manera:

- Resumen de parámetros (para ver los detalles completos de un parámetro, haga clic en su nombre)
- Detalles de parámetros

Parameter	Description
Plugins.MongoDB.Default.AuthMechanism	Especifica el mecanismo de autenticación predeterminado para conectarse al server de MongoDB.
Plugins.MongoDB.Default.AuthSource	Especifica la base de datos predeterminada para autenticar la conexión al server de MongoDB.
Plugins.MongoDB.Default.Dbstr	Deshabilita la detección automática de nodos del conjunto de réplicas de MongoDB.
Plugins.MongoDB.Default.Password	Especifica la contraseña predeterminada.
Plugins.MongoDB.Default.ReplicaSet	Especifica el nombre predeterminado del conjunto de réplicas, usado para la detección automática de nodos del conjunto de réplicas de MongoDB.
Plugins.MongoDB.Default.ReadPreference	Especifica el orden predeterminado para conectarse a nodos del conjunto de réplicas de MongoDB o como alternativa.
Plugins.MongoDB.Default.TLSConnect	Especifica el nivel de cifrado requerido para las comunicaciones entre Zabbix agent 2 y las bases de datos monitorizadas.
Plugins.MongoDB.Default.TLSCAFile	Especifica la ruta completa del archivo que contiene los certificados de la CA de nivel superior para la verificación del certificado del par.
Plugins.MongoDB.Default.TLSCertFile	Especifica la ruta completa del archivo que contiene el certificado del agent o la cadena de certificados.
Plugins.MongoDB.Default.TLSKeyFile	Especifica la ruta completa del archivo que contiene la clave privada de la base de datos.
Plugins.MongoDB.Default.Uri	Especifica el URI predeterminado.
Plugins.MongoDB.Default.User	Especifica el nombre de usuario predeterminado.
Plugins.MongoDB.KeepAlive	Establece el tiempo máximo de espera antes de cerrar las conexiones del plugin no utilizadas.
Plugins.MongoDB.Sessions.<SessionName>.AuthMechanism	Especifica el mecanismo de autenticación de la sesión con nombre para conectarse al server de MongoDB.
Plugins.MongoDB.Sessions.<SessionName>.AuthSource	Especifica la base de datos de la sesión con nombre para autenticar la conexión al server de MongoDB.
Plugins.MongoDB.Sessions.<SessionName>.Dbstr	Deshabilita la detección automática de nodos del conjunto de réplicas de MongoDB para la sesión con nombre.
Plugins.MongoDB.Sessions.<SessionName>.Password	Especifica la contraseña de la sesión con nombre.
Plugins.MongoDB.Sessions.<SessionName>.ReplicaSet	Especifica el nombre del conjunto de réplicas de la sesión con nombre, usado para la detección automática de nodos del conjunto de réplicas de MongoDB.
Plugins.MongoDB.Sessions.<SessionName>.ReadPreference	Especifica el orden de la sesión con nombre para conectarse a nodos del conjunto de réplicas de MongoDB o como alternativa.
Plugins.MongoDB.Sessions.<SessionName>.TLSConnect	Especifica el nivel de cifrado requerido para las comunicaciones entre Zabbix agent 2 y las bases de datos monitorizadas para la sesión con nombre.
Plugins.MongoDB.Sessions.<SessionName>.TLSCAFile	Especifica la ruta completa del archivo que contiene los certificados de la CA de nivel superior para la verificación del certificado del par para la sesión con nombre.
Plugins.MongoDB.Sessions.<SessionName>.TLSCertFile	Especifica la ruta completa del archivo que contiene el certificado del agent o la cadena de certificados para la sesión con nombre.
Plugins.MongoDB.Sessions.<SessionName>.TLSKeyFile	Especifica la ruta completa del archivo que contiene la clave privada de la base de datos para la sesión con nombre.
Plugins.MongoDB.Sessions.<SessionName>.Uri	Especifica el URI de la sesión con nombre.
Plugins.MongoDB.Sessions.<SessionName>.User	Especifica el nombre de usuario de la sesión con nombre.
Plugins.MongoDB.System.Path	Especifica la ruta al ejecutable del plugin.
Plugins.MongoDB.Timeout	Establece el tiempo de espera de ejecución de la solicitud.

Tenga en cuenta que:

- Los valores predeterminados reflejan los valores predeterminados del proceso, no los valores de los archivos de configuración suministrados.
- Los valores admiten variables de entorno.
- Zabbix solo admite archivos de configuración en codificación UTF-8 sin BOM.
- Los comentarios que comienzan con ”#” solo se admiten al principio de la línea.

Detalles del parámetro

Plugins.MongoDB.Default.AuthMechanism

Especifique el mecanismo de autenticación predeterminado para conectarse al server de MongoDB.
 <SessionName> - defina el nombre de una sesión para usar en las claves de item.

Rango: `SCRAM-SHA-1`, `SCRAM-SHA-256` o `MONGODB-X509`

Plugins.MongoDB.Default.AuthSource

Especifique la base de datos predeterminada para autenticar la conexión al server de MongoDB.
 <SessionName> - define el nombre de una sesión para usarlo en las claves de item.

Predeterminado: `admin`

Si se utiliza el mecanismo de autenticación `MONGODB-X509`, este parámetro se establece automáticamente en `$external`; modificarlo a cualquier otro valor provocará un error al iniciarse.

Plugins.MongoDB.Default.Direct

Deshabilita el descubrimiento automático de nodos del conjunto de réplicas de MongoDB.
 <SessionName> - define el nombre de una sesión para usar en las claves de item.

Predeterminado: `false`
 Rango: `true` - conectarse solo al server especificado, ignorando a los demás miembros del conjunto de réplicas, `false` - descubrir automáticamente todos los nodos del conjunto de réplicas.

Si se usa el esquema `tcp://` (obsoleto), el parámetro se establece en `true` de forma predeterminada.

Plugins.MongoDB.Default.Password

Especifique la contraseña predeterminada.
 <SessionName> - defina el nombre de una sesión para usar en las claves de item.

Plugins.MongoDB.Default.ReplicaSet

Especifique el nombre predeterminado del conjunto de réplicas, utilizado para el descubrimiento automático de nodos del conjunto de réplicas de MongoDB.
 <SessionName> - defina el nombre de una sesión para usarlo en las claves de item.
 Si este parámetro se usa con el esquema `mongodb+srv://`, se comportará como si se hubieran proporcionado varias URL; un servidor DNS puede tener un registro TXT que puede anular este parámetro (o cualquier otro).
 No está disponible si el parámetro `Plugins.MongoDB.Sessions.<SessionName>.Direct` está establecido en `true`; el agent no podrá iniciarse.

Plugins.MongoDB.Default.ReadPreference

Especifique el orden predeterminado para conectarse a los nodos del conjunto de réplicas de MongoDB o la alternativa.
 <SessionName>: defina el nombre de una sesión para utilizarla en las claves de los item.
 No disponible si el parámetro `Plugins.MongoDB.Sessions.<SessionName>.Direct` está establecido en `true`; el agent no podrá iniciarse.

Rango: `primary`, `primaryPreferred`, `secondary`, `secondaryPreferred`, `nearest`, `PrimaryMode`, `PrimaryPreferredMode`, `SecondaryMode`, `SecondaryPreferredMode` o `NearestMode`

Se recomiendan los valores `primary`, `primaryPreferred`, `secondary`, `secondaryPreferred` y `nearest`. Los valores que terminan en `Mode` son compatibles como alias heredados.

Plugins.MongoDB.Default.TLSConnect

Especifique el nivel de cifrado requerido para las comunicaciones entre Zabbix agent 2 y las bases de datos monitorizadas.
 <SessionName> - define el nombre de una sesión para usarlo en las claves de item.
 Compatible desde la versión 1.2.1 del plugin.

Rango: `required` - requiere conexión TLS, `verify_ca` - verifica certificados, `verify_full` - verifica certificados y dirección IP

Plugins.MongoDB.Default.TLSCAFile

Especifique la ruta completa del archivo que contiene los certificados de CA de nivel superior para la verificación del certificado del par, utilizados para las comunicaciones cifradas entre Zabbix agent 2 y las bases de datos supervisadas.
 <SessionName> - define el nombre de una sesión para usar en las claves de item.

Obligatorio: `yes`, si `Plugins.MongoDB.Sessions.<SessionName>.TLSConnect` está establecido en `verify_ca` o `verify_full`

Plugins.MongoDB.Default.TLSCertFile

Especifique la ruta completa del archivo que contiene el certificado del agent o la cadena de certificados, utilizado para las comunicaciones cifradas entre Zabbix agent 2 y las bases de datos monitorizadas.
 <SessionName> - define el nombre de una sesión para usarlo en las claves de item.

Obligatorio: `yes`, si se especifica `Plugins.MongoDB.Sessions.<SessionName>.TLSKeyFile`

Plugins.MongoDB.Default.TLSKeyFile

Especifique la ruta completa del archivo que contiene la clave privada de la base de datos, utilizada para las comunicaciones cifradas entre Zabbix agent 2 y las bases de datos monitorizadas.
 <SessionName> - define el nombre de una sesión para usar en las claves de item.

Obligatorio: `yes`, si se especifica `Plugins.MongoDB.Sessions.<SessionName>.TLSCertFile`

Plugins.MongoDB.Default.Uri

Especifique el URI predeterminado.
 <SessionName> - defina el nombre de una sesión para usarlo en las claves de item.
 Debe coincidir con el formato del URI.
 Esquemas compatibles: `mongodb://` y `mongodb+srv://`; el esquema `tcp://` está obsoleto y se mantiene por compatibilidad con configuraciones existentes.
 Se puede omitir un puerto (default=27017).
Ejemplos: “

Predeterminado: `tcp://127.0.0.1:27017`

Ejemplos:

```
mongodb://127.0.0.1:27017
mongodb+srv://example.com
localhost
```

Plugins.MongoDB.Default.User

Especifique el nombre de usuario predeterminado.
 <SessionName> - defina el nombre de una sesión para usarlo en las claves de item.

Plugins.MongoDB.KeepAlive

Establece el tiempo máximo de espera (en segundos) antes de que se cierren las conexiones del plugin no utilizadas.

Predeterminado: 300
 Rango: 60-900

Plugins.MongoDB.Sessions.<SessionName>.AuthMechanism

Especifica el mecanismo de autenticación de la sesión con nombre para conectarse al server de MongoDB.
 <SessionName> - define el nombre de una sesión para usarlo en las claves de item.

Rango: `SCRAM-SHA-1`, `SCRAM-SHA-256` o `MONGODB-X509`

Plugins.MongoDB.Sessions.<SessionName>.AuthSource

Especifique la base de datos de la sesión con nombre para autenticar la conexión al server de MongoDB.
 <SessionName> - define el nombre de una sesión para usarlo en las claves de item.

Predeterminado: `admin`

Si se utiliza el mecanismo de autenticación `MONGODB-X509`, este parámetro se establece automáticamente en `$external`; modificarlo a cualquier otro valor provocará un error al iniciar.

Plugins.MongoDB.Sessions.<SessionName>.Direct

Deshabilita el descubrimiento automático de nodos del replica set de MongoDB para la sesión con nombre.
 <SessionName> - define el nombre de una sesión para usarlo en las claves de item.

Predeterminado: `false`
 Rango: `true` - conectarse solo al server especificado, ignorando a otros miembros del replica set, `false` - descubrir automáticamente todos los nodos del replica set.

Si se usa el esquema `tcp://` (obsoleto), el parámetro se establece en `true` de forma predeterminada.

Plugins.MongoDB.Sessions.<SessionName>.Password

Especifique la contraseña de la sesión con nombre.
 <SessionName> - defina el nombre de una sesión para usarlo en las claves de item.

Plugins.MongoDB.Sessions.<SessionName>.ReplicaSet

Especifica el nombre del replica set de la sesión con nombre, utilizado para el descubrimiento automático de nodos del replica set de MongoDB.
 <SessionName> - define el nombre de una sesión para usarlo en las claves de item.
 Si este parámetro se usa con el esquema `mongodb+srv://`, se comportará como si se hubieran proporcionado varias URL; un servidor DNS puede tener un registro TXT que puede anular este parámetro (o cualquier otro).
 No está disponible si el parámetro `Plugins.MongoDB.Sessions.<SessionName>.Direct` está establecido en `true`; el agent no podrá iniciarse.

Plugins.MongoDB.Sessions.<SessionName>.ReadPreference

Especifique el orden de sesión con nombre para conectarse a los nodos del conjunto de réplicas de MongoDB o utilizar la alternativa.
 <SessionName> - defina el nombre de una sesión para utilizarla en las claves de los item.
 No disponible si el parámetro `Plugins.MongoDB.Sessions.<SessionName>.Direct` está establecido en `true`; el agent no podrá iniciarse.

Rango: `primary`, `primaryPreferred`, `secondary`, `secondaryPreferred`, `nearest`, `PrimaryMode`, `PrimaryPreferredMode`, `SecondaryMode`, `SecondaryPreferredMode` o `NearestMode`

Se prefieren los valores `primary`, `primaryPreferred`, `secondary`, `secondaryPreferred` y `nearest`. Los valores que terminan en `Mode` se admiten como alias de compatibilidad heredados.

Plugins.MongoDB.Sessions.<SessionName>.TLSConnect

Especifique el nivel de cifrado requerido para las comunicaciones entre Zabbix agent 2 y las bases de datos monitorizadas para la sesión indicada.
 <SessionName> - define el nombre de una sesión para usarlo en las claves de item.
 Compatible desde la versión 1.2.1 del plugin.

Rango: `required` - requiere conexión TLS, `verify_ca` - verifica los certificados, `verify_full` - verifica los certificados y la dirección IP

Plugins.MongoDB.Sessions.<SessionName>.TLSCAFile

Especifique la ruta completa del archivo que contiene los certificados de la CA de nivel superior para la verificación del certificado del par, utilizados para las comunicaciones cifradas entre Zabbix agent 2 y las bases de datos monitorizadas para la sesión con nombre.
 <SessionName> - defina el nombre de una sesión para usarlo en las claves de item.

Obligatorio: `yes`, si `Plugins.MongoDB.Sessions.<SessionName>.TLSConnect` está establecido en `verify_ca` o `verify_full`

Plugins.MongoDB.Sessions.<SessionName>.TLSCertFile

Especifique la ruta completa del archivo que contiene el certificado del agent o la cadena de certificados, utilizado para las comunicaciones cifradas entre Zabbix agent 2 y las bases de datos monitorizadas para la sesión indicada.
 <SessionName> - defina el nombre de una sesión para usarlo en las claves de item.

Obligatorio: `yes`, si se especifica `Plugins.MongoDB.Sessions.<SessionName>.TLSKeyFile`

Plugins.MongoDB.Sessions.<SessionName>.TLSKeyFile

Especifique la ruta completa del archivo que contiene la clave privada de la base de datos, utilizada para comunicaciones cifradas entre Zabbix agent 2 y las bases de datos monitorizadas para la sesión especificada.
 <SessionName> - define el nombre de una sesión para usarlo en las claves de item.

Obligatorio: `yes`, si se especifica `Plugins.MongoDB.Sessions.<SessionName>.TLSCertFile`

Plugins.MongoDB.Sessions.<SessionName>.Uri

Especifique el URI de la sesión con nombre.
 <SessionName> - defina el nombre de una sesión para usarlo en las claves de item.
 Debe coincidir con el formato URI.
 Esquemas compatibles: `mongodb://` y `mongodb+srv://`; el esquema `tcp://` está obsoleto y se mantiene por compatibilidad con configuraciones existentes.
 Se puede omitir un puerto (predeterminado=27017).
Ejemplos: “

Predeterminado: `tcp://127.0.0.1:27017`

Ejemplos:

```
mongodb://127.0.0.1:27017
mongodb+srv://example.com
localhost
```

Plugins.MongoDB.Sessions.<SessionName>.User

Especifica el nombre de usuario de la sesión indicada.
 <SessionName> - define el nombre de una sesión para usar en las claves de item.

Plugins.MongoDB.System.Path

Especifique la ruta al ejecutable del plugin de MongoDB.

Ejemplo de uso: `Plugins.MongoDB.System.Path=/usr/libexec/zabbix/zabbix-agent2-plugin-mongodb`

Plugins.MongoDB.Timeout

Establece el tiempo de espera de ejecución de la solicitud (la duración, en segundos, que se espera a que una solicitud se complete antes de interrumpirla).

Predeterminado: `global timeout`
 Rango: `1-30`

Opciones

Parameter	Description
-V --version	Muestra la versión del plugin y la información de la licencia.
-h --help	Muestra la información de ayuda (atajo).
-t, --test <item key>	Inicia el plugin para pruebas (se ignora la configuración del plugin).

Véase también

- Descripción de los parámetros de configuración generales de Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Instrucciones para configurar plugins

Complemento de MySQL

Resumen

Los parámetros de configuración permiten personalizar el comportamiento del plugin de MySQL.

Se recomienda especificarlos en su propio archivo de configuración (por ejemplo, `mysql.conf`), en lugar de en el archivo de configuración de agent 2. Luego, use la directiva Include para incluir este archivo en la configuración de Zabbix agent 2.

Todos los parámetros son **opcionales**, salvo que se indique explícitamente que el parámetro es obligatorio.

Esta página está estructurada de la siguiente manera:

- Resumen de parámetros (para ver los detalles completos de un parámetro, haga clic en su nombre)
- Detalles de parámetros

Parameter	Description
Plugins.Mysql.CallTimeout	Especifica el tiempo máximo, en segundos, que se esperará para que se complete una solicitud.
Plugins.Mysql.CustomQueriesPath	Especifica la ruta completa del directorio que contiene archivos `.sql` con consultas personalizadas.
Plugins.Mysql.CustomQueriesEnabled	Habilita la ejecución de la clave de item `mysql.custom.query`.
Plugins.Mysql.Default.Password	Especifica la contraseña predeterminada.
Plugins.Mysql.Default.TLSConnect	Especifica el nivel predeterminado de cifrado para las comunicaciones entre Zabbix agent 2 y las bases de datos monitorizadas.
Plugins.Mysql.Default.TLSCAFile	Especifica la ruta completa del archivo que contiene los certificados de las CA de nivel superior para la verificación del certificado del par.
Plugins.Mysql.Default.TLSCertFile	Especifica la ruta completa del archivo que contiene el certificado del agent o la cadena de certificados.
Plugins.Mysql.Default.TLSKeyFile	Especifica la ruta completa del archivo que contiene la clave privada de la base de datos.
Plugins.Mysql.Default.Uri	Especifica la URI predeterminada.
Plugins.Mysql.Default.User	Especifica el nombre de usuario predeterminado.
Plugins.Mysql.KeepAlive	Especifica el tiempo máximo de espera (en segundos) antes de cerrar las conexiones del plugin que no se estén usando.
Plugins.Mysql.Sessions.<SessionName>.Password	Especifica la contraseña de la sesión con nombre.
Plugins.Mysql.Sessions.<SessionName>.TLSConnect	Especifica el nivel de cifrado de la sesión con nombre para las comunicaciones entre Zabbix agent 2 y las bases de datos monitorizadas.
Plugins.Mysql.Sessions.<SessionName>.TLSCAFile	Especifica la ruta completa del archivo que contiene los certificados de las CA de nivel superior para la verificación del certificado del par de la sesión con nombre.
Plugins.Mysql.Sessions.<SessionName>.TLSCertFile	Especifica la ruta completa del archivo que contiene el certificado del agent o la cadena de certificados de la sesión con nombre.
Plugins.Mysql.Sessions.<SessionName>.TLSKeyFile	Especifica la ruta completa del archivo que contiene la clave privada de la base de datos de la sesión con nombre.
Plugins.Mysql.Sessions.<SessionName>.Uri	Especifica la URI de la sesión con nombre.
Plugins.Mysql.Sessions.<SessionName>.User	Especifica el nombre de usuario de la sesión con nombre.
Plugins.Mysql.Timeout	Establece el tiempo máximo de espera, en segundos, cuando se debe establecer una conexión.

Tenga en cuenta que:

- Los valores predeterminados reflejan los valores predeterminados del proceso, no los valores de los archivos de configuración suministrados;
- Los valores admiten variables de entorno;
- Zabbix solo admite archivos de configuración en codificación UTF-8 sin BOM;
- Los comentarios que comienzan con ”#” solo se admiten al principio de la línea.

Detalles del parámetro

Plugins.Mysql.CallTimeout

Especifica el tiempo máximo, en segundos, que se esperará a que se complete una solicitud.

Valor predeterminado: `global timeout`
 Rango: 1-30

Plugins.Mysql.CustomQueriesPath

Especifique la ruta completa del directorio que contiene archivos `.sql` con consultas personalizadas.

Valores predeterminados:
`/usr/local/share/zabbix/custom-queries/mysql` - para sistemas Unix,
`*:\Program Files\Zabbix Agent 2\Custom Queries\Mysql` - para sistemas Windows (donde `*` es el nombre de la unidad de la variable de entorno `ProgramFiles`)

Plugins.Mysql.CustomQueriesEnabled

Habilita la ejecución de la clave de item `mysql.custom.query`.
 Si está deshabilitado, no se cargarán consultas desde la ruta del directorio de consultas personalizadas.

Predeterminado: `false`
 Rango: `true/false`

Plugins.Mysql.Default.Password

Especifique la contraseña predeterminada para un server MySQL protegido.

Plugins.Mysql.Default.TLSConnect

Especifica el nivel predeterminado de cifrado para las comunicaciones entre Zabbix agent 2 y las bases de datos monitorizadas.

Rango:
required - requiere conexión TLS;
verify_ca - verifica los certificados;
verify_full - verifica los certificados y la dirección IP

Plugins.Mysql.Default.TLSCAFile

Especifique la ruta completa del archivo que contiene los certificados de CA de nivel superior para la verificación del certificado del par, utilizado para las comunicaciones cifradas entre Zabbix agent 2 y las bases de datos monitorizadas.

Obligatorio: `yes`, si `Plugins.Mysql.Default.TLSConnect` está establecido en `verify_ca` o `verify_full`

Plugins.Mysql.Default.TLSCertFile

Especifique la ruta completa del archivo que contiene el certificado del agent o la cadena de certificados, utilizado para las comunicaciones cifradas entre Zabbix agent 2 y las bases de datos monitorizadas.

Obligatorio: `yes`, si se especifica `Plugins.Mysql.Default.TLSKeyFile`

Plugins.Mysql.Default.TLSKeyFile

Especifique la ruta completa del archivo que contiene la clave privada de la base de datos, utilizada para las comunicaciones cifradas entre Zabbix agent 2 y las bases de datos monitorizadas.

Obligatorio: `yes`, si se especifica `Plugins.Mysql.Default.TLSCertFile`

Plugins.Mysql.Default.Uri

Especifique el URI predeterminado para conectarse al server de MySQL.
 Las credenciales incrustadas se ignorarán.
 Esquemas admitidos: `tcp`, `unix`
 El esquema se puede omitir.
 El puerto se puede omitir (default=3306).

Ejemplos:

```
tcp://localhost:3306
localhost
unix:/var/run/mysql.sock
```

Plugins.Mysql.Default.User

Especifica el nombre de usuario predeterminado para un servidor MySQL protegido.

Plugins.Mysql.KeepAlive

Especifica el tiempo máximo de espera (en segundos) antes de que se cierren las conexiones del plugin que no se estén utilizando.

Valor predeterminado: 300
 Rango: 60-900

Plugins.Mysql.Sessions.<SessionName>.Password

Especifique la contraseña de la sesión con nombre para un servidor MySQL protegido.
 <SessionName> - defina el nombre de una sesión para usarlo en las claves de item.

Plugins.Mysql.Sessions.<SessionName>.TLSConnect

Especifique el nivel de cifrado de la sesión con nombre para las comunicaciones entre Zabbix agent 2 y las bases de datos monitorizadas.
 <SessionName> - define el nombre de una sesión para usarlo en las claves de item.

Rango:
required - requiere conexión TLS;
verify_ca - verifica los certificados;
verify_full - verifica los certificados y la dirección IP

Plugins.Mysql.Sessions.<SessionName>.TLSCAFile

Especifique la ruta completa del archivo que contiene los certificados de CA de nivel superior para la verificación del certificado del par en una sesión con nombre, utilizado para las comunicaciones cifradas entre Zabbix agent 2 y las bases de datos supervisadas.
 <SessionName> - defina el nombre de una sesión para usarlo en las claves de item.

Obligatorio: `yes`, si `Plugins.Mysql.Sessions.<SessionName>.TLSConnect` está establecido en `verify_ca` o `verify_full`

Plugins.Mysql.Sessions.<SessionName>.TLSCertFile

Especifique la ruta completa del archivo que contiene el certificado del agent de la sesión con nombre o la cadena de certificados, utilizado para las comunicaciones cifradas entre Zabbix agent 2 y las bases de datos monitorizadas.
 <SessionName> - define el nombre de una sesión para usarlo en las claves de item.

Obligatorio: `yes`, si se especifica `Plugins.Mysql.Sessions.<SessionName>.TLSKeyFile`

Plugins.Mysql.Sessions.<SessionName>.TLSKeyFile

Especifique la ruta completa del archivo que contiene la clave privada de la base de datos de la sesión con nombre, utilizada para las comunicaciones cifradas entre Zabbix agent 2 y las bases de datos monitorizadas.
 <SessionName> - defina el nombre de una sesión para usarlo en las claves de item.

Obligatorio: `yes`, si se especifica `Plugins.Mysql.Sessions.<SessionName>.TLSCertFile`

Plugins.Mysql.Sessions.<SessionName>.Uri

Especifique la URI de la sesión con nombre para conectarse al server MySQL.
 <SessionName> - defina el nombre de una sesión para usarlo en las claves de item.
 Las credenciales incrustadas se ignorarán.
 Esquemas admitidos: `tcp`, `unix`
 El esquema se puede omitir.
 El puerto se puede omitir (predeterminado=3306).

Ejemplos:

```
tcp://localhost:3306
localhost
unix:/var/run/mysql.sock
```

Plugins.Mysql.Sessions.<SessionName>.User

Especifique el nombre de usuario de la sesión con nombre para un servidor MySQL protegido.
 <SessionName> - defina el nombre de una sesión para usarlo en las claves de item.

Plugins.Mysql.Timeout

Establece el tiempo máximo, en segundos, de espera cuando se debe establecer una conexión.

Predeterminado: `global timeout`
 Rango: 1–30

Véase también

- Descripción de los parámetros de configuración generales de Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Instrucciones para configurar plugins

Complemento de Oracle

Descripción general

Los parámetros de configuración permiten personalizar el comportamiento del plugin de Oracle.

Se recomienda especificarlos en su propio archivo de configuración (por ejemplo, `oracle.conf`), en lugar de hacerlo en el archivo de configuración del agent 2. Luego, use la directiva Include para incluir este archivo en la configuración de Zabbix agent 2.

Todos los parámetros son **opcionales**, salvo que se indique explícitamente que el parámetro es obligatorio.

Esta página está estructurada de la siguiente manera:

- Descripción general de los parámetros (para ver los detalles completos de un parámetro, haga clic en su nombre)
- Detalles de los parámetros

Parameter	Description
Plugins.Oracle.CallTimeout	El tiempo máximo de espera, en segundos, para que se complete una solicitud. Actualmente, este parámetro no tiene efecto; en su lugar se utiliza la configuración de item timeout.
Plugins.Oracle.ConnectTimeout	Especifica el tiempo máximo de espera, en segundos, para que se establezca una conexión.
Plugins.Oracle.CustomQueriesPath	Especifica la ruta completa del directorio que contiene archivos `.sql` con consultas personalizadas.
Plugins.Oracle.CustomQueriesEnabled	Habilita la ejecución de la clave de item `oracle.custom.query`.
Plugins.Oracle.Default.Password	Especifica la contraseña predeterminada.
Plugins.Oracle.Default.Service	Especifica el nombre de servicio predeterminado.
Plugins.Oracle.Default.URI	Especifica el URI predeterminado.
Plugins.Oracle.Default.User	Especifica el nombre de usuario predeterminado.
Plugins.Oracle.KeepAlive	Establece el tiempo máximo de espera, en segundos, antes de cerrar las conexiones del plugin que no se estén usando.
Plugins.Oracle.ResolveTNS	Especifica cómo interpretar la cadena de conexión (ConnString) para conectarse al server de Oracle.
Plugins.Oracle.Sessions.<SessionName>.Password	Especifica la contraseña de la sesión con nombre.
Plugins.Oracle.Sessions.<SessionName>.Service	Especifica el nombre de servicio de la sesión con nombre.
Plugins.Oracle.Sessions.<SessionName>.URI	Especifica el URI de la sesión con nombre.
Plugins.Oracle.Sessions.<SessionName>.User	Especifica el nombre de usuario de la sesión con nombre.

Tenga en cuenta lo siguiente:

- Los valores predeterminados reflejan los valores predeterminados del proceso, no los valores de los archivos de configuración suministrados.
- Los valores admiten variables de entorno.
- Zabbix solo admite archivos de configuración en codificación UTF-8 sin BOM.
- Los comentarios que comienzan con "#" solo se admiten al principio de la línea.

Detalles del parámetro

Plugins.Oracle.CallTimeout

El tiempo máximo de espera, en segundos, para que se complete una solicitud.
 Este parámetro actualmente no tiene efecto. El tiempo máximo de espera lo determina en su lugar el item timeout.

Predeterminado: `Timeout`[1]
 Rango: 1-30

Plugins.Oracle.ConnectTimeout

Especifique el tiempo máximo de espera, en segundos, para que se establezca una conexión.

Predeterminado: `Timeout`[1]
 Rango: 1-30

Plugins.Oracle.CustomQueriesPath

Especifique la ruta completa del directorio que contiene archivos `.sql` con consultas personalizadas.

Valores predeterminados:
`/usr/local/share/zabbix/custom-queries/oracle` - para sistemas Unix,
`*:\Program Files\Zabbix Agent 2\Custom Queries\Oracle` - para sistemas Windows (donde `*` es el nombre de la unidad de la variable de entorno `ProgramFiles`)

Ejemplo:

```
/etc/zabbix/oracle/sql
```

Plugins.Oracle.CustomQueriesEnabled

Habilita la ejecución de la clave de item `oracle.custom.query`.
 Si está deshabilitado, no se cargarán consultas desde la ruta del directorio de consultas personalizadas.

Predeterminado: `false`
 Rango: `true/false`

Plugins.Oracle.Default.Password

Especifique la contraseña predeterminada para conectarse a Oracle.
 Se usa si no se especifica ningún valor en una clave de item o en una sesión con nombre.

Plugins.Oracle.Default.Service

Especifica el nombre de servicio predeterminado para conectarse a Oracle (SID no es compatible).
 Se usa si no se especifica ningún valor en una clave de item o en una sesión con nombre.

Plugins.Oracle.Default.Uri

Especifique el URI predeterminado para conectarse a Oracle.
 Se usa si no se especifica ningún valor en una clave de item o en una sesión con nombre.
 Las credenciales incrustadas se ignorarán.
 Debe coincidir con el formato URI.
 Esquemas compatibles: `tcp`, `tcps`
 El esquema puede omitirse.
 El puerto puede omitirse (default=1521).
 También es posible especificar la clave TNS o el valor TNS como cadena de conexión. El valor TNS debe componerse sin espacios en blanco.

Ejemplos:

```
tcp://127.0.0.1:1521
localhost
zbx_tns_example (TNS key)
(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=localhost)(PORT=1521))(CONNECT_DATA=(SERVICE_NAME=xe))) (TNS val
```

Plugins.Oracle.Default.User

Especifique el nombre de usuario predeterminado para conectarse a Oracle.
 Se usa si no se especifica ningún valor en una clave de item o en una sesión con nombre.

Plugins.Oracle.KeepAlive

Establece el tiempo máximo de espera (en segundos) antes de que se cierren las conexiones del plugin no utilizadas.

Predeterminado: 300
 Rango: 60-900

Plugins.Oracle.ResolveTNS

Especifique cómo interpretar la cadena de conexión (ConnString) para conectarse al server de Oracle.
 Si se establece en `false`, el esquema y el puerto predeterminados se agregarán a ConnString (si faltan).
 Si la opción se establece en `true`, el esquema y el puerto predeterminados se omitirán (a menos que se especifiquen explícitamente en ConnString), y ConnString se pasará al cliente de Oracle tal como está. Si el cliente de Oracle encuentra ConnString en el archivo `tnsnames.ora`, se usará la descripción de conexión encontrada para conectarse al server de Oracle.

Predeterminado: `true`
 Rango: `true/false`

Plugins.Oracle.Sessions.<SessionName>.Password

Especifique la contraseña de la sesión con nombre para conectarse a Oracle.
 <SessionName> - defina el nombre de una sesión para usarlo en las claves de item.

Plugins.Oracle.Sessions.<SessionName>.Service

Especifique el nombre del servicio de sesión para conectarse a Oracle (SID no es compatible).
 <SessionName> - defina el nombre de una sesión para usarlo en las claves de item.

Plugins.Oracle.Sessions.<SessionName>.Uri

Especifique el URI de la sesión con nombre para conectarse a Oracle.
 <SessionName> - defina el nombre de una sesión para usarlo en las claves de item.
 Las credenciales incrustadas se ignorarán.
 Debe coincidir con el formato URI.
 Esquemas admitidos: `tcp`, `tcps`
 El esquema puede omitirse.
 El puerto puede omitirse (default=1521).
 También es posible especificar la clave TNS o el valor TNS como cadena de conexión. El valor TNS debe componerse sin espacios en blanco.

Ejemplos:

```
tcp://127.0.0.1:1521
localhost
zbx_tns_example (TNS key)
(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=localhost)(PORT=1521))(CONNECT_DATA=(SERVICE_NAME=xe))) (TNS val
```

Plugins.Oracle.Sessions.<SessionName>.User

Especifique el nombre de usuario de la sesión nombrada para conectarse a Oracle.
 <SessionName> - defina el nombre de una sesión para usarlo en las claves de item.

Notas al pie

[1] El tiempo de espera está definido por la opción `Timeout` en el archivo de configuración (`zabbix_agent2.conf`) de Zabbix agent 2 (UNIX) o Zabbix agent 2 (Windows).

Véase también

- Descripción de los parámetros de configuración generales de Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Instrucciones para configurar plugins

Complemento de PostgreSQL

Resumen

Los parámetros de configuración permiten personalizar el comportamiento del plugin de PostgreSQL.

Se recomienda especificarlos en su propio archivo de configuración (por ejemplo, `postgresql.conf`), en lugar de en el archivo de configuración de agent 2. Luego use la directiva Include para incluir este archivo en la configuración de Zabbix agent 2.

El plugin de PostgreSQL es un plugin cargable y está disponible y completamente descrito en el repositorio del plugin de PostgreSQL.

Todos los parámetros son **opcionales**, a menos que se indique explícitamente que el parámetro es obligatorio.

Esta página está estructurada de la siguiente manera:

- Resumen de parámetros (para ver los detalles completos de un parámetro, haga clic en su nombre)
- Detalles de parámetros

Parameter	Description
Plugins.PostgreSQL.CallTimeout	Establece el tiempo máximo de espera, en segundos, para que una solicitud se complete.
Plugins.PostgreSQL.CustomQueriesPath	Especifica la ruta completa del directorio que contiene archivos `.sql` con consultas personalizadas.
Plugins.PostgreSQL.CustomQueriesEnabled	Habilita la ejecución de la clave de item `postgresql.custom.query`.
Plugins.PostgreSQL.Default.CacheMode	Especifica el modo de caché predeterminado para la conexión de PostgreSQL.
Plugins.PostgreSQL.Default.Database	Especifica el nombre de la base de datos predeterminada.
Plugins.PostgreSQL.Default.Password	Especifica la contraseña predeterminada.
Plugins.PostgreSQL.Default.TLSCAFile	Especifica la ruta completa del archivo que contiene el certificado de la(s) CA de nivel superior para la verificación del certificado de pares.
Plugins.PostgreSQL.Default.TLSCertFile	Especifica la ruta completa del archivo que contiene el certificado o la cadena de certificados de PostgreSQL.
Plugins.PostgreSQL.Default.TLSConnect	Especifica el nivel predeterminado de cifrado para las comunicaciones entre Zabbix agent 2 y las bases de datos monitorizadas.
Plugins.PostgreSQL.Default.TLSKeyFile	Especifica la ruta completa del archivo que contiene la clave privada de PostgreSQL.
Plugins.PostgreSQL.Default.Uri	Especifica el URI predeterminado.
Plugins.PostgreSQL.Default.User	Especifica el nombre de usuario predeterminado.
Plugins.PostgreSQL.KeepAlive	Establece el tiempo máximo de espera, en segundos, antes de cerrar las conexiones del plugin no utilizadas.
Plugins.PostgreSQL.Sessions.<SessionName>.CacheMode	Especifica el modo de caché de la sesión con nombre para la conexión de PostgreSQL.
Plugins.PostgreSQL.Sessions.<SessionName>.Database	Especifica el nombre de la base de datos de la sesión con nombre.
Plugins.PostgreSQL.Sessions.<SessionName>.Password	Especifica la contraseña de la sesión con nombre.
Plugins.PostgreSQL.Sessions.<SessionName>.TLSCAFile	Especifica la ruta completa del archivo que contiene el certificado de la(s) CA de nivel superior para la verificación del certificado de pares.
Plugins.PostgreSQL.Sessions.<SessionName>.TLSCertFile	Especifica la ruta completa del archivo que contiene el certificado o la cadena de certificados de PostgreSQL.
Plugins.PostgreSQL.Sessions.<SessionName>.TLSConnect	Especifica el nivel de cifrado de la sesión con nombre para las comunicaciones entre Zabbix agent 2 y las bases de datos monitorizadas.
Plugins.PostgreSQL.Sessions.<SessionName>.TLSKeyFile	Especifica la ruta completa del archivo que contiene la clave privada de PostgreSQL.
Plugins.PostgreSQL.Sessions.<SessionName>.Uri	Especifica el URI de la sesión con nombre.
Plugins.PostgreSQL.Sessions.<SessionName>.User	Especifica el nombre de usuario de la sesión con nombre.

|Plugins.PostgreSQL.System.Path|Especifica la ruta al ejecutable del plugin de PostgreSQL.| |Plugins.PostgreSQL.Timeout|Establece el tiempo de espera de ejecución de la solicitud.|

Tenga en cuenta que:

- Los valores predeterminados reflejan los valores predeterminados del proceso, no los valores de los archivos de configuración suministrados.
- Los valores admiten variables de entorno;
- Zabbix solo admite archivos de configuración en codificación UTF-8 sin BOM.
- Los comentarios que comienzan con ”#” solo se admiten al principio de la línea.

Detalles del parámetro

Plugins.PostgreSQL.CallTimeout

Establece el tiempo máximo de espera, en segundos, para que se complete una solicitud.

Predeterminado: `global timeout`
 Rango: 1-30

Plugins.PostgreSQL.CustomQueriesPath

Especifique la ruta completa del directorio que contiene archivos `.sql` con consultas personalizadas.

Valores predeterminados:
`/usr/local/share/zabbix/custom-queries/postgreSQL` - para sistemas Unix,
`*:\Program Files\Zabbix Agent 2\Custom Queries\PostgreSQL` - para sistemas Windows (donde `*` es el nombre de la unidad de la variable de entorno `ProgramFiles`)

Ejemplo:

```
/etc/zabbix/oracle/sql
```

Plugins.PostgreSQL.CustomQueriesEnabled

Habilita la ejecución de la clave de item `postgresql.custom.query`.
 Si está deshabilitado, no se cargarán consultas desde la ruta del directorio de consultas personalizadas.

Predeterminado: `false`
 Rango: `true/false`

Plugins.PostgreSQL.Default.CacheMode

Especifica el modo de caché predeterminado para la conexión de PostgreSQL.

Predeterminado: `prepare`
 Rango:
`prepare` - crea sentencias preparadas en el server de PostgreSQL;
`describe` - usa la sentencia preparada anónima para describir una sentencia sin crear una sentencia en el server.

Tenga en cuenta que `describe` es útil principalmente cuando el entorno no permite sentencias preparadas, como al ejecutar un pooler de conexiones como PgBouncer.

Plugins.PostgreSQL.Default.Database

Especifique la base de datos predeterminada para conectarse a PostgreSQL.
 Se usa si no se especifica ningún valor en una clave de item o en una sesión con nombre.

Plugins.PostgreSQL.Default.Password

Especifique la contraseña predeterminada para conectarse a PostgreSQL.
 Se usa si no se especifica ningún valor en una clave de item o en una sesión con nombre.

Plugins.PostgreSQL.Default.TLSCAFile

Especifique la ruta completa del archivo que contiene el certificado de la CA de nivel superior para la verificación del certificado del par en las comunicaciones cifradas entre Zabbix agent 2 y las bases de datos monitorizadas.
 Se usa si no se especifica ningún valor en una sesión con nombre.

Obligatorio: `yes`, si `Plugins.PostgreSQL.Default.TLSConnect` está establecido en `verify_ca` o `verify_full`

Plugins.PostgreSQL.Default.TLSCertFile

Especifique la ruta completa del archivo que contiene el certificado de PostgreSQL o la cadena de certificados para las comunicaciones cifradas entre Zabbix agent 2 y las bases de datos monitorizadas.
 Se usa si no se especifica ningún valor en una sesión con nombre.

Obligatorio: `yes`, si se especifica `Plugins.PostgreSQL.Default.TLSKeyFile`

Plugins.PostgreSQL.Default.TLSConnect

Especifique el nivel predeterminado de cifrado para las comunicaciones entre Zabbix agent 2 y las bases de datos monitorizadas.
 Se usa si no se especifica ningún valor en una sesión con nombre.

Rango:
`required` - conectarse usando TLS como modo de transporte sin comprobaciones de identidad;
`verify_ca` - conectarse usando TLS y verificar el certificado;
`verify_full` - conectarse usando TLS, verificar el certificado y verificar que la identidad de la base de datos (CN) especificada por DBHost coincida con su certificado.

Un tipo de cifrado no definido significa una conexión sin cifrar.

Plugins.PostgreSQL.Default.TLSKeyFile

Especifique la ruta completa del archivo que contiene la clave privada de PostgreSQL para las comunicaciones cifradas entre Zabbix agent 2 y las bases de datos monitorizadas.
 Se usa si no se especifica ningún valor en una sesión con nombre.

Obligatorio: `yes`, si se especifica `Plugins.PostgreSQL.Default.TLSCertFile`

Plugins.PostgreSQL.Default.Uri

Especifique la URI predeterminada para conectarse a PostgreSQL.
 Se usa si no se especifica ningún valor en una clave de item o en una sesión con nombre.
 Las credenciales incrustadas se ignorarán.
 Debe coincidir con el formato de URI.
 Esquemas compatibles: `tcp`, `unix`

Ejemplos:

```
tcp://127.0.0.1:5432
tcp://localhost
unix:/var/run/postgresql/.s.PGSQL.5432
```

Plugins.PostgreSQL.Default.User

Especifique el nombre de usuario predeterminado para conectarse a PostgreSQL.
 Se utiliza si no se especifica ningún valor en una clave de item o en una sesión con nombre.

Plugins.PostgreSQL.KeepAlive

Establece el tiempo máximo de espera (en segundos) antes de que se cierren las conexiones del plugin que no se estén utilizando.

Plugins.PostgreSQL.Sessions.<SessionName>.CacheMode

Especifique el modo de caché de sesión con nombre para la conexión de PostgreSQL.
 <SessionName> - defina el nombre de una sesión para usarlo en las claves de item.

Predeterminado: `prepare`
 Rango:
`prepare` - crea sentencias preparadas en el server de PostgreSQL;
`describe` - usa la sentencia preparada anónima para describir una sentencia sin crear una sentencia en el server.

Tenga en cuenta que `describe` es principalmente útil cuando el entorno no permite sentencias preparadas, como al ejecutar un pooler de conexiones como PgBouncer.

Plugins.PostgreSQL.Sessions.<SessionName>.Database

Especifica la base de datos de la sesión con nombre para conectarse a PostgreSQL.
 <SessionName> - define el nombre de una sesión para usarlo en las claves de item.

Plugins.PostgreSQL.Sessions.<SessionName>.Password

Especifique la contraseña de la sesión con nombre para conectarse a PostgreSQL.
 <SessionName> - define el nombre de una sesión para usar en las claves de item.
 Debe coincidir con el formato de la contraseña.

Plugins.PostgreSQL.Sessions.<SessionName>.TLSCAFile

Especifique la ruta completa del archivo que contiene el certificado de la CA de nivel superior para la verificación del certificado del par en las comunicaciones cifradas entre Zabbix agent 2 y las bases de datos monitorizadas.
 <SessionName> - defina el nombre de una sesión para usarlo en las claves de item.

Obligatorio: `yes`, si `Plugins.PostgreSQL.Sessions.<SessionName>.TLSConnect` está establecido en `verify_ca` o `verify_full`

Plugins.PostgreSQL.Sessions.<SessionName>.TLSCertFile

Especifique la ruta completa del archivo que contiene el certificado de PostgreSQL o la cadena de certificados para las comunicaciones cifradas entre Zabbix agent 2 y las bases de datos monitorizadas.
 <SessionName> - defina el nombre de una sesión para usarlo en las claves de item.

Obligatorio: `yes`, si se especifica `Plugins.PostgreSQL.Sessions.<SessionName>.TLSKeyFile`

Plugins.PostgreSQL.Sessions.<SessionName>.TLSConnect

Especifica el nivel de cifrado con nombre de la sesión para las comunicaciones entre Zabbix agent 2 y las bases de datos monitorizadas.
 <SessionName> - define el nombre de una sesión para usarlo en las claves de item.

Rango:
`required` - conectarse usando TLS como modo de transporte sin comprobaciones de identidad;
 `verify_ca` - conectarse usando TLS y verificar el certificado;
 `verify_full` - conectarse usando TLS, verificar el certificado y verificar que la identidad de la base de datos (CN) especificada por DBHost coincida con su certificado.

Un tipo de cifrado no definido significa una conexión sin cifrar.

Plugins.PostgreSQL.Sessions.<SessionName>.TLSKeyFile

Especifique la ruta completa del archivo que contiene la clave privada de PostgreSQL para las comunicaciones cifradas entre Zabbix agent 2 y las bases de datos supervisadas.
 <SessionName> - defina el nombre de una sesión para usarlo en las claves de item.

Obligatorio: `yes`, si se especifica `Plugins.PostgreSQL.Sessions.<SessionName>.TLSCertFile`.

Plugins.PostgreSQL.Sessions.<SessionName>.Uri

Especifique la URI de la sesión con nombre para conectarse a PostgreSQL.
 <SessionName> - defina el nombre de una sesión para usarlo en las claves de item.
 Las credenciales incrustadas se ignorarán.
 Debe coincidir con el formato de URI.
 Esquemas compatibles: `tcp`, `unix`

Ejemplos:

```
tcp://127.0.0.1:5432
tcp://localhost
unix:/var/run/postgresql/.s.PGSQL.5432
```

Plugins.PostgreSQL.Sessions.<SessionName>.User

Especifique el nombre de usuario de la sesión con nombre para conectarse a PostgreSQL.
 <SessionName> - defina el nombre de una sesión para usarlo en las claves de item.

Plugins.PostgreSQL.System.Path

Especifique la ruta al ejecutable del plugin de PostgreSQL.

Obligatorio: `yes`

Ejemplo:

```
Plugins.PostgreSQL.System.Path=/usr/libexec/zabbix/zabbix-agent2-plugin-postgresql
```

Plugins.PostgreSQL.Timeout

Establece el tiempo de espera para la ejecución de la solicitud (la duración, en segundos, que se espera a que una solicitud se complete antes de finalizarla).

Predeterminado: `global timeout`
 Rango: 1–30

Opciones

Parameter	Description
-V --version	Imprime la versión del plugin y la información de la licencia.
-h --help	Imprime la información de ayuda (abreviado).
-t, --test <item key>	Inicia el plugin para pruebas (se ignora la configuración del plugin).

Véase también

- Descripción de los parámetros de configuración generales de Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Instrucciones para configurar plugins

Complemento de Redis

Resumen

Los parámetros de configuración permiten personalizar el comportamiento del plugin de Redis.

Se recomienda especificarlos en su propio archivo de configuración (por ejemplo, `redis.conf`), en lugar de en el archivo de configuración de agent 2. Luego, use la directiva Include para incluir este archivo en la configuración de Zabbix agent 2.

Todos los parámetros son **opcionales**, salvo que se indique explícitamente que el parámetro es obligatorio.

Esta página está estructurada de la siguiente manera:

- Resumen de parámetros (para ver los detalles completos de un parámetro, haga clic en su nombre)
- Detalles de los parámetros

Parameter	Description
Plugins.Redis.Default.Password	Especifique la contraseña predeterminada para conectarse a Redis.
Plugins.Redis.Default.TLSConnect	Especifique el nivel de cifrado requerido para las comunicaciones de Zabbix agent 2.
Plugins.Redis.Default.TLSCAFile	Especifique la ruta completa del archivo que contiene los certificados de la(s) CA de nivel superior.
Plugins.Redis.Default.TLSCertFile	Especifique la ruta completa del archivo que contiene el certificado de agent.
Plugins.Redis.Default.TLSKeyFile	Especifique la ruta completa del archivo que contiene la clave privada de agent.
Plugins.Redis.Default.Uri	Especifique la URI predeterminada para conectarse a Redis.

Parameter	Description
Plugins.Redis.Default.User	Especifique el nombre de usuario predeterminado para conectarse a Redis.
Plugins.Redis.KeepAlive	Establezca el tiempo máximo de espera (en segundos) antes de cerrar las conexiones del plugin no utilizadas.
Plugins.Redis.Sessions.<SessionName>.Password	Especifique la contraseña de la sesión con nombre para conectarse a Redis.
Plugins.Redis.Sessions.<SessionName>.TLSConnect	Especifique el nivel de cifrado requerido para las comunicaciones de Zabbix agent 2 para la sesión con nombre.
Plugins.Redis.Sessions.<SessionName>.TLSCAFile	Especifique la ruta completa del archivo que contiene los certificados de la(s) CA de nivel superior para la sesión con nombre.
Plugins.Redis.Sessions.<SessionName>.TLSCertFile	Especifique la ruta completa del archivo que contiene el certificado de agent para la sesión con nombre.
Plugins.Redis.Sessions.<SessionName>.TLSKeyFile	Especifique la ruta completa del archivo que contiene la clave privada de agent para la sesión con nombre.
Plugins.Redis.Sessions.<SessionName>.Uri	Especifique la URI de la sesión con nombre para conectarse a Redis.
Plugins.Redis.Sessions.<SessionName>.User	Especifique el nombre de usuario de la sesión con nombre para conectarse a Redis.
Plugins.Redis.Timeout	Establezca el tiempo de espera de ejecución de la solicitud.

Tenga en cuenta que:

- Los valores predeterminados reflejan los valores predeterminados del proceso, no los valores de los archivos de configuración suministrados.
- Los valores admiten variables de entorno.
- Zabbix solo admite archivos de configuración en codificación UTF-8 sin BOM.
- Los comentarios que comienzan con ”#” solo se admiten al principio de la línea.

Detalles del parámetro

Plugins.Redis.Default.Password

Especifique la contraseña predeterminada para conectarse a Redis.
 Se usa si no se especifica ningún valor en una clave de item o en una sesión con nombre.

Plugins.Redis.Default.TLSConnect

Especifica el nivel de cifrado requerido para las comunicaciones de Zabbix agent 2.
 Un valor vacío o no establecido se trata como sin TLS.

Rango: `required` - cifrado, no verificado (solo para pruebas), `verify_ca` - cifrado, certificado del servidor, `verify_full` - cifrado, certificado del servidor y nombre del servidor verificado con SAN

Plugins.Redis.Default.TLSCAFile

Especifique la ruta completa del archivo que contiene los certificados de la(s) CA de nivel superior.

Plugins.Redis.Default.TLSCertFile

Especifique la ruta completa del archivo que contiene el certificado del agent.

Plugins.Redis.Default.TLSKeyFile

Especifique la ruta completa del archivo que contiene la clave privada del agent.

Plugins.Redis.Default.Uri

Especifique el URI predeterminado para conectarse a Redis.
 Se usa si no se especifica ningún valor en una clave de item o en una sesión con nombre.
 Las credenciales incrustadas se ignorarán. Debe coincidir con el formato de URI.
 Esquemas admitidos: `tcp`, `unix`
 Se puede omitir un esquema.
 Se puede omitir un puerto (predeterminado=6379).

Predeterminado: `tcp://localhost:6379`

Ejemplos:

```
tcp://localhost:6379
localhost
unix:/var/run/redis.sock
```

Plugins.Redis.Default.User

Especifique el nombre de usuario predeterminado para conectarse a Redis.
 Se usa si no se especifica ningún valor en una clave de item o en una sesión con nombre.

Predeterminado: `default`

Plugins.Redis.KeepAlive

Establece el tiempo máximo de espera (en segundos) antes de que se cierren las conexiones del plugin no utilizadas.

Predeterminado: 300
 Rango: 60-900

Plugins.Redis.Sessions.<SessionName>.Password

Especifique la contraseña de la sesión con nombre para conectarse a Redis.
 <SessionName> - defina el nombre de la sesión que se usará en las claves de item.

Plugins.Redis.Sessions.<SessionName>.TLSConnect

Especifique el nivel de cifrado requerido para las comunicaciones de Zabbix agent 2 para la sesión con nombre.
 <SessionName> - define el nombre de la sesión que se usará en las claves de item.
 Un valor vacío o no establecido se trata como ausencia de TLS.

Rango: `required` - cifrado, no verificado (solo para pruebas), `verify_ca` - cifrado, certificado del servidor, `verify_full` - cifrado, certificado del servidor y nombre del servidor verificado con SAN

Plugins.Redis.Sessions.<SessionName>.TLSCAFile

Especifique la ruta completa del archivo que contiene los certificados de la CA de nivel superior para la sesión nombrada.
 <SessionName> - define el nombre de la sesión que se usará en las claves de item.

Obligatorio: `yes`, si `Plugins.Redis.Sessions.<SessionName>.TLSConnect` está configurado en `verify_ca` o `verify_full`

Plugins.Redis.Sessions.<SessionName>.TLSCertFile

Especifique la ruta completa del archivo que contiene el certificado del agent para la sesión con nombre.
 <SessionName> - defina el nombre de la sesión que se usará en las claves de item.

Plugins.Redis.Sessions.<SessionName>.TLSKeyFile

Especifique la ruta completa del archivo que contiene la clave privada del agent para la sesión con nombre.
 <SessionName> - define el nombre de la sesión que se usará en las claves de item.

Plugins.Redis.Sessions.<SessionName>.Uri

Especifique el URI de la sesión con nombre para conectarse a Redis.
 <SessionName> - defina el nombre de la sesión que se usará en las claves de item.
 Las credenciales incrustadas se ignorarán. Debe coincidir con el formato de URI.
 Esquemas admitidos: `tcp`, `unix`
 Se puede omitir un esquema.
 Se puede omitir un puerto (default=6379).

Default: `tcp://localhost:6379`

Examples:

```
tcp://localhost:6379
localhost
unix:/var/run/redis.sock
```

Plugins.Redis.Sessions.<SessionName>.User

Especifique el nombre de usuario de la sesión con nombre para conectarse a Redis.
 <SessionName> - defina el nombre de la sesión que se usará en las claves de item.

Plugins.Redis.Timeout

Establece el tiempo de espera de ejecución de la solicitud (la duración, en segundos, que se espera a que una solicitud se complete antes de interrumpirla).

Predeterminado: `global timeout`
 Rango: 1-30

Véase también

- Descripción de los parámetros de configuración generales de Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Instrucciones para configurar plugins

Complemento Ember+

Overview

Los parámetros de configuración permiten personalizar el comportamiento del plugin Ember+.

Se recomienda especificarlos en su propio archivo de configuración (por ejemplo, `emberplus.conf`), en lugar de en el archivo de configuración de agent 2. Luego, use la directiva Include para incluir este archivo en la configuración de Zabbix agent 2.

El plugin Ember+ es un plugin cargable y está disponible y completamente descrito en el repositorio del plugin Ember+.

Actualmente, este plugin solo está disponible para compilarse desde el código fuente (tanto para Unix como para Windows).

Todos los parámetros son **opcionales**, salvo que se indique explícitamente que el parámetro es obligatorio.

Esta página está estructurada de la siguiente manera:

- Resumen de parámetros (para ver los detalles completos de un parámetro, haga clic en su nombre)
- Detalles de parámetros

Parameter	Description
Plugins.EmberPlus.Default.Uri	Especifica la URI predeterminada para conectarse a Ember+.
Plugins.EmberPlus.KeepAlive	Establece el tiempo máximo de espera antes de cerrar las conexiones del plugin no utilizadas.
Plugins.EmberPlus.Sessions.<SessionName>.Uri	Especifica la URI de la sesión con nombre para conectarse a Ember+.
Plugins.EmberPlus.System.Path	Especifica la ruta al ejecutable del plugin Ember+.
Plugins.EmberPlus.Timeout	Establece el tiempo de espera de ejecución de la solicitud.

Tenga en cuenta que:

- Los valores predeterminados reflejan los valores predeterminados del proceso, no los valores de los archivos de configuración suministrados.
- Los valores admiten variables de entorno;
- Zabbix solo admite archivos de configuración en codificación UTF-8 sin BOM.
- Los comentarios que comienzan con "#" solo se admiten al principio de la línea.

Detalles del parámetro

Plugins.EmberPlus.Default.Uri

Especifique el URI predeterminado para conectarse a Ember+.
 El único esquema compatible es `tcp://`. Se puede omitir un esquema. Las credenciales incrustadas se ignorarán.

Predeterminado: `tcp://localhost:9998`

Plugins.EmberPlus.KeepAlive

Establece el tiempo máximo de espera (en segundos) antes de cerrar las conexiones del plugin que no se estén utilizando.

Predeterminado: 300
 Rango: 60-900

Plugins.EmberPlus.Sessions.<SessionName>.Uri

Especifique la URI de la sesión con nombre para conectarse a Ember+.
 El único esquema compatible es `tcp://`. Se puede omitir un esquema. Las credenciales incrustadas se ignorarán.
 <SessionName> - defina el nombre de una sesión para usarlo en las claves de item.

Predeterminado: `tcp://localhost:9998`

Plugins.EmberPlus.System.Path

Especifique la ruta al ejecutable del plugin Ember+.

Ejemplo: `Plugins.EmberPlus.System.Path=/usr/libexec/zabbix/zabbix-agent2-plugin-ember-plus`

Plugins.EmberPlus.Timeout

Establezca la duración, en segundos, que se esperará a que un server responda al conectarse por primera vez y en las operaciones posteriores de la sesión.

Predeterminado: `global timeout`
 Rango: 1-30

Opciones

Parameter	Description
-V --version	Muestra la versión del plugin y la información de la licencia.
-h --help	Muestra la información de ayuda (abreviado).
-t, --test <item key>	Inicia el plugin para pruebas (se ignora la configuración del plugin).

Véase también:

- Descripción de los parámetros de configuración generales de Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Instrucciones para configurar plugins

Complemento Modbus

Resumen

Los parámetros de configuración permiten personalizar el comportamiento del plugin Modbus.

Se recomienda especificarlos en su propio archivo de configuración (por ejemplo, `modbus.conf`), en lugar de hacerlo en el archivo de configuración de agent 2. Luego, use la directiva Include para incluir este archivo en la configuración de Zabbix agent 2.

Todos los parámetros son **opcionales**, salvo que se indique explícitamente que el parámetro es obligatorio.

Esta página está estructurada de la siguiente manera:

- Resumen de parámetros (para ver los detalles completos de un parámetro, haga clic en su nombre)
- Detalles de parámetros

Parameter	Description
Plugins.Modbus.Sessions.<SessionName>.Endpoint	Especifica el endpoint de la sesión con nombre para conectarse a Modbus.
Plugins.Modbus.Sessions.<SessionName>.SlaveID	Especifica el ID de esclavo de la sesión con nombre para conectarse a Modbus.
Plugins.Modbus.Sessions.<SessionName>.Timeout	Establece el tiempo de espera de la sesión con nombre en segundos.

Tenga en cuenta que:

- Los valores predeterminados reflejan los valores predeterminados del proceso, no los valores de los archivos de configuración distribuidos;
- Los valores admiten variables de entorno;
- Zabbix solo admite archivos de configuración en codificación UTF-8 sin BOM;
- Los comentarios que comienzan con "#" solo se admiten al principio de la línea.

Detalles del parámetro

Plugins.Modbus.Sessions.<SessionName>.Endpoint

Especifique el endpoint de la sesión con nombre para conectarse a Modbus.
 La cadena de conexión consta de un esquema de protocolo, una dirección de host y un puerto o nombre de puerto serie y atributos.
 <SessionName> - defina el nombre de una sesión para usarlo en las claves de item.

Plugins.Modbus.Sessions.<SessionName>.SlaveID

Especifique el ID de esclavo de la sesión con nombre para conectarse a Modbus.
 <SessionName> - defina el nombre de una sesión para usarlo en las claves de item.

Ejemplo: `Plugins.Modbus.Sessions.MB1.SlaveID=20`

Nota que este parámetro de sesión con nombre solo se comprueba si el valor proporcionado en el parámetro de ID de esclavo de la item key está vacío.

Plugins.Modbus.Sessions.<SessionName>.Timeout

Establece el tiempo de espera de la sesión especificada en segundos.
 <SessionName> - define el nombre de una sesión para usarlo en las claves de item.

Ejemplo: `Plugins.Modbus.Sessions.MB1.Timeout=2`

Si necesita establecer el tiempo de espera de ejecución de la solicitud (cuánto tiempo esperar a que una solicitud se complete antes de finalizarla), use el formulario de configuración de item.

Véase también

- Descripción de los parámetros de configuración generales de Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Instrucciones para configurar plugins

Complemento MQTT

Resumen

Los parámetros de configuración permiten personalizar el comportamiento del plugin MQTT.

Se recomienda especificarlos en su propio archivo de configuración (por ejemplo, `mqtt.conf`), en lugar de en el archivo de configuración de agent 2. Luego, use la directiva Include para incluir este archivo en la configuración de Zabbix agent 2.

Todos los parámetros son **opcionales**, salvo que se indique explícitamente que el parámetro es obligatorio.

Esta página está estructurada de la siguiente manera:

- Resumen de parámetros (para ver los detalles completos de un parámetro, haga clic en su nombre)
- Detalles de parámetros

Parameter	Description
Plugins.MQTT.Default.Password	Especifique la contraseña predeterminada para conectarse a MQTT.
Plugins.MQTT.Default.TLSCAFile	Especifique la ruta completa del archivo que contiene los certificados de la(s) CA de nivel superior para la verificación del certificado del par.
Plugins.MQTT.Default.TLSCertFile	Especifique la ruta completa del archivo que contiene el certificado del agent o la cadena de certificados.
Plugins.MQTT.Default.TLSKeyFile	Especifique la ruta completa del archivo que contiene la clave privada de MQTT.
Plugins.MQTT.Default.Topic	Especifique el tema predeterminado para la suscripción MQTT.
Plugins.MQTT.Default.Url	Especifique la cadena de conexión predeterminada del broker MQTT.
Plugins.MQTT.Default.User	Especifique el nombre de usuario predeterminado para conectarse a MQTT.
Plugins.MQTT.Sessions.<Session Name>.Password	Especifique la contraseña de la sesión con nombre para conectarse a MQTT.
Plugins.MQTT.Sessions.<Session Name>.TLSCAFile	Especifique la ruta completa del archivo que contiene los certificados de la(s) CA de nivel superior para la verificación del certificado del par para la sesión con nombre.
Plugins.MQTT.Sessions.<Session Name>.TLSCertFile	Especifique la ruta completa del archivo que contiene el certificado del agent o la cadena de certificados para la sesión con nombre.
Plugins.MQTT.Sessions.<Session Name>.TLSKeyFile	Especifique la ruta completa del archivo que contiene la clave privada de MQTT para la sesión con nombre.
Plugins.MQTT.Sessions.<Session Name>.Topic	Especifique el tema de la sesión con nombre para la suscripción MQTT.
Plugins.MQTT.Sessions.<Session Name>.Url	Especifique la cadena de conexión del broker MQTT para la sesión con nombre.
Plugins.MQTT.Sessions.<Session Name>.User	Especifique el nombre de usuario de la sesión con nombre para conectarse a MQTT.

Tenga en cuenta que:

- Los valores predeterminados reflejan los valores predeterminados del proceso, no los valores de los archivos de configuración suministrados;
- Los valores admiten variables de entorno;
- Zabbix solo admite archivos de configuración en codificación UTF-8 sin BOM;
- Los comentarios que comienzan con ”#” solo se admiten al principio de la línea.

Detalles del parámetro

Plugins.MQTT.Default.Password

Especifique la contraseña predeterminada para conectarse a MQTT.
 Se usa si no se especifica ningún valor en una clave de item o en una sesión con nombre.

Plugins.MQTT.Default.TLSCAFile

Especifique la ruta completa del archivo que contiene los certificados de CA de nivel superior para la verificación del certificado del par en las comunicaciones cifradas entre Zabbix agent 2 y el broker MQTT.
 Se usa si no se especifica ningún valor en una sesión con nombre.

Plugins.MQTT.Default.TLSCertFile

Especifique la ruta completa del archivo que contiene el certificado del agent o la cadena de certificados para las comunicaciones cifradas entre Zabbix agent 2 y el broker MQTT.
 Se usa si no se especifica ningún valor en una sesión con nombre.

Plugins.MQTT.Default.TLSKeyFile

Especifique la ruta completa del archivo que contiene la clave privada de MQTT para las comunicaciones cifradas entre Zabbix agent 2 y el broker MQTT.
 Se usa si no se especifica ningún valor en una sesión con nombre.

Plugins.MQTT.Default.Topic

Especifique el tema predeterminado para la suscripción MQTT.
 Se usa si no se especifica ningún valor en una clave de item o en una sesión con nombre.
 El tema puede contener comodines (”+”,”#”).

Ejemplos:

```
path/to/file
path/to/#
path/+/topic
```

Plugins.MQTT.Default.Url

Especifica la cadena de conexión predeterminada del broker MQTT.
 Se usa si no se especifica ningún valor en una clave de item o en una sesión con nombre.
 No debe incluir parámetros de consulta.
 Debe coincidir con el formato de URL.
 Esquemas admitidos: `tcp` (predeterminado), `ws`, `tls`
 Se puede omitir un esquema.
 Se puede omitir un puerto (predeterminado=1883)

Predeterminado: `tcp://localhost:1883`

Ejemplos:

```
tcp://host:1883
localhost
ws://host:8080
```

Plugins.MQTT.Default.User

Especifica el nombre de usuario predeterminado para conectarse a MQTT.
 Se usa si no se especifica ningún valor en una clave de item o en una sesión con nombre.

Plugins.MQTT.Sessions.<SessionName>.Password

Especifique la contraseña de la sesión con nombre para conectarse a MQTT.
 <SessionName> - defina el nombre de una sesión para usarlo en las claves de item.

Plugins.MQTT.Sessions.<SessionName>.TLSCAFile

Especifique la ruta completa del archivo que contiene los certificados de CA de nivel superior para la verificación del certificado del par para la sesión nombrada, utilizado para las comunicaciones cifradas entre Zabbix agent 2 y el broker MQTT.
 <SessionName> - define el nombre de una sesión para usarlo en las claves de item.

Plugins.MQTT.Sessions.<SessionName>.TLSCertFile

Especifique la ruta completa del archivo que contiene el certificado del agent o la cadena de certificados para la sesión especificada, utilizado para las comunicaciones cifradas entre Zabbix agent 2 y el broker MQTT.
 <SessionName> - defina el nombre de una sesión para usarlo en las claves de item.

Plugins.MQTT.Sessions.<SessionName>.TLSKeyFile

Especifique la ruta completa del archivo que contiene la clave privada de MQTT para la sesión indicada, utilizada para las comunicaciones cifradas entre Zabbix agent 2 y el broker MQTT.
 <SessionName> - defina el nombre de una sesión para usarlo en las claves de item.

Plugins.MQTT.Sessions.<SessionName>.Topic

Especifique el tema de la sesión con nombre para la suscripción MQTT.
 <SessionName> - defina el nombre de una sesión para usarlo en las claves de item.
 El tema puede contener comodines (”+”,”#”).

Ejemplos:

```
path/to/file
path/to/#
path/+/topic
```

Plugins.MQTT.Sessions.<SessionName>.Url

Especifica la cadena de conexión del broker MQTT para la sesión con nombre.
 <SessionName> - define el nombre de una sesión para usarlo en las claves de item.
 No debe incluir parámetros de consulta.
 Debe coincidir con el formato de URL.
 Esquemas admitidos: `tcp` (predeterminado), `ws`, `tls`
 Se puede omitir un esquema.
 Se puede omitir un puerto (predeterminado=1883)

Predeterminado: `tcp://localhost:1883`

Ejemplos:

```
tcp://host:1883
localhost
ws://host:8080
```

Plugins.MQTT.Sessions.<SessionName>.User

Especifica el nombre de usuario de la sesión con nombre para conectarse a MQTT.
 <SessionName> - define el nombre de una sesión para usar en las claves de item.

Si necesita establecer el tiempo de espera de ejecución de la solicitud (cuánto tiempo esperar a que una solicitud se complete antes de cerrarla), use el formulario de configuración del item.

Véase también

- Descripción de los parámetros generales de configuración de Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Instrucciones para configurar plugins

Complemento MSSQL

Overview

Los parámetros de configuración permiten personalizar el comportamiento del plugin MSSQL.

Se recomienda especificarlos en su propio archivo de configuración (por ejemplo, `mssql.conf`), en lugar de hacerlo en el archivo de configuración de agent 2. Luego use la directiva Include para incluir este archivo en la configuración de Zabbix agent 2.

El plugin MSSQL es un plugin cargable y está disponible y completamente descrito en el repositorio del plugin MSSQL.

Todos los parámetros son **opcionales**, salvo que se indique explícitamente que el parámetro es obligatorio.

Esta página está estructurada de la siguiente manera:

- Resumen de parámetros (para ver los detalles completos de un parámetro, haga clic en su nombre)
- Detalles de los parámetros

Parameter	Description
Plugins.MSSQL.CustomQueriesDir	Especifique la ruta al directorio que contiene archivos `.sql` definidos por el usuario con consultas personalizadas que el plugin puede ejecutar.
Plugins.MSSQL.CustomQueriesEnabled	Habilite la ejecución de la clave de item `mssql.custom.query`.
Plugins.MSSQL.Default.CACertPath	Especifique la ruta de archivo predeterminada al certificado de clave pública de la autoridad de certificación (CA) que emitió el certificado del server MSSQL.
Plugins.MSSQL.Default.Database	Especifique el nombre de base de datos predeterminado.
Plugins.MSSQL.Default.Encrypt	Especifique el nivel de cifrado predeterminado.
Plugins.MSSQL.Default.HostNameInCertificate	Especifique de forma predeterminada el nombre común (CN) del certificado del server MSSQL.
Plugins.MSSQL.Default.Password	Especifique la contraseña predeterminada.
Plugins.MSSQL.Default.TLSMinVersion	Especifique la versión mínima de TLS predeterminada.
Plugins.MSSQL.Default.TrustServerCertificate	Confíe en el certificado del server sin validarlo de forma predeterminada.
Plugins.MSSQL.Default.Uri	Especifique el URI predeterminado.
Plugins.MSSQL.Default.User	Especifique el nombre de usuario predeterminado.
Plugins.MSSQL.KeepAlive	Especifique el tiempo máximo de espera (en segundos) antes de cerrar las conexiones del plugin no utilizadas.
Plugins.MSSQL.Sessions.<SessionName>.CACertPath	Especifique la ruta de archivo de la sesión con nombre al certificado de clave pública de la autoridad de certificación (CA) que emitió el certificado del server MSSQL.
Plugins.MSSQL.Sessions.<SessionName>.Database	Especifique el nombre de base de datos de la sesión con nombre.
Plugins.MSSQL.Sessions.<SessionName>.Encrypt	Especifique el nivel de cifrado de la sesión con nombre.
Plugins.MSSQL.Sessions.<SessionName>.HostNameInCertificate	Especifique el nombre común (CN) del certificado del server MSSQL para la sesión con nombre.
Plugins.MSSQL.Sessions.<SessionName>.Password	Especifique la contraseña de la sesión con nombre.
Plugins.MSSQL.Sessions.<SessionName>.TLSMinVersion	Especifique la versión mínima de TLS de la sesión con nombre.
Plugins.MSSQL.Sessions.<SessionName>.TrustServerCertificate	Confíe en el certificado del server de la sesión con nombre sin validarlo.
Plugins.MSSQL.Sessions.<SessionName>.Uri	Especifique el URI de la sesión con nombre.
Plugins.MSSQL.Sessions.<SessionName>.User	Especifique el nombre de usuario de la sesión con nombre.
Plugins.MSSQL.System.Path	Especifique la ruta al ejecutable del plugin MSSQL.
Plugins.MSSQL.Timeout	Establezca la duración, en segundos, de la espera de respuesta de un server al conectarse por primera vez y en las operaciones posteriores de la sesión.

Tenga en cuenta que:

- Los valores predeterminados reflejan los valores predeterminados del proceso, no los valores de los archivos de configuración suministrados.
- Los valores admiten variables de entorno;
- Zabbix solo admite archivos de configuración en codificación UTF-8 sin BOM.
- Los comentarios que comienzan con "#" solo se admiten al principio de la línea.

Detalles del parámetro

Plugins.MSSQL.CustomQueriesDir

Especifique la ruta al directorio que contiene archivos `.sql` definidos por el usuario con consultas personalizadas que el plugin puede ejecutar.
 El plugin carga todos los archivos .sql disponibles en el directorio configurado al iniciarse. Esto significa que cualquier cambio en los archivos de consultas personalizadas no se reflejará hasta que se reinicie el plugin. El plugin se inicia y se detiene junto con Zabbix agent 2.

Valores predeterminados:
`/usr/local/share/zabbix/custom-queries/mssql` - para sistemas Unix,
`*:\Program Files\Zabbix Agent 2\Custom Queries\MSSQL` - para sistemas Windows (donde `*` es el nombre de la unidad de la variable de entorno `ProgramFiles`)

Plugins.MSSQL.CustomQueriesEnabled

Habilita la ejecución de la clave de item `mssql.custom.query`.
 Si está deshabilitado, no se cargarán consultas desde la ruta del directorio de consultas personalizadas.

Predeterminado: `false`
 Rango: `true/false`

Plugins.MSSQL.Default.CACertPath

Especifique la ruta de archivo predeterminada al certificado de clave pública de la autoridad de certificación (CA) que emitió el certificado del server MSSQL.
 El certificado debe estar en formato PEM.

Plugins.MSSQL.Default.Database

Especifique el nombre de la base de datos predeterminada.

Plugins.MSSQL.Default.Encrypt

Especifique el nivel de cifrado predeterminado.

Rango:
true - el envío de datos entre el plugin y el server está cifrado;
false - el envío de datos entre el plugin y el server no está cifrado más allá del paquete de inicio de sesión;
strict - el envío de datos entre el plugin y el server está cifrado de extremo a extremo usando TDS8;
disable - el envío de datos entre el plugin y el server no está cifrado.

Plugins.MSSQL.Default.HostNameInCertificate

Especifique de forma predeterminada el nombre común (CN) del certificado del server MSSQL.

Plugins.MSSQL.Default.Password

Especifique la contraseña predeterminada para un server MSSQL protegido.

Plugins.MSSQL.Default.TLSMinVersion

Especifica la versión mínima de TLS de forma predeterminada.

Rango: `1.0`, `1.1`, `1.2`, `1.3`

Plugins.MSSQL.Default.TrustServerCertificate

Confía en el certificado del server sin validarlo de forma predeterminada.

Rango: `true`, `false`

Plugins.MSSQL.Default.Uri

Especifique el URI predeterminado para conectarse al server MSSQL.
 Las credenciales incrustadas se ignorarán.
 El único esquema compatible es `sqlserver://`.
 El esquema puede omitirse.

Predeterminado: `sqlserver://localhost:1433`

Plugins.MSSQL.Default.User

Especifique el nombre de usuario predeterminado para un servidor MSSQL protegido.

Plugins.MSSQL.KeepAlive

Especifica el tiempo máximo de espera (en segundos) antes de que se cierren las conexiones del plugin no utilizadas.

Predeterminado: 300
 Rango: 60-900

Plugins.MSSQL.Sessions.<SessionName>.CACertPath

Especifique la ruta del archivo de sesión con nombre al certificado de clave pública de la autoridad de certificación (CA) que emitió el certificado del server MSSQL.
 <SessionName> - define el nombre de una sesión para usarlo en las claves de item.
 El certificado debe estar en formato PEM.

Plugins.MSSQL.Sessions.<SessionName>.Database

Especifica el nombre de la base de datos de la sesión con nombre.
 <SessionName> - define el nombre de una sesión para usarlo en las claves de item.

Plugins.MSSQL.Sessions.<SessionName>.Encrypt

Especifica el nivel de cifrado de la sesión con nombre.
 <SessionName> - define el nombre de una sesión para usarlo en las claves de item.

Rango:
true - los datos enviados entre el plugin y el server están cifrados;
false - los datos enviados entre el plugin y el server no están cifrados más allá del paquete de inicio de sesión;
strict - los datos enviados entre el plugin y el server están cifrados de extremo a extremo usando TDS8;
disable - los datos enviados entre el plugin y el server no están cifrados.

Plugins.MSSQL.Sessions.<SessionName>.HostNameInCertificate

Especifique el nombre común (CN) del certificado del server MSSQL para la sesión con nombre.
 <SessionName> - defina el nombre de una sesión para usarlo en las claves de item.

Plugins.MSSQL.Sessions.<SessionName>.Password

Especifique la contraseña de la sesión con nombre para un servidor MSSQL protegido.
 <SessionName> - defina el nombre de una sesión para usarlo en las claves de item.

Plugins.MSSQL.Sessions.<SessionName>.TLSMinVersion

Especifica la versión mínima de TLS de la sesión con nombre.
 <SessionName> - define el nombre de una sesión para usarlo en las claves de item.

Rango: `1.0`, `1.1`, `1.2`, `1.3`

Plugins.MSSQL.Sessions.<SessionName>.TrustServerCertificate

Confía en el certificado del servidor de la sesión especificada sin validarlo.
 <SessionName> - define el nombre de una sesión para usarlo en las claves de item.

Rango: `true`, `false`

Plugins.MSSQL.Sessions.<SessionName>.Uri

Especifique la URI de la sesión con nombre para conectarse al server MSSQL.
 <SessionName> - defina el nombre de una sesión para usarlo en las claves de item.
 Las credenciales incrustadas se ignorarán.
 El único esquema compatible es `sqlserver://`.
 El esquema puede omitirse.

Predeterminado: `sqlserver://localhost:1433`

Plugins.MSSQL.Sessions.<SessionName>.User

Especifique el nombre de usuario de la sesión con nombre para un servidor MSSQL protegido.
 <SessionName> - defina el nombre de una sesión para usarlo en las claves de item.

Plugins.MSSQL.System.Path

Especifique la ruta al ejecutable del plugin MSSQL.
 Configuración global para el plugin MSSQL. Se aplica a todas las conexiones.

Ejemplo: `Plugins.MSSQL.System.Path=/usr/libexec/zabbix/zabbix-agent2-plugin-mssql`

Plugins.MSSQL.Timeout

Establezca la duración, en segundos, que se esperará a que un server responda al conectarse por primera vez y en las operaciones posteriores de la sesión.

Predeterminado: `global timeout`
 Rango: 1-30

Opciones

Parameter	Description
-V --version	Muestra la versión del plugin y la información de la licencia.
-h --help	Muestra la información de ayuda (forma abreviada).

Parameter	Description
-t, --test <item key>	Inicia el plugin para pruebas (se ignora la configuración del plugin).

Véase también

- Descripción de los parámetros de configuración generales de Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Instrucciones para configurar plugins

Complemento NVIDIA GPU

Resumen

Los parámetros de configuración permiten personalizar el comportamiento del plugin de NVIDIA GPU.

Se recomienda especificarlos en su propio archivo de configuración (por ejemplo, `nvidiagpu.conf`), en lugar de hacerlo en el archivo de configuración de agent 2. Luego, use la directiva Include para incluir este archivo en la configuración de Zabbix agent 2.

El plugin de NVIDIA GPU es un plugin cargable y está disponible y completamente descrito en el repositorio del plugin de NVIDIA GPU.

Todos los parámetros son **opcionales**, salvo que se indique explícitamente que el parámetro es obligatorio.

Esta página está estructurada de la siguiente manera:

- Resumen de parámetros (para ver los detalles completos de un parámetro, haga clic en su nombre)
- Detalles de los parámetros

Parameter	Description
Plugins.NVIDIA.System.Path	Especifica la ruta al ejecutable del plugin de NVIDIA GPU.
Plugins.NVIDIA.Timeout	Especifica el tiempo máximo (en segundos) que se debe esperar una respuesta del server durante los intentos de conexión y las operaciones posteriores en la sesión.

Tenga en cuenta que:

- Los valores predeterminados reflejan los valores predeterminados del proceso, no los valores de los archivos de configuración suministrados;
- Los valores admiten variables de entorno;
- Zabbix solo admite archivos de configuración en codificación UTF-8 sin BOM;
- Los comentarios que comienzan con ”#” solo se admiten al principio de la línea.

Detalles del parámetro

Plugins.NVIDIA.System.Path

Especifique la ruta al ejecutable del plugin de GPU NVIDIA.

Ejemplo:

```
Plugins.NVIDIA.System.Path=/path/to/executable/nvidia
```

Plugins.NVIDIA.Timeout

Especifique el tiempo máximo (en segundos) que se esperará una respuesta del server durante los intentos de conexión y las operaciones posteriores en la sesión.
 El tiempo de espera global del tipo de item o el tiempo de espera individual del item anulará este valor si es mayor.

Predeterminado: `global timeout`
 Rango: 1–30

Véase también

- Descripción de los parámetros de configuración generales de Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Instrucciones para configurar plugins

Complemento SMART

Resumen

Los parámetros de configuración permiten personalizar el comportamiento del plugin SMART.

Se recomienda especificarlos en su propio archivo de configuración (por ejemplo, `smart.conf`), en lugar de en el archivo de configuración de agent 2. Luego, use la directiva Include para incluir este archivo en la configuración de Zabbix agent 2.

Todos los parámetros son **opcionales**, salvo que se indique explícitamente que el parámetro es obligatorio.

Esta página está estructurada de la siguiente manera:

- Resumen de parámetros (para ver los detalles completos de un parámetro, haga clic en su nombre)
- Detalles de parámetros

Parameter	Description
Plugins.Smart.Path	Especifique la ruta al ejecutable smartctl.
Plugins.Smart.Timeout	Establezca el tiempo de espera de ejecución de la solicitud (la duración, en segundos, que se espera a que una solicitud se complete antes de detenerla).

Tenga en cuenta que:

- Los valores predeterminados reflejan los valores predeterminados del proceso, no los valores de los archivos de configuración suministrados;
- Los valores admiten variables de entorno;
- La ruta al ejecutable `smartctl` debe establecerse agregando su directorio a la variable de entorno `PATH` del sistema o configurando `Plugins.Smart.Path`; esto se aplica tanto a Linux como a Windows;
- Zabbix solo admite archivos de configuración en codificación UTF-8 sin BOM;
- Los comentarios que comienzan con "#" solo se admiten al principio de la línea.

Detalles del parámetro

Plugins.Smart.Path

Especifique la ruta al ejecutable smartctl.

Predeterminado: `smartctl`

Plugins.Smart.Timeout

Establece el tiempo de espera de ejecución de la solicitud (la duración, en segundos, que se espera a que una solicitud se complete antes de interrumpirla).

Predeterminado: `global timeout<br>` Rango: 1–30

Véase también

- Descripción de los parámetros de configuración generales de Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Instrucciones para configurar plugins

Protocolo del plugin de Zabbix agent 2

El protocolo de Zabbix agent 2 se basa en código, tamaño y modelo de datos.

Código

Type	Size	Comments
Byte	4	Tipo de carga útil, actualmente solo se admite JSON.

Tamaño

Type	Size	Comments
Byte	4	Tamaño de la carga útil actual en bytes.

Datos de carga útil

Type	Size	Comments
Byte	Definido por el campo *Size*	Datos con formato JSON.

Definición de datos de carga útil

Datos comunes

Estos parámetros están presentes en todas las solicitudes/respuestas:

Name	Type	Comments
id	uint32	Para las solicitudes: el identificador incremental utilizado para vincular las solicitudes con las respuestas. Único dentro de una dirección de solicitud (es decir, de agent a plugin o de plugin a agent). Para las respuestas: ID de la solicitud correspondiente.
type	uint32	El tipo de solicitud.

Solicitud de registro

Una solicitud enviada por un plugin para escribir un mensaje de registro en el archivo de registro del agent.

direction	plugin → agent
response	no

Parámetros específicos de las solicitudes de registro:

Name	Type	Comments
severity	uint32	La severidad del mensaje (nivel de registro).
message	string	El mensaje que se va a registrar.

Ejemplo:

```
{"id":0,"type":1,"severity":3,"message":"message"}
```

Solicitud de registro

Una solicitud enviada por el agent durante la fase de inicio del agent para obtener las métricas proporcionadas y registrar un plugin.

direction	agent → plugin
response	yes

Parámetros específicos de las solicitudes de registro:

Name	Type	Comments
version	string	La versión del protocolo <major>.<minor>

Ejemplo:

```
{"id":1,"type":2,"version":"1.0"}
```

Respuesta de registro

Respuesta del plugin a la solicitud de registro.

direction	plugin → agent
response	n/a

Parámetros específicos de las respuestas de registro:

Name	Type	Comments
name	string	El nombre del plugin.
metrics	array of strings (optional)	Las métricas con descripciones tal como se usan en el plugin. Devuelve RegisterMetrics(). No está presente si se devuelve un error.
interfaces	uint32 (optional)	La máscara de bits de las interfaces compatibles del plugin. No está presente si se devuelve un error.
error	string (optional)	Un mensaje de error devuelto si un plugin no puede iniciarse. No está presente si se devuelven métricas.

Ejemplos:

```
{"id":2,"type":3,"metrics":["external.test", "External exporter Test."], "interfaces": 4}
```

o

```
{"id":2,"type":3,"error":"error message"}
```

Solicitud de inicio

Una solicitud para ejecutar la función Start de la interfaz Runner.

direction	agent → plugin
response	no

La solicitud no tiene parámetros específicos; solo contiene parámetros de datos comunes.

Ejemplo:

```
{"id":3,"type":4}
```

Solicitud de terminación

Una solicitud enviada por el agent para apagar un plugin.

direction	agent → plugin
response	no

La solicitud no tiene parámetros específicos; solo contiene parámetros de datos comunes.

Ejemplo:

```
{"id":3,"type":5}
```

Solicitud de exportación

Una solicitud para ejecutar la función Export de la interfaz Exporter.

direction	agent → plugin
response	no

Parámetros específicos de las solicitudes de exportación:

Name	Type	Comments
key	string	La clave del plugin.
parameters	array of strings (optional)	Los parámetros para la función Export.

Ejemplo:

```
{"id":4,"type":6,"key":"test.key","parameters":["foo","bar"]}
```

Respuesta de exportación

Respuesta de la función Export del interfaz Exporter.

direction	plugin → agent
response	n/a

Parámetros específicos de las respuestas de exportación:

Name	Type	Comments
value	string (optional)	Valor de respuesta de la función Export. Ausente si se devuelve un error.
error	string (optional)	Mensaje de error si la función Export no se ha ejecutado correctamente. Ausente si se devuelve value.

Ejemplos:

```
{"id":5,"type":7,"value":"response"}
```

o

```
{"id":5,"type":7,"error":"error message"}
```

Configurar solicitud

Una solicitud para ejecutar la función *Configure* de la interfaz *Configurator*.

direction	agent → plugin
response	n/a

Parámetros específicos de las solicitudes *Configure*:

Name	Type	Comments
global_options	JSON object	Objeto JSON que contiene las opciones de configuración global del agent.
private_options	JSON object (optional)	Objeto JSON que contiene las opciones de configuración privadas del plugin, si se proporcionan.

Ejemplo:

```
{"id":6,"type":8,"global_options":{...},"private_options":{...}}
```

Validar solicitud

Una solicitud para ejecutar la función *Validate* de la interfaz *Configurator*.

direction	agent → plugin
response	yes

Parámetros específicos de las solicitudes *Validate*:

Name	Type	Comments
private_options	JSON object (optional)	Objeto JSON que contiene opciones privadas de configuración del plugin, si se proporcionan.

Ejemplo:

```
{"id":7,"type":9,"private_options":{...}}
```

Validar respuesta

Respuesta de la función *Validate* de la interfaz *Configurator*.

direction	plugin → agent
response	n/a

Parámetros específicos de las respuestas *Validate*:

Name	Type	Comments
error	string (optional)	Un mensaje de error devuelto si la función Validate no se ejecuta correctamente. Ausente si se ejecuta correctamente.

Ejemplo:

```
{"id":8,"type":10}
```

o

```
{"id":8,"type":10,"error":"error message"}
```

Java gateway

Descripción general Zabbix Java gateway puede instalarse desde el código fuente o mediante paquetes.

La compatibilidad nativa para la monitorización de aplicaciones JMX existe en forma de un demonio de Zabbix denominado "Zabbix Java gateway". Zabbix Java gateway es un demonio escrito en Java. Para obtener el valor de un contador JMX concreto en un host, Zabbix server consulta Zabbix Java gateway, que utiliza la API de gestión de JMX para consultar remotamente la aplicación correspondiente. La aplicación no necesita tener instalado ningún software adicional; solo debe iniciarse con la opción `-Dcom.sun.management.jmxremote` en la línea de comandos.

Java gateway acepta conexiones entrantes de Zabbix server o proxy y solo puede utilizarse como un "proxy pasivo". A diferencia de Zabbix proxy, también puede utilizarse desde Zabbix proxy (los proxy de Zabbix no pueden encadenarse). El acceso a cada Java gateway se configura directamente en el archivo de configuración de Zabbix server o de Zabbix proxy; por lo tanto, solo se puede configurar un Java gateway por cada Zabbix server o Zabbix proxy. Si un host tiene items de tipo *JMX agent* y items de otros tipos, solo los items de tipo *JMX agent* se enviarán a Java gateway para su recuperación.

Cuando un item debe actualizarse mediante Java gateway, el server o el proxy de Zabbix se conectará a Java gateway y solicitará el valor, que Java gateway recuperará y devolverá al server o al proxy. Por lo tanto, Java gateway no almacena ningún valor en caché.

El server o el proxy de Zabbix tiene un tipo específico de procesos que se conectan a Java gateway, controlado por la opción `StartJavaPollers`. Internamente, Java gateway inicia varios hilos, controlados por la opción `START_POLLERS` opción. En el lado del server, si una conexión tarda más de "Timeout" segundos, se terminará, pero Java gateway podría seguir ocupado recuperando el valor del contador JMX. Para solucionar esto, Java gateway proporciona la opción `TIMEOUT`, que permite establecer un tiempo de espera para las operaciones de red JMX.

Véase también:

- Parámetros de configuración: Zabbix Java gateway
- Configuración desde paquetes de Debian/Ubuntu
- Configuración desde paquetes de RHEL

- Configuración desde las fuentes

El server o proxy de Zabbix intentará agrupar las solicitudes a un único destino JMX tanto como sea posible (según los intervalos de los item) y enviarlas al Java gateway en una sola conexión para mejorar el rendimiento.

Se recomienda que `StartJavaPollers` sea menor o igual que `START_POLLERS`; de lo contrario, podría haber situaciones en las que no haya hilos disponibles en el Java gateway para atender las solicitudes entrantes. En tales casos, el Java gateway usa `ThreadPoolExecutor.CallerRunsPolicy`, lo que significa que el hilo principal atenderá la solicitud entrante y, temporalmente, no aceptará ninguna solicitud nueva.

Si está intentando supervisar aplicaciones Java basadas en Wildfly con Zabbix Java gateway, instale el archivo `jboss-client.jar` más reciente disponible en la página de descargas de Wildfly.

Parámetros de configuración: Zabbix Java gateway

Si utiliza los scripts `startup.sh` y `shutdown.sh` para iniciar Zabbix Java gateway, puede especificar los parámetros de configuración necesarios en el archivo `settings.sh`. Los scripts de inicio y cierre cargan el archivo de configuración y se encargan de convertir las variables de shell (enumeradas en la primera columna) en propiedades de Java (enumeradas en la segunda columna).

Si inicia Zabbix Java gateway manualmente ejecutando `java` directamente, entonces especifica las propiedades de Java correspondientes en la línea de comandos.

Variable	Property	Mandatory	Range	Default	Description
LISTEN_IP	zabbix.listenIP	no		0.0.0.0	Dirección IP en la que escuchar.
LISTEN_PORT	zabbix.listenPort	no	1024-32767	10052	Puerto en el que escuchar.
PID_FILE	zabbix.pidFile	no		/tmp/zabbix_java.pid	Nombre del archivo PID. Si se omite, Zabbix Java Gateway se inicia como una aplicación de consola.
PROPERTIES_FILE	zabbix.propertiesFile	no			Nombre del archivo de propiedades. Puede utilizarse para establecer propiedades adicionales mediante un formato clave-valor, de modo que no sean visibles en la línea de comandos, o para sobrescribir las existentes. Por ejemplo: "javax.net.ssl.trustStorePassword=<p
START_POLLERS	zabbix.startPollers	no	1-1000	5	Número de hilos de trabajo que se iniciarán.
TIMEOUT	zabbix.timeout	no	1-30	3	Tiempo de espera para las operaciones de red (en segundos).

Warning:
El puerto 10052 no está registrado en IANA.

Configuración desde el código fuente

Resumen

Si se instala desde el código fuente, la siguiente información le ayudará a configurar el Java gateway de Zabbix.

Descripción general de los archivos

Si obtuvo Java gateway a partir del código fuente, debería haber terminado con una colección de scripts de shell, archivos JAR y archivos de configuración en `$PREFIX/sbin/zabbix_java`. A continuación se resume la función de estos archivos.

`bin/zabbix-java-gateway-$VERSION.jar`

El propio archivo JAR de Java gateway.

```
lib/logback-core-1.5.25.jar
lib/logback-classic-1.5.25.jar
lib/slf4j-api-2.0.16.jar
lib/android-json-4.3_r3.1.jar
```

Dependencias de Java gateway: las bibliotecas Logback, SLF4J y Android JSON.

```
lib/logback.xml
lib/logback-console.xml
```

Archivos de configuración de Logback.

```
shutdown.sh
startup.sh
```

Scripts auxiliares para iniciar y detener Java gateway.

`settings.sh`

Archivo de configuración que cargan los scripts de inicio y detención anteriores.

Configuración y ejecución de Java gateway

De forma predeterminada, Java gateway escucha en el puerto 10052. Si planea ejecutar Java gateway en un puerto diferente, puede especificarlo en el script `settings.sh`. Consulte la descripción del archivo de configuración de Java gateway para saber cómo especificar esta y otras opciones.

> **Warning:**
> El puerto 10052 no está registrado por IANA.

Una vez que esté conforme con la configuración, puede iniciar Java gateway ejecutando el script `startup.sh`:

```
./startup.sh
```

Del mismo modo, cuando ya no necesite Java gateway, ejecute el script `shutdown.sh` para detenerlo:

```
./shutdown.sh
```

Tenga en cuenta que, a diferencia de server o proxy, Java gateway es ligero y no necesita una base de datos.

Configuración de server para su uso con Java gateway

Con Java gateway en funcionamiento, debe indicar a Zabbix server dónde encontrar Zabbix Java gateway. Esto se hace especificando los parámetros `JavaGateway` y `JavaGatewayPort` en el archivo de configuración de server. Si el host en el que se ejecuta la aplicación JMX es monitorizado por Zabbix proxy, entonces debe especificar los parámetros de conexión en el archivo de configuración de proxy en su lugar.

```
JavaGateway=192.168.3.14
JavaGatewayPort=10052
```

De forma predeterminada, server no inicia ningún proceso relacionado con la monitorización JMX. Sin embargo, si desea usarla, debe especificar el número de instancias pre-fork de Java pollers. Esto se hace de la misma manera que se especifican los pollers y trappers normales.

```
StartJavaPollers=5
```

No olvide reiniciar server o proxy una vez que haya terminado de configurarlos.

Depuración de Java gateway

En caso de que haya algún problema con Java gateway o de que el mensaje de error que vea sobre un item en el frontend no sea lo suficientemente descriptivo, quizá desee echar un vistazo al archivo de registro de Java gateway.

De forma predeterminada, Java gateway registra sus □□□□□□ades en el archivo `/tmp/zabbix_java.log` con el nivel de registro "info". A veces esa información no es suficiente y se necesita información con el nivel de registro "debug". Para aumentar el nivel de registro, modifique el archivo `lib/logback.xml` y cambie el atributo level de la etiqueta `<root>` a `debug`:

```
<root level="debug">
  <appender-ref ref="FILE" />
</root>
```

Tenga en cuenta que, a diferencia de Zabbix server o Zabbix proxy, no es necesario reiniciar Zabbix Java gateway después de cambiar el archivo `logback.xml`; los cambios en `logback.xml` se aplicarán automáticamente. Cuando termine con la depuración, puede volver a establecer el nivel de registro en ”info”.

Si desea registrar en un archivo diferente o en un medio completamente distinto, como una base de datos, ajuste el archivo `logback.xml` para que se adapte a sus necesidades. Consulte Logback Manual para obtener más detalles.

A veces, con fines de depuración, resulta útil iniciar Java gateway como una aplicación de consola en lugar de un daemon. Para hacerlo, comente la variable `PID_FILE` en `settings.sh`. Si se omite `PID_FILE`, el script `startup.sh` inicia Java gateway como una aplicación de consola y hace que Logback use el archivo `lib/logback-console.xml` en su lugar, lo que no solo registra en la consola, sino que también tiene habilitado el nivel de registro ”debug”.

Por último, tenga en cuenta que, dado que Java gateway usa SLF4J para el registro, puede reemplazar Logback por el framework de su elección colocando un archivo JAR adecuado en el directorio lib. Consulte SLF4J Manual para obtener más detalles.

Monitoreo JMX

Consulte la página JMX monitoring para obtener más detalles.

Configuración desde paquetes de Debian/Ubuntu

Descripción general

Si se ha instalado desde paquetes, la siguiente información le ayudará a configurar el Java gateway de Zabbix.

Configuración y ejecución de Java gateway

La configuración de Java gateway se puede ajustar en el archivo:

```
/etc/zabbix/zabbix_java_gateway.conf
```

Para obtener más detalles, consulte los parámetros de configuración de Zabbix Java gateway.

Para iniciar Zabbix Java gateway:

```
systemctl restart zabbix-java-gateway
```

Para iniciar automáticamente Zabbix Java gateway al arrancar:

```
systemctl enable zabbix-java-gateway
```

Configuración de server para su uso con Java gateway

Con Java gateway en ejecución, debe indicar a Zabbix server dónde encontrar Zabbix Java gateway. Esto se hace especificando los parámetros `JavaGateway` y `JavaGatewayPort` en el archivo de configuración de server. Si el host en el que se ejecuta la aplicación JMX es supervisado por Zabbix proxy, entonces debe especificar los parámetros de conexión en el archivo de configuración de proxy en su lugar.

```
JavaGateway=192.168.3.14
JavaGatewayPort=10052
```

De forma predeterminada, server no inicia ningún proceso relacionado con la supervisión de JMX. Sin embargo, si desea usarla, debe especificar el número de instancias pre-fork de Java pollers. Esto se hace de la misma manera que se especifican los pollers y trappers normales.

```
StartJavaPollers=5
```

No olvide reiniciar server o proxy una vez que haya terminado de configurarlos.

Depuración de Java gateway

El archivo de registro de Java gateway de Zabbix es:

```
/var/log/zabbix/zabbix_java_gateway.log
```

Si desea aumentar el nivel de registro, edite el archivo:

```
/etc/zabbix/zabbix_java_gateway_logback.xml
```

y cambie `info` en `level="info"` a debug o incluso `trace` (para una depuración profunda):

```
<configuration scan="true" scanPeriod="15 seconds">
[...]
      <root level="info">
              <appender-ref ref="FILE" />
      </root>

</configuration>
```

Monitoreo JMX

Consulte la página JMX monitoring para obtener más detalles.

Configuración desde paquetes de RHEL

Descripción general

Si se instala desde paquetes, la siguiente información le ayudará a configurar el Java gateway de Zabbix.

Configuración y ejecución de Java gateway

Los parámetros de configuración de Zabbix Java gateway se pueden ajustar en el archivo:

```
/etc/zabbix/zabbix_java_gateway.conf
```

Para obtener más detalles, consulte los parámetros de configuración de Zabbix Java gateway.

Para iniciar Zabbix Java gateway:

```
systemctl restart zabbix-java-gateway
```

Para iniciar automáticamente Zabbix Java gateway al arrancar:

RHEL 7 y posteriores:

```
systemctl enable zabbix-java-gateway
```

RHEL anteriores a 7:

```
chkconfig --level 12345 zabbix-java-gateway on
```

Configuración de server para su uso con Java gateway

Con Java gateway en funcionamiento, debe indicar a Zabbix server dónde encontrar Zabbix Java gateway. Esto se hace especificando los parámetros `JavaGateway` y `JavaGatewayPort` en el archivo de configuración de server. Si el host en el que se ejecuta la aplicación JMX es supervisado por Zabbix proxy, entonces debe especificar los parámetros de conexión en el archivo de configuración de proxy en su lugar.

```
JavaGateway=192.168.3.14
JavaGatewayPort=10052
```

De forma predeterminada, server no inicia ningún proceso relacionado con la supervisión de JMX. Sin embargo, si desea usarla, debe especificar el número de instancias pre-fork de Java pollers. Esto se hace de la misma manera que se especifica el número de pollers y trappers normales.

```
StartJavaPollers=5
```

No olvide reiniciar server o proxy una vez que haya terminado de configurarlos.

Depuración de Java gateway

El archivo de registro de Java gateway de Zabbix es:

```
/var/log/zabbix/zabbix_java_gateway.log
```

Si desea aumentar el nivel de registro, edite el archivo:

```
/etc/zabbix/zabbix_java_gateway_logback.xml
```

y cambie `info` en `level="info"` a debug o incluso `trace` (para una solución de problemas en profundidad):

```
<configuration scan="true" scanPeriod="15 seconds">
[...]
      <root level="info">
              <appender-ref ref="FILE" />
      </root>
```

```
</configuration>
```

Monitoreo JMX

Consulte la página Monitoreo JMX para obtener más detalles.

Remitente

Descripción general Zabbix sender es una utilidad de línea de comandos que puede utilizarse para enviar datos de rendimiento al server o proxy de Zabbix para su procesamiento.

La utilidad suele utilizarse en scripts de usuario de ejecución prolongada para enviar periódicamente datos de disponibilidad y rendimiento.

Para enviar resultados directamente al server o proxy de Zabbix, debe configurarse un tipo de item *Zabbix trapper*.

Véase también:

- Protocolo de Zabbix sender
- Biblioteca de enlace dinámico de Zabbix sender para Windows
- Biblioteca de Python para Zabbix, que incluye funcionalidad integrada para actuar como Zabbix sender

Ejecución de Zabbix sender Un ejemplo de ejecución de Zabbix sender en UNIX:

```
cd bin
./zabbix_sender -z zabbix -s "Linux DB3" -k db.connections -o 43
```

donde:

- `z` - host del Zabbix server (también se puede utilizar una dirección IP);
- `s` - nombre técnico del host supervisado (tal como está registrado en el frontend de Zabbix);
- `k` - clave del item;
- `o` - valor que se enviará.

Attention:
Las opciones que contienen espacios en blanco deben escribirse entre comillas dobles.

Zabbix sender se puede utilizar para enviar varios valores desde un archivo de entrada. Consulte la página del manual de Zabbix sender para obtener más información.

Si se especifica un archivo de configuración, Zabbix sender utiliza todas las direcciones definidas en el parámetro de configuración `ServerActive` del agent para enviar los datos. Si falla el envío a una dirección, el sender intenta enviarlos a las demás direcciones. Si falla el envío de datos por lotes a una dirección, los lotes siguientes no se envían a esa dirección.

Zabbix sender acepta cadenas codificadas en UTF-8 (tanto en sistemas similares a UNIX como en Windows) sin una marca de orden de bytes (BOM) al principio del archivo.

Zabbix sender en Windows se puede ejecutar de forma similar:

```
zabbix_sender.exe [options]
```

Los escenarios de envío en tiempo real de `zabbix_sender` recopilan varios valores que se le pasan en rápida sucesión y los envían al server en una sola conexión. Un valor cuya diferencia con respecto al valor anterior no sea superior a 0,2 segundos puede incluirse en la misma pila, pero el tiempo máximo de sondeo sigue siendo de 1 segundo.

Note:
Zabbix sender finalizará si el archivo de configuración especificado contiene una entrada de parámetro no válida (que no siga la notación `parameter=value`).

Ejecución de Zabbix sender con descubrimiento de bajo nivel Un ejemplo de ejecución de Zabbix sender para enviar un valor en formato JSON para el descubrimiento de bajo nivel:

```
./zabbix_sender -z 192.168.1.113 -s "Zabbix server" -k trapper.discovery.item -o '[{"{#FSNAME}":"/","{#FST
```

Para que esto funcione, la regla de descubrimiento de bajo nivel debe tener el tipo de item *Zabbix trapper* (en este ejemplo, con la clave `trapper.discovery.item`).

Protocolo de Zabbix sender

Zabbix server y Zabbix proxy usan un protocolo de comunicación basado en JSON para recibir datos de Zabbix sender. Los datos pueden recibirse con la ayuda de un trapper item, o un HTTP agent item con trapping habilitado.

Los mensajes de solicitud y respuesta deben comenzar con header and data length.

Solicitud de Zabbix sender

```
{
    "request": "sender data",
    "data": [
        {
            "host": "<hostname>",
            "key": "trap",
            "value": "test value"
        }
    ]
}
```

Respuesta del server de Zabbix

```
{
    "response": "success",
    "info": "processed: 1; failed: 0; total: 1; seconds spent: 0.060753"
}
```

Solicitud de Zabbix sender con una marca de tiempo

Alternativamente, Zabbix sender puede enviar una solicitud con una marca de tiempo y nanosegundos.

```
{
    "request": "sender data",
    "data": [
        {
            "host": "<hostname>",
            "key": "trap",
            "value": "test value",
            "clock": 1516710794,
            "ns": 592397170
        },
        {
            "host": "<hostname>",
            "key": "trap",
            "value": "test value",
            "clock": 1516710795,
            "ns": 192399456
        }
    ],
    "clock": 1516712029,
    "ns": 873386094
}
```

Respuesta del server de Zabbix

```
{
    "response": "success",
    "info": "processed: 2; failed: 0; total: 2; seconds spent: 0.060904"
}
```

Biblioteca de enlace dinámico de Zabbix sender para Windows

En un entorno Windows, las aplicaciones pueden enviar datos al server/proxy de Zabbix mediante la biblioteca de enlace dinámico de Zabbix sender (`zabbix_sender.dll`) en lugar de tener que iniciar un proceso externo (`zabbix_sender.exe`).

Se requieren `zabbix_sender.h` y `zabbix_sender.lib` para compilar aplicaciones de usuario con `zabbix_sender.dll`.

Obtención

Hay dos formas de obtener `zabbix_sender.dll`.

1. Descargue los archivos `zabbix_sender.h`, `zabbix_sender.lib` y `zabbix_sender.dll` como un archivo ZIP.

Al elegir las opciones de descarga, asegúrese de seleccionar "No encryption" en *Encryption* y "Archive" en *Packaging*. Después, descargue Zabbix agent (no Zabbix agent 2).

Los archivos `zabbix_sender.h`, `zabbix_sender.lib` y `zabbix_sender.dll` estarán dentro del archivo ZIP descargado en el directorio `bin\dev`. Descomprima los archivos donde los necesite.

2. Compile `zabbix_sender.dll` desde el código fuente (consulte las instrucciones).

La biblioteca de enlace dinámico con los archivos de desarrollo se ubicará en el directorio `bin\winXX\dev`. Para usarla, incluya el archivo de encabezado `zabbix_sender.h` y vincule la biblioteca `zabbix_sender.lib`.

Véase también

- ejemplo de una utilidad sencilla de Zabbix sender implementada con la biblioteca de enlace dinámico de Zabbix sender para ilustrar el uso de la biblioteca;
- archivo zabbix_sender.h con las funciones de interfaz de la biblioteca de enlace dinámico de Zabbix sender. Este archivo contiene documentación que explica el propósito de cada función de interfaz, sus argumentos y su valor de retorno.

Obtener

Descripción general Zabbix get es una utilidad de línea de comandos que se puede usar para comunicarse con el agent de Zabbix y recuperar la información requerida del agent.

La utilidad se usa normalmente para la resolución de problemas de los agents de Zabbix.

Consulte también la biblioteca de Python para Zabbix, que tiene funcionalidad integrada para actuar como Zabbix get.

Ejecutar Zabbix get Un ejemplo de ejecución de Zabbix get en UNIX para obtener el valor de carga del procesador desde el agent:

```
cd bin
./zabbix_get -s 127.0.0.1 -p 10050 -k system.cpu.load[all,avg1]
```

Otro ejemplo de ejecución de Zabbix get para capturar una cadena de un sitio web:

```
cd bin
./zabbix_get -s 192.168.1.1 -p 10050 -k "web.page.regexp[www.example.com,,,\"USA: ([a-zA-Z0-9.-]+)\",,\1]"
```

Tenga en cuenta que la clave del item aquí contiene un espacio, por lo que se usan comillas para marcar la clave del item para la shell. Las comillas no forman parte de la clave del item; la shell las recortará y no se pasarán al agent de Zabbix.

Si una clave de item no es compatible, Zabbix get devolverá el código de salida 1.

Zabbix get acepta los siguientes parámetros de línea de comandos:

```
-s --host <host name or IP>             Especifica el nombre de host o la dirección IP de un host
-p --port <port number>                 Especifica el número de puerto del agent que se ejecuta en el host
-I --source-address <IP address>        Especifica la dirección IP de origen
-t --timeout <seconds>                  Especifica el tiempo de espera. Rango válido: 1-600 segundos (pred
-k --key <item key>                     Especifica la clave del item del que se recuperará el valor
-P --protocol <value>                   Protocolo usado para comunicarse con el agent. Valores:
                                            auto - se conecta usando el protocolo JSON, con retroceso y re
                                            json - se conecta usando el protocolo JSON
                                            plaintext - se conecta usando el protocolo de texto plano, don
-h --help                               Muestra este mensaje de ayuda
-V --version                            Muestra el número de versión

--tls-connect <value>                   Cómo conectarse al agent. Valores:
                                            unencrypted - se conecta sin cifrado (predeterminado)
                                            psk - se conecta usando TLS y una clave previamente compartida
                                            cert - se conecta usando TLS y un certificado
--tls-ca-file <CA file>                 Ruta completa de un archivo que contiene los certificados de la(s)
```

```
--tls-crl-file <CRL file>                Ruta completa de un archivo que contiene certificados revocados
--tls-agent-cert-issuer <cert issuer>    Emisor del certificado del agent permitido
--tls-agent-cert-subject <cert subject> Sujeto del certificado del agent permitido
--tls-cert-file <cert file>              Ruta completa de un archivo que contiene el certificado o la caden
--tls-key-file <key file>                Ruta completa de un archivo que contiene la clave privada
--tls-psk-identity <PSK-identity>        Cadena única, sensible a mayúsculas y minúsculas, usada para ident
--tls-psk-file <PSK-file>                Ruta completa de un archivo que contiene la clave previamente comp
--tls-cipher13 <cipher-string>           Cadena de cifrado para OpenSSL 1.1.1 o posterior para TLS 1.3. Anu
--tls-cipher <cipher-string>             Cadena de prioridad de GnuTLS (para TLS 1.2 y superior) o cadena d
```

Consulte también la página de manual de Zabbix get para obtener más información.

Zabbix get en Windows se puede ejecutar de forma similar:

```
zabbix_get.exe [options]
```

JS

Resumen zabbix_js es una utilidad de línea de comandos que se puede usar para probar scripts incrustados.

Esta utilidad ejecutará un script de usuario con un parámetro de cadena e imprimirá el resultado. Los scripts se ejecutan mediante el motor de scripting integrado de Zabbix.

En caso de errores de compilación o ejecución, zabbix_js imprimirá el error en stderr y saldrá con el código 1.

```
zabbix_js -s script-file -p input-param [-l log-level] [-t timeout]
zabbix_js -s script-file -i input-file [-l log-level] [-t timeout]
zabbix_js -h
zabbix_js -V
```

Uso zabbix_js acepta los siguientes parámetros de línea de comandos:

```
-s, --script script-file          Especifica el nombre del archivo del script que se ejecutará
                                  Si se especifica "-" como nombre de archivo, el script se leerá desde st
-i, --input input-file            Especifica el nombre del archivo con la información de entrada
                                  El contenido se pasará al script como un valor
                                  Mutuamente excluyente con -p
                                  Si se especifica "-" como nombre de archivo, el script se leerá desde st
-p, --param input-param           Especifica el parámetro de entrada: la variable que se pasará al script
                                  Mutuamente excluyente con -i
-l, --loglevel log-level          Especifica el nivel de registro
                                  Rango: 0-5
-t, --timeout timeout             Especifica el tiempo de espera en segundos
                                  Rango válido: 1-600 segundos (valor predeterminado: 10 segundos)
-h, --help                        Muestra información de ayuda
-V, --version                     Muestra el número de versión
-w <webdriver url>                Habilita la supervisión del navegador
```

Ejemplos: Ejemplo 1: Ejecutar un script con una variable que contiene un valor

El script (script-file.js):

```
return value;
```

Uso:

```
zabbix_js -s script-file.js -p example_value
```

Salida: example_value

Ejemplo 2: Ejecutar un script con un archivo que contiene información de entrada

El archivo con el script (script-file.js):

```
return value;
```

El archivo con la información de entrada (`example.txt`):

```
Ejemplo de información de entrada desde el archivo
```

Uso:

```
zabbix_js -s script-file.js -i example.txt
```

Salida: `Ejemplo de información de entrada desde el archivo`

Ejemplo 3: Ejecutar un script con entrada leída desde stdin

El archivo con el script (`script-file.js`):

```
return value;
```

Uso:

```
zabbix_js -s script-file.js -i -
```

Entrada (stdin):

```
Ejemplo de entrada desde stdin
```

Salida: `Ejemplo de entrada desde stdin`

Servicio web

Descripción general Zabbix web service es un proceso que se utiliza para la comunicación con servicios web externos. Actualmente, Zabbix web service se utiliza para generar y enviar informes programados, con planes de añadir funcionalidad adicional en el futuro.

El server de Zabbix se conecta al web service mediante HTTP(S). Zabbix web service requiere que Google Chrome esté instalado en el mismo host; en algunas distribuciones, el servicio también puede funcionar con Chromium (consulte los problemas conocidos).

Instalación El paquete oficial zabbix-web-service está disponible en el repositorio de Zabbix.

Para compilar el servicio web de Zabbix a partir de las fuentes, especifique la opción de configuración `--enable-webservice`.

Para configurar el servicio web de Zabbix, actualice los parámetros del archivo de configuración del servicio web.

> **Attention:**
> Se recomienda encarecidamente configurar el cifrado entre el server de Zabbix y el servicio web de Zabbix mediante certificados. De forma predeterminada, los datos transmitidos entre el server de Zabbix y el servicio web de Zabbix no están cifrados, lo que puede permitir el acceso no autorizado.

Parámetros de configuración: Zabbix web service

Overview

El servicio web de Zabbix es un proceso que se utiliza para la comunicación con servicios web externos.

Los parámetros admitidos por el archivo de configuración del servicio web de Zabbix (`zabbix_web_service.conf`) se enumeran en esta sección.

Los parámetros se enumeran sin información adicional. Haga clic en el parámetro para ver los detalles completos.

Parameter	Description
AllowedIP	Una lista de direcciones IP separadas por comas, opcionalmente en notación CIDR, o nombres DNS de servers de Zabbix y proxies de Zabbix.
DebugLevel	El nivel de depuración.
IgnoreURLCertErrors	Especifica el manejo de errores de validación del certificado TLS al acceder a la URL del frontend.
Include	Puede incluir archivos individuales o todos los archivos de un directorio en el archivo de configuración.

Parameter	Description
ListenPort	El servicio escuchará en este puerto las conexiones del server.
LogFile	El nombre del archivo de registro.
LogFileSize	El tamaño máximo del archivo de registro.
LogType	El tipo de salida del registro.
Timeout	El tiempo máximo (en segundos) dedicado al formateo del report PDF de un dashboard.
TLSAccept	Qué conexiones entrantes aceptar.
TLSCAFile	La ruta completa de un archivo que contiene los certificados de la(s) CA de nivel superior para la verificación del certificado del par, utilizado para las comunicaciones cifradas entre componentes de Zabbix.
TLSCertFile	La ruta completa de un archivo que contiene el certificado del servicio o la cadena de certificados, utilizado para las comunicaciones cifradas entre componentes de Zabbix.
TLSKeyFile	La ruta completa de un archivo que contiene la clave privada del servicio.

Todos los parámetros son opcionales, salvo que se indique explícitamente que el parámetro es obligatorio.

Tenga en cuenta que:

- Los valores predeterminados reflejan los valores predeterminados del proceso, no los valores de los archivos de configuración suministrados;
- Los valores admiten variables de entorno;
- Zabbix solo admite archivos de configuración en codificación UTF-8 sin BOM;
- Los comentarios que comienzan con # solo se admiten al principio de la línea.

Detalles del parámetro

AllowedIP

Una lista de direcciones IP separadas por comas, opcionalmente en notación CIDR, o nombres DNS de los servers de Zabbix y proxies de Zabbix. Las conexiones entrantes solo se aceptarán desde los hosts enumerados aquí.
Si la compatibilidad con IPv6 está habilitada, entonces `127.0.0.1`, `::127.0.0.1`, `::ffff:127.0.0.1` se tratan por igual y `::/0` permitirá cualquier dirección IPv4 o IPv6. Se puede usar `0.0.0.0/0` para permitir cualquier dirección IPv4.

Ejemplo:

```
127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.example.com
```

Obligatorio: sí

DebugLevel

Especifique el nivel de depuración: - *0* - información básica sobre el inicio y la detención de los procesos de Zabbix; - *1* - información crítica; - *2* - información de error; - *3* - advertencias; - *4* - para depuración (genera mucha información); - *5* - depuración extendida (genera aún más información).

Predeterminado: 3
 Rango: 0-5

IgnoreURLCertErrors

Especifica el manejo de errores de validación del certificado TLS al acceder a la URL del frontend:
0 - no ignorar los errores del certificado;
1 - ignorar los errores del certificado.

Predeterminado: 0
 Rango: 0-1

Include

Especifique un directorio o archivos individuales que se incluirán en el archivo de configuración.
 Durante la instalación de Zabbix, este parámetro se establece en `/usr/local/etc`, a menos que lo cambie durante el tiempo de compilación.

Reglas generales para el parámetro `Include`:

- La ruta puede ser relativa a la ubicación del archivo `zabbix_web_service.conf`.
- Puede especificar varios parámetros `Include` en un solo archivo de configuración.
- Zabbix procesa el parámetro `Include` en cuanto lo lee, así que considere colocarlo al final del archivo de configuración.

Reglas para los archivos incluidos:

- Cada archivo debe ser legible.
- Cada archivo debe tener una sintaxis correcta. De lo contrario, el componente no se iniciará.

Reglas para incluir directorios o patrones de archivos:

- No asuma ningún orden de inclusión (por ejemplo, los archivos no se incluyen en orden alfabético).

- No defina el mismo parámetro en más de un archivo incluido (por ejemplo, para sobrescribir una configuración general con una específica), ya que no puede predecir qué valor se utilizará.
- Tenga cuidado con las copias de seguridad que algunos editores de texto crean automáticamente (por ejemplo, al editar `include/my_specific.conf` puede crearse `include/my_specific.conf.BAK`), ya que ambos archivos se incluirán.

Para incluir solo los archivos relevantes de un directorio, use el comodín (`*`) para la coincidencia de patrones.

Ejemplo:

```
Include=/absolute/path/to/config/files/*.conf
```

ListenPort

El servicio escuchará en este puerto las conexiones procedentes del server.

Predeterminado: 10053
 Rango: 1024-32767

LogFile

El nombre del archivo de registro.

Ejemplo:

`/tmp/zabbix_web_service.log`

Obligatorio: Sí, si `LogType` está configurado en `file`; de lo contrario, no

LogFileSize

El tamaño máximo de un archivo de registro en MB.
0 - deshabilita la rotación automática de registros.
Nota: Si se alcanza el límite de tamaño del archivo de registro y la rotación de archivos falla, por cualquier motivo, el archivo de registro existente se trunca y se inicia uno nuevo.

Predeterminado: 1
 Rango: 0-1024

LogType

El tipo de salida del registro: - *file* - escribe el registro en el archivo especificado por el parámetro LogFile; - *system* - escribe el registro en syslog; - *console* - escribe el registro en la salida estándar.

Predeterminado: `file`

Tiempo de espera

El tiempo máximo (en segundos) dedicado a dar formato al PDF report de un dashboard.

Predeterminado: 10
 Rango: 1-30

TLSAccept

Qué conexiones entrantes aceptar: - *unencrypted* - aceptar conexiones sin cifrado (predeterminado); - *cert* - aceptar conexiones con TLS y un certificado.

Predeterminado: `unencrypted`

TLSCAFile

La ruta completa del archivo que contiene los certificados de la(s) CA de nivel superior para la verificación del certificado del par, utilizada para las comunicaciones cifradas entre los componentes de Zabbix.

TLSCertFile

La ruta completa del archivo que contiene el certificado del servicio o la cadena de certificados, utilizada para comunicaciones cifradas con los componentes de Zabbix.

TLSKeyFile

La ruta completa del archivo que contiene la clave privada del servicio, utilizada para las comunicaciones cifradas entre los componentes de Zabbix.

Adaptador bridge

El adaptador bridge de Zabbix permite que su server de Zabbix se comunique con la aplicación móvil de Zabbix.

La aplicación móvil de Zabbix no se conecta directamente a su server de Zabbix. En su lugar, dos componentes intercambian datos entre sí:

- Zabbix Bridge: un servicio alojado de forma centralizada que intercambia datos con la aplicación móvil de Zabbix en su dispositivo móvil.
- Adaptador bridge de Zabbix: un componente que instala y ejecuta. Conecta su server de Zabbix con Zabbix Bridge.

El adaptador bridge puede instalarse en el mismo sistema que su server de Zabbix o en un sistema diferente mediante paquetes.

Para configurar el adaptador bridge de Zabbix, actualice su archivo de configuración (ruta predeterminada: `/etc/zabbix/zabbix_bridge_`

Para mantener seguro el adaptador bridge de Zabbix, consulte Acceso móvil seguro.

Flujo de datos Cuando utiliza la aplicación móvil de Zabbix, los datos fluyen de la siguiente manera:

1. Realiza una acción en la aplicación móvil de Zabbix, que envía una consulta a Zabbix Bridge.
2. Zabbix Bridge pasa la consulta al adaptador de Zabbix Bridge como una solicitud JSON-RPC.
3. El adaptador de Zabbix Bridge comprueba si se permite el método de la API de Zabbix solicitado. Si no se permite el método, el adaptador de Bridge rechaza la solicitud. Si se permite el método, el adaptador de Bridge envía la solicitud a la API de Zabbix.
4. La API de Zabbix procesa la solicitud y devuelve una respuesta al adaptador de Zabbix Bridge.
5. El adaptador de Zabbix Bridge devuelve la respuesta a Zabbix Bridge.
6. Zabbix Bridge pasa la respuesta a la aplicación móvil de Zabbix, que muestra los datos en su dispositivo móvil.

Parámetros de configuración: adaptador de puente de Zabbix

Descripción general

Los parámetros de configuración permiten personalizar el comportamiento del adaptador de bridge de Zabbix.

Especifique los valores de los parámetros en el archivo de configuración (`zabbix_bridge_adapter.conf`). Todos los parámetros son **opcionales**, a menos que se indique explícitamente que el parámetro es obligatorio.

Esta página está estructurada de la siguiente manera:

- Descripción general de los parámetros (para consultar todos los detalles de un parámetro, haga clic en su nombre)
- Detalles de los parámetros

Parámetro	Descripción
AllowedAPIMethods	Especifique qué métodos de la API de Zabbix se pueden llamar a través del adaptador de bridge de Zabbix.
DataDir	Especifique el directorio donde se crean y almacenan el archivo de estado y las claves de cifrado del adaptador de bridge.
DebugLevel	Establezca el nivel de depuración.
Include	Especifique un directorio o archivos individuales que se incluirán en el archivo de configuración.
ListenIP	Especifique la dirección IP de escucha del adaptador de bridge para las conexiones HTTP(S) entrantes desde el server de Zabbix.
ListenHealthPort	Especifique el puerto de escucha del adaptador de bridge para las conexiones HTTP entrantes de comprobación de estado.
ListenPort	Especifique el puerto de escucha del adaptador de bridge para las conexiones HTTP(S) entrantes desde el server de Zabbix.
LogFile	Especifique la ruta completa al archivo de registro del adaptador de bridge.
LogFileSize	Establezca el tamaño máximo del archivo de registro (en MB).
LogType	Especifique el tipo de salida del registro.
PidFile	Especifique el archivo PID.
TLSConfig.AllowedCN	Especifique una lista separada por comas de nombres comunes permitidos de los clientes (solo `mTLS`).
TLSConfig.CAFile	Especifique la ruta completa a un archivo que contenga los certificados de CA de nivel superior utilizados para verificar los certificados de los clientes en el modo `mTLS`.
TLSConfig.CertFile	Especifique la ruta completa a un archivo que contenga el certificado o la cadena de certificados.
TLSConfig.KeyFile	Especifique la ruta completa a un archivo que contenga la clave privada.
TLSConfig.Mode	Establezca el modo TLS para las conexiones entrantes en `ListenPort`.
ServerId	Especifique el identificador único del server de Zabbix. **Obligatorio**.
ZabbixAPITLSCAFile	Especifique la ruta completa a un archivo que contenga los certificados de CA de nivel superior utilizados por el adaptador de bridge para verificar el certificado de la API de Zabbix.
ZabbixAPITLSConnect	Establezca el modo TLS para las conexiones salientes a la API de Zabbix.

Parámetro	Descripción
ZabbixAPIURL	Especifique la URL del endpoint de la API de Zabbix. **Obligatorio**.
ZabbixBridgeURL	Especifique la URL de Zabbix Bridge para recuperar la configuración de Zabbix Bridge (se recupera cuando se inicia el adaptador de bridge).

Tenga en cuenta lo siguiente:

- Los valores predeterminados reflejan los valores predeterminados del proceso, no los valores de los archivos de configuración incluidos.
- Los valores admiten variables de entorno.
- Zabbix admite archivos de configuración únicamente con codificación UTF-8 y sin BOM;
- Los comentarios que comienzan con # solo se admiten al principio de la línea.

Detalles de los parámetros

AllowedAPIMethods

Especifique qué métodos de la API de Zabbix se pueden llamar a través del adaptador de Zabbix bridge.
 Para varios métodos, repita el parámetro, un método por línea.

Cada valor del parámetro debe ser un nombre de método exacto (por ejemplo, `problem.get`) o un patrón comodín que sustituya únicamente el nombre completo de la API (por ejemplo, `*.get`) o el nombre completo del método (por ejemplo, `problem.*`).

Cualquier otro valor no es válido e impide que se inicie el adaptador de Zabbix bridge. Esto incluye un comodín independiente (`*`), un patrón comodín parcial (`host*` o `host*.*`) o un valor vacío (`AllowedAPIMethods=`).

Ejemplo:

```
AllowedAPIMethods=apiinfo.version
AllowedAPIMethods=problem.*
AllowedAPIMethods=*.get
```

El conjunto mínimo de métodos que requiere la aplicación móvil de Zabbix:

```
AllowedAPIMethods=apiinfo.version
AllowedAPIMethods=device.onboard
AllowedAPIMethods=device.offboard
AllowedAPIMethods=user.get
AllowedAPIMethods=role.get
AllowedAPIMethods=problem.get
AllowedAPIMethods=event.get
AllowedAPIMethods=event.acknowledge
AllowedAPIMethods=trigger.get
AllowedAPIMethods=alert.get
AllowedAPIMethods=host.get
AllowedAPIMethods=hostgroup.get
```

Predeterminado: no establecido (todos los métodos de la API de Zabbix están denegados)

DataDir

Especifique el directorio donde se crean y almacenan el archivo de estado del adaptador bridge y las claves de cifrado.
 El directorio debe tener permisos 0700, y todos los archivos almacenados en él deben tener permisos 0600.

Valor predeterminado: `/var/lib/zabbix-bridge-adapter`

DebugLevel

Establece el nivel de depuración:

- 0 - información básica sobre el inicio y la detención de los procesos de Zabbix
- 1 - información crítica
- 2 - información sobre errores
- 3 - advertencias
- 4 - para la depuración (genera mucha información)
- 5 - depuración ampliada (genera aún más información)

Predeterminado: 3
 Rango: 0-5

Incluir

Especifique un directorio o archivos individuales que se incluirán en el archivo de configuración.
 Durante la instalación de Zabbix, este parámetro se establece en `/usr/local/etc`, a menos que lo cambie durante la compilación.

Reglas generales para el parámetro `Include`:

- La ruta puede ser relativa a la ubicación del archivo `zabbix_bridge_adapter.conf`.
- Puede especificar varios parámetros `Include` en un mismo archivo de configuración.
- Zabbix procesa el parámetro `Include` en cuanto lo lee, por lo que se recomienda colocarlo al final del archivo de configuración.

Reglas para los archivos incluidos:

- Cada archivo debe ser legible.
- Cada archivo debe tener una sintaxis correcta. De lo contrario, el componente no se iniciará.

Reglas para incluir directorios o patrones de archivos:

- No dé por supuesto ningún orden de inclusión (por ejemplo, los archivos no se incluyen en orden alfabético).
- No defina el mismo parámetro en más de un archivo incluido (por ejemplo, para sobrescribir una configuración general con una específica), ya que no puede predecir qué valor se utilizará.
- Preste atención a las copias de seguridad que algunos editores de texto crean automáticamente (por ejemplo, al editar `include/my_specific.conf` se puede crear `include/my_specific.conf.BAK`), ya que ambos archivos se incluirán.

Para incluir únicamente los archivos relevantes de un directorio, utilice el comodín (`*`) para la coincidencia de patrones.

Ejemplo:

```
Include=/absolute/path/to/config/files/*.conf
```

ListenIP

Especifique la dirección IP de escucha del adaptador de puente para las conexiones HTTP(S) entrantes desde Zabbix server.
 Si no se especifica, el adaptador de puente escucha en todas las direcciones IP.

ListenHealthPort

Especifique el puerto de escucha del adaptador de puente para las conexiones entrantes de comprobación de estado HTTP.
 Proporciona los endpoints `/health` y `/ready`.
 Si no se especifica, el servicio de estado no se inicia.

Valor predeterminado: 8081
 Rango: 1024-32767

ListenPort

Especifique el puerto de escucha del adaptador puente para las conexiones HTTP(S) entrantes desde el Zabbix server.

Predeterminado: 10054
 Rango: 1024-32767

LogFile

Especifique la ruta completa al archivo de registro del adaptador bridge.

Obligatorio: **Sí** si `LogType` está establecido en `file`

Predeterminado: `/tmp/zabbix_bridge_adapter.log`

LogFileSize

Establece el tamaño máximo del archivo de registro (en MB).
 0 - deshabilita la rotación automática de registros.

Valor predeterminado: 1
 Rango: 0-1024

LogType

Especifique el tipo de salida del registro:

- `file`: escribe el registro en el archivo especificado por el parámetro `LogFile`.
- `system`: escribe el registro en syslog.
- `console`: escribe el registro en la salida estándar.

Valor predeterminado: `file`

PidFile

Especifique el archivo PID.

Predeterminado: `/tmp/zabbix_bridge_adapter.pid`

ServerId

Especifique el identificador único del server de Zabbix.
 Disponible en *Reports* > *System information* en la interfaz de usuario de Zabbix.

Obligatorio: **Sí**

TLSConfig.AllowedCN

Especifique una lista separada por comas de los Common Names de cliente permitidos (solo `mTLS`).
 Si no se especifica, el adaptador de bridge no comprueba el Common Name del cliente.

TLSConfig.CAFile

Especifique la ruta completa a un archivo que contenga el/los certificado(s) de CA de nivel superior utilizados para verificar los certificados de cliente en el modo `mTLS`.

TLSConfig.CertFile

Especifique la ruta completa a un archivo que contenga el certificado o la cadena de certificados.

TLSConfig.KeyFile

Especifique la ruta completa a un archivo que contenga la clave privada.

TLSConfig.Mode

Establece el modo TLS para las conexiones entrantes en `ListenPort`:

- `none`: acepta conexiones sin cifrado.
- `tls`: TLS estándar con v1.2 como versión mínima.
- `mtls`: amplía TLS con autenticación mutua.

Valor predeterminado: `none`

ZabbixAPITLSCAFile

Especifique la ruta completa de un archivo que contenga el/los certificado(s) de la CA de nivel superior utilizados por el adaptador de puente para verificar el certificado de la API de Zabbix.

Obligatorio: **Sí** si `ZabbixAPITLSConnect` está establecido en `verify_ca` o `verify_full`.

ZabbixAPITLSConnect

Establece el modo TLS para las conexiones salientes a la API de Zabbix:

- `disabled` (o vacío): usar HTTP sin cifrado; `ZabbixAPIURL` debe usar `http://`.
- `required`: usar HTTPS sin verificar el certificado (solo para pruebas).
- `verify_ca`: usar HTTPS y verificar el certificado de la API de Zabbix.
- `verify_full`: usar HTTPS y verificar el certificado y el nombre de host de la API de Zabbix.

ZabbixAPIURL

Especifique la URL del endpoint de la API de Zabbix.

Obligatorio: **Sí**

Ejemplo:

```
ZabbixAPIURL=https://example.com/zabbix/api_jsonrpc.php
```

ZabbixBridgeURL

Especifique la URL del bridge de Zabbix para recuperar la configuración del Zabbix Bridge (se recupera cuando se inicia el adaptador del bridge).

Valor predeterminado: `ZabbixBridgeURL=https://test.aws.zabbix-lab.win/.well-known/bridge-configuration`

MCP server

Descripción general El server MCP de Zabbix es un proceso que permite a los agentes de IA trabajar con Zabbix mediante el Model Context Protocol (MCP). El server MCP se comunica con los clientes MCP y la API de Zabbix.

Mediante el server MCP, un agente de IA puede:

- Recuperar grupos de hosts, grupos de templates, hosts, templates, items, reglas LLD, interfaces, macros, triggers, problemas, períodos de mantenimiento y datos históricos de los items (excepto los items binarios);

- Realizar actualizaciones de problemas: añadir comentarios, reconocer o dejar de reconocer, cambiar la gravedad, suprimir o dejar de suprimir, o cerrar problemas;
- Crear, actualizar y eliminar períodos de mantenimiento.

Los tokens de la API de Zabbix se utilizan para la autorización; los tokens se envían con cada solicitud. El acceso está limitado por los permisos del propietario del token: solo son visibles los datos de los grupos de hosts disponibles para ese usuario, y se rechazan las acciones que el usuario no tiene permitido realizar.

El server MCP no mantiene estado: no se conserva ninguna sesión entre solicitudes.

> **Note:**
> Aunque el server MCP de Zabbix puede funcionar con versiones anteriores o posteriores de Zabbix, su compatibilidad se admite oficialmente solo con la misma versión de Zabbix que utiliza el propio server MCP.

Véase también:

- Parámetros de configuración: server MCP
- Herramientas y recursos

Instalación El paquete oficial zabbix-mcp-server está disponible en el repositorio de Zabbix.

Para instalar Zabbix MCP server desde las fuentes, compílelo con `make`.

Para configurar Zabbix MCP server, actualice los parámetros del archivo de configuración del MCP server. Como mínimo, deben configurarse los parámetros `FrontendURL` y `AllowedIP`.

El archivo de configuración predeterminado (`zabbix_mcp_server.conf`) se encuentra en `/etc/zabbix/`; deniega las herramientas que con mayor frecuencia exponen datos confidenciales. Para habilitar una herramienta, elimine la línea `DenyTool` correspondiente.

Consulte también: Prácticas recomendadas de seguridad

Ejecución Para iniciar el servidor MCP, ejecute:

```
zabbix_mcp_server -c /etc/zabbix/zabbix_mcp_server.conf
```

Para comprobar el archivo de configuración y salir sin iniciar el servidor, utilice la opción `-T`:

```
zabbix_mcp_server -c /etc/zabbix/zabbix_mcp_server.conf -T
```

Se admiten las siguientes opciones de línea de comandos:

Opción	Descripción
`-c <file>`	La ruta al archivo de configuración. El valor predeterminado es `/etc/zabbix/zabbix_mcp_server.conf`.
`-T`	Valida el archivo de configuración y sale.
`-h`	Muestra información de ayuda.
`-V`	Muestra el número de versión.

Cuando el servidor MCP se instala desde el código fuente, se debe crear manualmente una unidad de systemd, por ejemplo:

```
[Unit]
Description=Zabbix MCP server
After=network-online.target

[Service]
User=zabbix
ExecStart=/usr/local/sbin/zabbix_mcp_server -c /etc/zabbix/zabbix_mcp_server.conf
Restart=on-failure

[Install]
WantedBy=multi-user.target
```

Conexión de un cliente MCP Cualquier cliente MCP compatible con el transporte HTTP transmitible puede conectarse al servidor MCP. Se requieren los siguientes ajustes:

Ajuste	Valor
Transporte	HTTP transmitible
URL	`https://<host>:<ListenPort>/mcp`
Encabezado	`Authorization: Bearer <Zabbix API token>`, enviado con cada solicitud

Cada usuario debe configurar su propio token de API.

Para verificar la configuración, envíe una única solicitud al servidor MCP:

```
curl -sS https://mcp.example.com:8443/mcp \
  -H "Authorization: Bearer $ZABBIX_API_TOKEN" \
  -H "Content-Type: application/json" \
  -H "Accept: application/json, text/event-stream" \
  -d '{"jsonrpc":"2.0","id":1,"method":"tools/list"}'
```

El código de estado HTTP 200 y una lista de las herramientas disponibles en la respuesta confirman que se puede acceder al servidor MCP y que acepta el token; la lista muestra qué herramientas están expuestas. Otros códigos de estado indican los siguientes problemas:

- 400: el encabezado `Accept` no incluye ambos tipos de medios requeridos por el transporte;
- 401: falta el encabezado `Authorization` con el token;
- 403: la dirección del cliente no aparece en el parámetro `AllowedIP`.

Aprobación y acceso a las herramientas Si el cliente MCP admite la solicitud de formularios, el usuario debe aprobar cada cambio antes de que se envíe a Zabbix. El cliente MCP muestra al usuario qué se cambiará, y el cambio se aplica únicamente si el usuario lo confirma; si el usuario rechaza o descarta la solicitud, no se realiza ningún cambio en Zabbix. La aprobación es válida durante cinco minutos y únicamente para el cambio exacto que se mostró; durante este tiempo, la aprobación se puede reutilizar.

Si el cliente MCP no admite la solicitud de formularios (incluidos los clientes que solo admiten la solicitud mediante URL), los cambios se aplican inmediatamente, sin aprobación. En este caso, restrinja las herramientas de escritura y los permisos de la API de Zabbix.

La aprobación es especialmente importante porque el resultado de las herramientas puede contener texto no confiable procedente de sistemas supervisados, que podría influir en un agent.

Utilice AllowTool, DenyTool, AllowToolRegexp y DenyToolRegexp para controlar qué herramientas expone el server MCP. Las herramientas denegadas no se pueden llamar ni enumerar, y no se puede acceder a los recursos asociados a ellas.

Warning:
Las reglas AllowTool, DenyTool, AllowToolRegexp y DenyToolRegexp limitan el acceso a MCP, pero no cambian los permisos del token de Zabbix. Si la dirección del frontend está expuesta a un agent de IA, este puede intentar enviar llamadas directas a la API. Mantenga inaccesibles para el agent la ubicación del frontend de Zabbix y la configuración del server MCP, de modo que las solicitudes deban pasar por el server MCP y sus controles de seguridad.

Mejores prácticas de seguridad Tokens de API

La autenticación y la autorización son gestionadas íntegramente por Zabbix. El cliente MCP envía un token de API de Zabbix con cada solicitud; el server MCP lo reenvía sin almacenarlo ni registrarlo, mientras que Zabbix valida el token, aplica los permisos y registra las acciones en el registro de auditoría.

Utilice un token independiente, con los privilegios mínimos necesarios y con fecha de expiración, para cada agent y usuario. Almacene los tokens en una ubicación segura y rótelos periódicamente.

Cifrado TLS

Para evitar que los tokens se envíen en texto sin cifrar, establezca `TLSAccept=cert` y especifique el certificado y la clave (`TLSCertFile` y `TLSKeyFile`), o ejecute el servidor MCP detrás de un proxy inverso de terminación TLS de confianza y establezca `TLSAccept=unencrypted`. El protocolo HTTP sin cifrar solo es aceptable en una red de confianza.

Restricciones de acceso

Restrinja el parámetro `AllowedIP` a las direcciones de los clientes MCP de confianza y utilice los permisos de Zabbix para limitar lo que pueden hacer los tokens de API.

El servidor MCP no proporciona limitación de la tasa de solicitudes. Cada solicitud MCP normalmente da lugar a una llamada a la API de Zabbix, aunque un agent que se ejecute en un bucle aún puede generar una carga considerable.

Protección de datos confidenciales

Las claves de los item pueden contener credenciales (por ejemplo, en los item de SSH, JMX, ODBC y HTTP agent), los valores de las macros pueden contener contraseñas y cadenas de comunidad, las interfaces revelan direcciones internas y los valores de los item de registro o de texto pueden contener cualquier dato. Por este motivo, las herramientas `item_get`, `lld_get`, `interface_get`, `macro_get` y `history_get` están denegadas de forma predeterminada en el archivo de configuración incluido.

Para evitar que los valores confidenciales de las macros queden expuestos al proveedor de LLM del agent conectado, utilice macros secretas; sus valores nunca son devueltos por la API.

Herramientas y recursos

Descripción general

El server MCP de Zabbix expone la funcionalidad de Zabbix a los clientes MCP como *tools* y *resources*.

Los nombres de las tools describen la actividad. Si se establece el parámetro ToolPrefix, el prefijo se añade a cada nombre de tool; las reglas de acceso a las tools coinciden con los nombres simples que se indican a continuación.

Herramientas de lectura

Herramienta	Método de la API de Zabbix	Argumentos	Ordenado por	Recurso
`hostgroup_get`	hostgroup.get	`groupids`, `names`, `search`, `limit`	name	`zabbix://hostgroup/<i`
`templategroup_get`	templategroup.get	`groupids`, `names`, `search`, `limit`	name	`zabbix://templategrou`
`host_get`	host.get	`hostids`, `groupids`, `search`, `limit`	name	`zabbix://host/<id>`
`template_get`	template.get	`templateids`, `groupids`, `search`, `limit`	name	`zabbix://template/<id`
`item_get`	item.get	`hostids`, `itemids`, `search`, `limit`	name	`zabbix://item/<id>`
`lld_get`	discoveryrule.get	`hostids`, `ruleids`, `limit`	name	-
`interface_get`	hostinterface.get	`hostids`, `interfaceids`, `limit`	ID de la interfaz	-
`macro_get`	usermacro.get	`hostids` o `global`, `macro_names`, `limit`	macro	-
`problem_get`	problem.get	`eventids`, `groupids`, `hostids`, `severities`, `acknowledged`, `suppressed`, `recent`, `limit`	ID del evento, descendente	-
`maintenance_get`	maintenance.get	`maintenanceids`, `groupids`, `hostids`, `limit`	name	-
`history_get`	history.get	`itemids`, `time_from`, `time_till`, `type`, `limit`	clock, descendente	-
`trigger_get`	trigger.get	`triggerids`, `hostids`, `search`, `limit`	description	`zabbix://trigger/<id>`

Tenga en cuenta lo siguiente:

- `host_get` y `template_get` también devuelven los triggers de cada objeto (un conjunto limitado de campos), las etiquetas y los templates vinculados. Los items, las reglas de descubrimiento, las interfaces y las macros se devuelven mediante las herramientas correspondientes.
- `history_get` requiere un intervalo de tiempo no superior a 7 días y cuyo inicio no sea anterior a hace 7 días. El argumento `type` debe ser uno de `float`, `unsigned`, `character`, `text`, `log` o `json`, y debe coincidir con el tipo de valor de los items; si el tipo es incorrecto, se devuelve un resultado vacío, que no puede distinguirse de un item sin datos. El historial de los items binarios (tipo de valor 5) no se puede recuperar mediante el servidor MCP.
- Las herramientas de lectura solicitan una lista fija de campos; nunca se solicitan todos los campos (`output: extend`).
- Los resultados de las herramientas que tienen un recurso contienen el campo `uri`, de modo que un agent puede pasar de una lista a una entidad individual.

Formato de respuesta

Todas las herramientas de lectura devuelven la respuesta en el mismo formato:

```
{
  "results": [ ... ],
  "count": 2,
  "truncated": true,
  "truncated_by": "count",
  "limit": 2,
  "note": "Se alcanzó el límite de cantidad de resultados ..."
}
```

La respuesta está limitada por el argumento `limit` (valor predeterminado: 100, máximo: 1000) y por el tamaño serializado de la respuesta (MaxResponseSize, valor predeterminado: 64K). La respuesta se detiene cuando se alcanza cualquiera de los límites. La respuesta se trunca únicamente entre resultados, nunca dentro de un resultado, y una respuesta truncada indica qué límite se

alcanzó. Cada herramienta tiene un orden de clasificación fijo, por lo que una respuesta truncada siempre contiene el principio del conjunto completo de resultados. No se admite la paginación; para obtener otros resultados, limite la consulta.

Herramientas de problemas

Todas las herramientas de problemas llaman a event.acknowledge con las marcas de acción correspondientes.
El argumento `eventids` es obligatorio; los ID de evento se pueden obtener con `problem_get`.
Se puede añadir un mensaje (comentario) con cualquier herramienta de problemas.
El argumento `message` es opcional, excepto para `problem_annotate`, donde es obligatorio; si se especifica, el mensaje se registra en la misma llamada a la API de Zabbix.

Tool	Effect	Additional arguments
`problem_acknowledge`	Reconocer el problema.	-
`problem_unacknowledge`	Eliminar el reconocimiento.	-
`problem_close`	Cerrar el problema manualmente (solo para triggers que permiten el cierre manual).	-
`problem_annotate`	Registrar un mensaje.	`message` (obligatorio)
`problem_change_severity`	Cambiar la gravedad del problema.	`severity` (0-5)
`problem_suppress`	Suprimir el problema hasta una hora especificada o indefinidamente.	`suppress_until`
`problem_unsuppress`	Eliminar la supresión.	-

No se pueden crear ni eliminar problemas; todas las herramientas de problemas actualizan problemas existentes.

Herramientas de mantenimiento

Herramienta	Método de la API de Zabbix	Argumentos
`maintenance_create`	maintenance.create	`name`, `groupids` y/o `hostids`, `active_since`, `active_till`, `timeperiods`, `description`, `tags`, `tags_evaltype`
`maintenance_update`	maintenance.update	`maintenanceid` y cualquiera de los argumentos anteriores. Los argumentos de tipo lista reemplazan el conjunto completo de valores; una lista vacía borra el conjunto.
`maintenance_delete`	maintenance.delete	`maintenanceids`

Cuando se actualiza un período de mantenimiento, los argumentos de tipo lista (como `hostids`, `groupids` y `timeperiods`) reemplazan los valores existentes en lugar de añadirse a ellos. Por ejemplo, si un período de mantenimiento incluye los hosts A y B, y se llama a `maintenance_update` con `hostids` que contiene únicamente el host C, el período de mantenimiento incluirá solo el host C.

`maintenance_create` es la única operación que no es idempotente, y `maintenance_delete` es la única operación que no se puede revertir. Para impedir su uso, deniegue estas herramientas mediante el parámetro DenyTool.

Comportamiento de las herramientas de escritura

Las herramientas de problemas y mantenimiento son *herramientas de escritura*. Funcionan de la siguiente manera:

- Los argumentos se validan antes de realizar cualquier llamada a la API de Zabbix; si la validación falla, el error indica la restricción que no se cumplió.
- Si el cliente lo admite, el cambio se aplica únicamente después de que el usuario lo apruebe.
- Zabbix realiza el cambio utilizando el token del autor de la llamada. Zabbix comprueba los permisos y registra el cambio en el registro de auditoría asociado al propietario del token.
- Los errores de Zabbix, incluidos los errores de permisos, se devuelven sin modificar como un objeto de error con los campos `code`, `message` y `data`.
- Las herramientas de escritura devuelven directamente el resultado de Zabbix; los límites de respuesta de las herramientas de lectura no se aplican.
- El servidor MCP nunca vuelve a intentar las llamadas de escritura.

Recursos

Los recursos utilizan el esquema de URI `zabbix://<kind>/<id>`. Solo `zabbix://about` se incluye en la respuesta de `resources/list`; los ID de las entidades no se pueden enumerar y se obtienen de los resultados de las herramientas. La respuesta de `resources/templates/list` muestra los tipos de entidades.

URI	Contenido
`zabbix://about`	La URL de la API, la URL del frontend, la versión de Zabbix, la identidad a la que se resuelve el token del autor de la llamada y la lista de herramientas expuestas.
`zabbix://host/<hostid>`	El registro del host sin items, triggers, interfaces ni macros.
`zabbix://hostgroup/<groupid>`	El registro del grupo de hosts sin la lista de miembros.
`zabbix://template/<templateid>`	El registro del template sin items, triggers ni reglas de descubrimiento.
`zabbix://templategroup/<groupid>`	El registro del grupo de templates sin la lista de miembros.
`zabbix://item/<itemid>`	La definición del item sin el valor actual.
`zabbix://trigger/<triggerid>`	La definición del trigger sin hosts, dependencias, funciones ni etiquetas.

Un recurso cuya herramienta correspondiente está denegada no existe. Tanto una entidad inexistente como una entidad a la que el propietario del token no tiene acceso se notifican como *resource not found*; estos casos son deliberadamente indistinguibles. El recurso `zabbix://about` no se ve afectado por las reglas de acceso a las herramientas.

Parámetros de configuración: Zabbix MCP server

Descripción general

El server MCP de Zabbix es un proceso que permite a los agentes de IA trabajar con Zabbix mediante el Model Context Protocol (MCP).

En esta sección se enumeran los parámetros compatibles con el archivo de configuración del server MCP de Zabbix (`zabbix_mcp_server.conf`).

Los parámetros se enumeran sin información adicional. Haga clic en un parámetro para consultar todos los detalles.

Parámetro	Descripción
AllowedIP	Una lista de direcciones IP de los clientes MCP, separadas por comas y, opcionalmente, en notación CIDR, o nombres DNS de los clientes MCP.
AllowTool	Una regla con comodines que permite una herramienta.
AllowToolRegexp	Una regla de expresión regular que permite herramientas.
DebugLevel	El nivel de depuración.
DenyTool	Una regla con comodines que deniega una herramienta.
DenyToolRegexp	Una regla de expresión regular que deniega herramientas.
FrontendURL	El endpoint de la API de Zabbix.
IgnoreURLCertErrors	Especifica cómo gestionar los errores de validación de certificados TLS al acceder a la API de Zabbix.
ListenIP	La dirección IP en la que se escucharán las conexiones.
ListenPort	El puerto en el que se escucharán las conexiones.
LogFile	El nombre del archivo de registro.
LogFileSize	El tamaño máximo del archivo de registro.
LogType	El tipo de salida del registro.
MaxResponseSize	El tamaño máximo de una respuesta devuelta por una herramienta de lectura.
PidFile	El nombre del archivo PID.
Timeout	El tiempo máximo (en segundos) de espera de una respuesta de la API de Zabbix.
TLSAccept	Qué conexiones entrantes se aceptarán.
TLSCertFile	La ruta completa de un archivo que contiene el certificado del server o la cadena de certificados.
TLSKeyFile	La ruta completa de un archivo que contiene la clave privada del server.
ToolPrefix	Un prefijo añadido a los nombres de todas las herramientas.

Todos los parámetros son opcionales, a menos que se indique explícitamente que el parámetro es obligatorio.

Tenga en cuenta lo siguiente:

- Los valores predeterminados reflejan los valores predeterminados del proceso, no los valores de los archivos de configuración incluidos;
- Los valores admiten variables de entorno en el formato `${NAME}`;
- Zabbix admite archivos de configuración únicamente con codificación UTF-8 y sin BOM;
- Los comentarios que comienzan con # solo se admiten al principio de la línea.

Detalles de los parámetros

AllowedIP

Una lista de direcciones IP separadas por comas, opcionalmente en notación CIDR, o nombres DNS de clientes MCP. Las conexiones entrantes solo se aceptarán desde los hosts incluidos aquí.

Ejemplo:

`AllowedIP=127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,mcp-client.example.com`

Obligatorio: sí

AllowTool

Una regla de comodín que permite una herramienta. El parámetro se puede especificar varias veces.
 Los patrones de comodín utilizan la sintaxis de clave de item, donde * coincide con cualquier carácter. La regla coincide con el nombre simple de la herramienta e ignora ToolPrefix.
 Todas las reglas `AllowTool`, `DenyTool`, `AllowToolRegexp` y `DenyToolRegexp` forman una única lista en el orden en que se especifican; la primera regla coincidente determina el resultado. Una herramienta que no coincide con ninguna regla está permitida.

Ejemplo:

`AllowTool=host_get`

AllowToolRegexp

Una regla de expresión regular que permite herramientas. El parámetro se puede especificar varias veces.
 Las expresiones regulares utilizan la sintaxis RE2 y coinciden con cualquier parte del nombre de la herramienta sin formato, a menos que se anclen.

Ejemplo:

`AllowToolRegexp=^(host|hostgroup|trigger|problem)_get$`

DebugLevel

Especifique el nivel de depuración: - *0* - información básica sobre el inicio y la detención de los procesos de Zabbix; - *1* - información crítica; - *2* - información sobre errores; - *3* - advertencias; - *4* - para la depuración (genera mucha información); - *5* - depuración ampliada (genera aún más información).

Valor predeterminado: 3
 Rango: 0-5

DenyTool

Una regla de comodín que deniega una herramienta. El parámetro se puede especificar varias veces.
 Una herramienta denegada no se incluye en la lista de herramientas y no se puede llamar. Consulte AllowTool para obtener más información.

Ejemplo:

`DenyTool=maintenance_delete`

DenyToolRegexp

Una regla de expresión regular que deniega herramientas. El parámetro se puede especificar varias veces.
 Consulte AllowToolRegexp para obtener más información.

Ejemplo:

`DenyToolRegexp=^problem_(close|suppress)$`

FrontendURL

El endpoint de la API de Zabbix: la URL del frontend de Zabbix que termina en `api_jsonrpc.php`.

Ejemplo:

`FrontendURL=https://zabbix.example.com/api_jsonrpc.php`

Obligatorio: sí

IgnoreURLCertErrors

Especifica el manejo de los errores de validación de certificados TLS al acceder a la API de Zabbix:
 false - no ignorar los errores de certificado;
 true - ignorar los errores de certificado (usar únicamente en un entorno de desarrollo).

Valor predeterminado: `false`

ListenIP

La dirección IP en la que se escuchará.

Valor predeterminado: `0.0.0.0`

ListenPort

El puerto en el que se escuchará. La ruta del endpoint MCP es `/mcp`.

Valor predeterminado: 8443
 Rango: 1024-32767

LogFile

El nombre del archivo de registro.

Valor predeterminado: `/tmp/zabbix_mcp_server.log`

Obligatorio: No

LogFileSize

El tamaño máximo de un archivo de registro en MB.
0 - desactiva la rotación automática de registros.

Valor predeterminado: 1
 Rango: 0-1024

LogType

El tipo de salida del registro: - *file* - escribe el registro en el archivo especificado por el parámetro LogFile; - *system* - escribe el registro en syslog; - *console* - escribe el registro en la salida estándar.

Valor predeterminado: `file`

MaxResponseSize

El tamaño máximo de una respuesta devuelta por una herramienta de lectura. La respuesta se detiene cuando se alcanza primero este límite o el argumento `limit` de la herramienta.
 Admite sufijos de unidades.

Valor predeterminado: 64K
 Rango: 16K-256K

PidFile

El nombre del archivo PID.

Valor predeterminado: `/tmp/zabbix_mcp_server.pid`

Tiempo de espera

El tiempo máximo (en segundos) que se debe esperar para recibir una respuesta de la API de Zabbix.

Valor predeterminado: 10
 Rango: 3-30

TLSAccept

Qué conexiones entrantes aceptar: - *sin cifrar*: aceptar conexiones sin cifrado (predeterminado); - *cert*: aceptar conexiones con TLS y un certificado.

Predeterminado: `unencrypted`

TLSCertFile

La ruta completa del archivo que contiene el certificado del server o la cadena de certificados.

Obligatorio: Sí, si `TLSAccept` está establecido en `cert`; de lo contrario, no

TLSKeyFile

La ruta completa del archivo que contiene la clave privada del server.

Obligatorio: Sí, si `TLSAccept` está establecido en `cert`; de lo contrario, no

ToolPrefix

Un prefijo añadido a los nombres de todas las herramientas, por ejemplo, para evitar conflictos de nombres con las herramientas de otros servidores MCP. Por ejemplo, con `ToolPrefix=zbx_`, la herramienta `host_get` se expone como `zbx_host_get`.
 Caracteres permitidos: letras, dígitos, guion bajo (`_`), guion (`-`) y punto (`.`).
 Las reglas de acceso a las herramientas coinciden con los nombres de las herramientas sin el prefijo.

Ejemplo:

```
ToolPrefix=zbx_
```

Variables de entorno

Resumen Las variables de entorno permiten configurar los componentes de Zabbix sin codificar valores directamente en los archivos de configuración. Esto facilita la gestión de configuraciones en entornos dinámicos, como Docker, donde las variables se pueden pasar en tiempo de ejecución para adaptarse a diferentes configuraciones.

En el caso más simple, al establecer el valor del parámetro de configuración DebugLevel del server de Zabbix en una variable de entorno, puede usarla para configurar el server al iniciarse:

```
### Archivo de configuración del server de Zabbix:
DebugLevel=${NEW_DEBUG_LEVEL}

### Inicio del server de Zabbix:
NEW_DEBUG_LEVEL=5 /usr/sbin/zabbix_server
```

Las variables de entorno son compatibles con los siguientes componentes de Zabbix:

- Server
- Proxy
- Agent (UNIX o Windows)
- Agent 2 (UNIX o Windows), incluidos los plugins
- Servicio web
- Zabbix sender (cuando se usa la opción -c, --config)

Notas importantes

- Cuando un parámetro de configuración se establece mediante una variable de entorno, y esta no se especifica al ejecutar el componente, se utiliza el valor predeterminado del parámetro.
- Al usar comandos en tiempo de ejecución (por ejemplo, para aumentar el nivel de registro del agent), deben especificarse todas las variables de entorno utilizadas previamente. Esto se debe a que los componentes de Zabbix usan su archivo de configuración para ejecutar comandos en tiempo de ejecución; si se omiten las variables de entorno, se usarán los valores predeterminados de los parámetros de configuración. Consulte Ejemplos.
- El comando en tiempo de ejecución `userparameter_reload` no admite la recarga de variables de entorno. Durante la recarga, las variables se ignoran y solo se recargan los parámetros con valores normales.
- Las variables de entorno actuales del proceso, que se usaron en los archivos de configuración, se eliminan después de que se inicia el componente de Zabbix. Esto garantiza que los procesos secundarios (por ejemplo, scripts remotos ejecutados por Zabbix) no puedan acceder a estas variables. Sin embargo, tenga en cuenta que las variables iniciales del proceso aún pueden recuperarse (por ejemplo, mediante el archivo `/proc/<PID>/environ`).

Sintaxis Las variables de entorno deben usar la siguiente sintaxis: `${alfanuméricos/guiones bajos}`.

El nombre de la variable solo puede incluir letras (a-z, A-Z), guiones bajos (_), y dígitos (0-9), y no debe comenzar con un dígito.

Las variables que no coincidan con la sintaxis requerida o que se combinen con un valor normal se tratarán como valores normales, lo que puede producir errores.

Sintaxis correcta de variables:

```
DebugLevel=${NEW_DEBUG_LEVEL}
Hostname=${ZBX_HOSTNAME}
LogFile=${LogFile_001}
```

Sintaxis incorrecta de variables:

```
DebugLevel=${5_DebugLevel}
Hostname=${ZBX.HOSTNAME 1}
LogFile=/${HOME}/zabbix/zabbix_server.log
```

> **Note:**
> En Windows, los nombres de las variables de entorno no distinguen entre mayúsculas y minúsculas.

Ejemplos Los siguientes ejemplos muestran cómo configurar y usar variables de entorno con los componentes de Zabbix.

Ejemplo 1: Configuración y prueba del agent de Zabbix

1. Establezca las variables de entorno en el archivo de configuración del agent:

```
Hostname=${ZBX_HOSTNAME}
ServerActive=${ServerActive}
```

2. Pruebe el archivo de configuración:

```
ZBX_HOSTNAME="New Zabbix agent" ServerActive=127.0.0.1 /usr/sbin/zabbix_agentd -c /etc/zabbix/zabbix_agent
```

3. Inicie el agent con las variables de entorno:

```
ZBX_HOSTNAME="New Zabbix agent" ServerActive=127.0.0.1 /usr/sbin/zabbix_agentd -c /etc/zabbix/zabbix_agent
```

Al utilizar comandos en tiempo de ejecución (por ejemplo, para aumentar el nivel de registro del agent), se deben especificar las variables de entorno utilizadas anteriormente:

```
ZBX_HOSTNAME="New Zabbix agent" ServerActive=127.0.0.1 /usr/sbin/zabbix_agentd -c /etc/zabbix/zabbix_agent
```

Esto se debe a que el agent utiliza su archivo de configuración para ejecutar comandos en tiempo de ejecución; si se omiten las variables de entorno, se utilizarán los valores predeterminados de los parámetros de configuración.

Como alternativa, después de establecer las variables de entorno en el archivo de configuración del agent, puede ponerlas a disposición de los procesos (por ejemplo, mediante el comando `export`). Esto reduce el riesgo de un comportamiento inesperado debido a variables ausentes o configuradas incorrectamente.

```
export ZBX_HOSTNAME="New Zabbix agent"
export ServerActive=127.0.0.1
/usr/sbin/zabbix_agentd -c /etc/zabbix/zabbix_agentd.conf --test-config
/usr/sbin/zabbix_agentd -c /etc/zabbix/zabbix_agentd.conf
/usr/sbin/zabbix_agentd -c /etc/zabbix/zabbix_agentd.conf -R log_level_increase
```

Ejemplo 2: Configuración del agent de Zabbix para un contenedor

Si está creando y configurando su propia imagen personalizada para los componentes de Zabbix (por ejemplo, el agent de Zabbix), puede definir los parámetros de configuración mediante variables de entorno y, a continuación, iniciar el contenedor con esas variables.

1. Al preparar la imagen, establezca las variables de entorno en el archivo de configuración del agent:

```
Hostname=${ZBX_HOSTNAME}
BufferSize=${BUFSZ}
ListenPort=${LISTENPORT}
UserParameter=${_UsrPar01}
UserParameter=${_UsrPar02}
```

2. Después de crear la imagen del contenedor, inicie el contenedor del agent (por ejemplo, Docker) con variables de entorno:

```
docker run --name my-zabbix-agent -e ZBX_HOSTNAME="new-hostname" -e BUFSZ=1000 -e LISTENPORT=20050 -e _Usr
```

3. Al utilizar comandos en tiempo de ejecución (por ejemplo, para aumentar el nivel de registro del agent), acceda al shell del contenedor y ejecute el comando en tiempo de ejecución:

```
docker exec -it <containerid> sh
/usr/sbin/zabbix_agentd -R log_level_increase
```

> **Attention:**
> El comando en tiempo de ejecución `userparameter_reload` no admite la recarga de variables de entorno. Durante la recarga, las variables se ignoran y solo se recargan los parámetros con valores normales.

Configuración

Resumen La configuración en Zabbix implica especificar qué hosts y sistemas supervisar, definir qué datos recopilar y establecer cómo se entregan las notificaciones cuando se producen problemas.

Utilice la barra lateral para navegar por los aspectos de configuración relevantes.

Tareas clave de configuración

- Configurar un nuevo host o grupo de hosts → Hosts and host groups
- Definir qué datos recopilar de los hosts → Items
- Crear reglas para detectar problemas → Problem detection with triggers
- Configurar cómo se entregan las notificaciones → Notifications upon events
- Mostrar datos de monitorización mediante paneles, gráficos y mapas → Visualization

- Reutilizar configuraciones en varios hosts → Templates and template groups, Templates out of the box
- Controlar el acceso de los usuarios y los permisos → Users and user groups

Mejores prácticas de configuración Las decisiones de configuración afectan al rendimiento y a la mantenibilidad. Aunque cada instalación de Zabbix es diferente, estas mejores prácticas pueden servir como base para realizar los ajustes necesarios.

Host separado para cada instancia de monitoreo

Un host en Zabbix es una entidad lógica más que una máquina o dispositivo físico. Cree un host separado para cada instancia independiente, como una máquina virtual, base de datos, contenedor o switch de red.

Este enfoque proporciona:

1. Datos de monitoreo más limpios, con items, triggers y notificaciones de alerta separados para cada host.
2. Control de acceso de usuarios más granular mediante user-roles, alineado con el principio de privilegio mínimo.

Usar templates y actualización masiva para hosts similares

Para hosts similares como *Network switch 1* y *Network switch 2*, puede crear un template con las métricas necesarias (*Network switch template*) y agrupar los hosts en un grupo de hosts (*Network switches*).

Al agregar nuevos switches, puede usar la actualización masiva en *Data Collection* > *Hosts* para vincular el template en lugar de clonar los hosts manualmente.

Configure elementos maestro y dependientes para minimizar solicitudes

Para minimizar las solicitudes a la entidad de destino, configure un elemento maestro para recopilar datos en una sola solicitud y, después, use elementos dependientes con preprocesamiento para extraer valores específicos.

Por ejemplo, el elemento maestro podría recopilar una respuesta JSON o XML con varias métricas, o ejecutar una consulta de base de datos que devuelva varias columnas (por ejemplo, número de conexiones abiertas, conexiones abortadas, máximo de conexiones simultáneas permitidas y total acumulado de conexiones desde el inicio), y los elementos dependientes analizarán y almacenarán cada valor por separado.

Descartar el historial del elemento maestro después de la recopilación y conservar solo los datos de los elementos dependientes.

Implementa un proxy de Zabbix cuando sea necesario

Si todos tus hosts se encuentran en la misma red local que el server de Zabbix y no existen problemas de escalabilidad ni de rendimiento, es posible que no necesites un proxy. En entornos más grandes o complejos, un proxy permite distribuir la carga de manera más uniforme.

Añade un proxy de Zabbix cuando:

1. Supervises varios hosts mediante distintos métodos de recopilación de métricas detrás de un firewall. El proxy recopila datos de los hosts y los reenvía al server de Zabbix, lo que reduce el número de puertos abiertos en el firewall.
2. Supervises ubicaciones remotas, sucursales o redes. Los proxies de Zabbix en ubicaciones remotas continúan recopilando datos durante las interrupciones de red y los envían cuando se restablece la conexión.
3. Administres implementaciones a gran escala. Si tienes muchos hosts o recopilas un gran número de valores por segundo, implementa un proxy para reducir la carga del server y mejorar el rendimiento.

Configurar la recuperación independiente de macros secretas

Es posible que desee usar macros de usuario secret como texto secreto o como macros de almacén secreto.

De forma predeterminada, los valores de las macros secretas son recuperados por Zabbix server y propagados a los proxies. Para una seguridad mejorada con macros de almacén secreto, configure los valores de las macros para que sean recuperados de forma independiente por Zabbix server y los proxies.

Configuración de un template

Descripción general

La configuración de una template requiere que primero cree una template definiendo sus parámetros generales y luego agregue entidades (items, triggers, gráficos, etc.) a la misma.

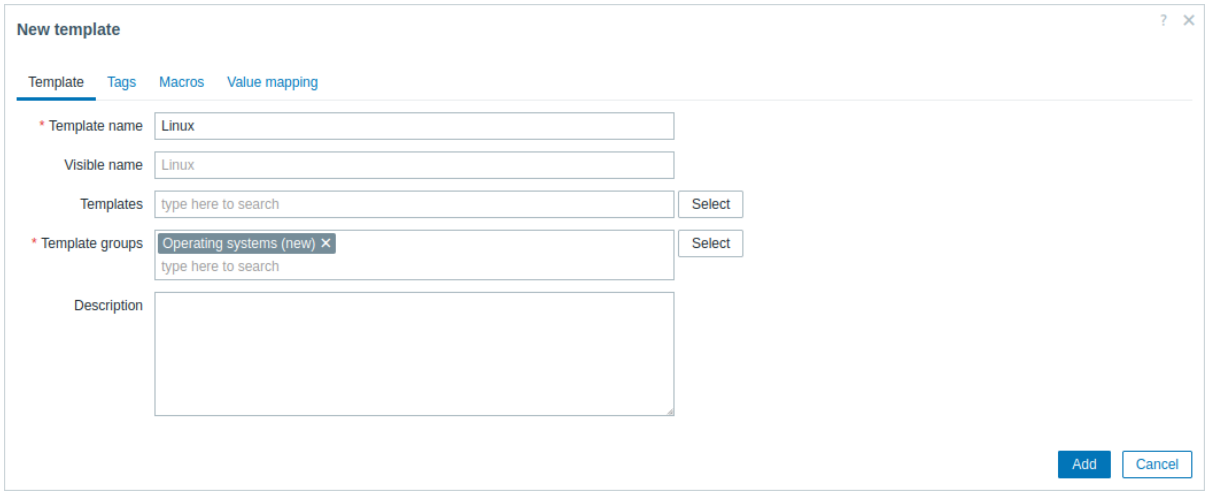
Crear un template

Para crear un template, haga lo siguiente:

1. Vaya a *Recopilación de datos* > *Templates*.
2. Haga clic en *Crear template*.

3. Edite los atributos del template.

La pestaña **Template** contiene los atributos generales del template.

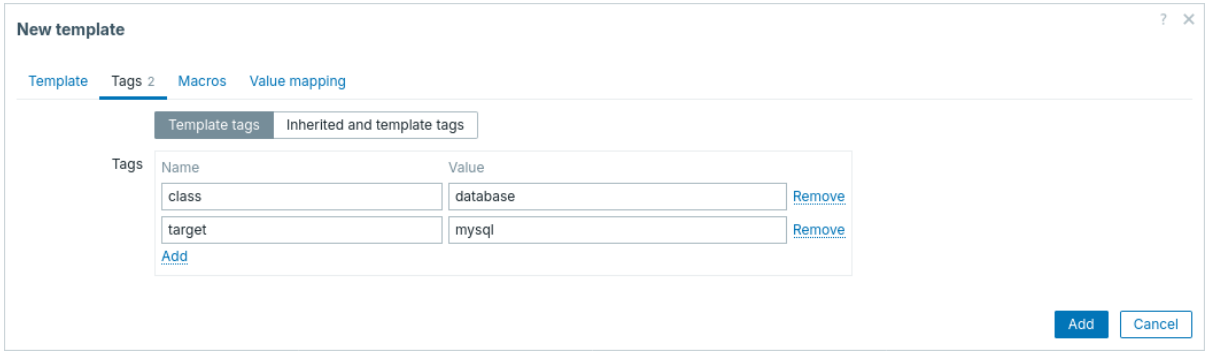


Todos los campos de entrada obligatorios están marcados con un asterisco rojo.

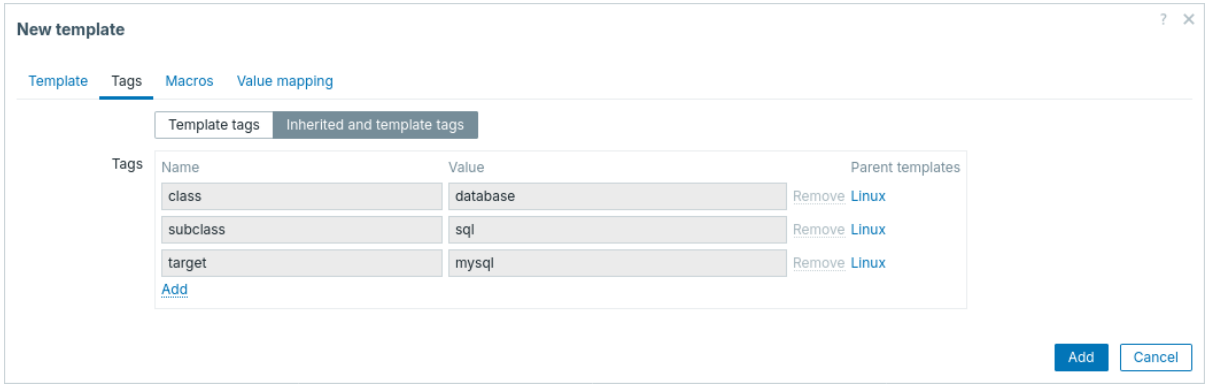
Atributos del template:

Parámetro	Descripción
Nombre del template	Nombre único del template. Se permiten caracteres alfanuméricos, espacios, puntos, guiones y guiones bajos. No se permiten espacios iniciales ni finales.
Nombre visible	Si establece este nombre, será el que se muestre en listas, mapas, etc.
Templates	Vincule uno o más templates a este template. Todas las entidades (items, triggers, etc.) se heredarán de los templates vinculados. Para vincular un nuevo template, escriba su nombre en el campo *Templates*; se mostrará una lista desplegable con los templates coincidentes. Como alternativa, haga clic en *Seleccionar* para abrir la ventana emergente *Templates*, que inicialmente no muestra ningún template. Para rellenar la lista, escriba el nombre de un grupo de templates en el campo *Grupo de templates* (y seleccione uno de la lista sugerida) o haga clic en el botón *Seleccionar* situado junto al campo *Grupo de templates* para abrir la ventana emergente *Grupos de templates*. Una vez que seleccione un grupo de templates, la ventana emergente *Templates* se actualizará para mostrar los templates pertenecientes a ese grupo. Los templates seleccionados se vincularán al guardar o actualizar la configuración. Para desvincular un template, utilice una de las dos opciones del bloque *Templates*: *Desvincular*: desvincula el template, pero conserva sus entidades (items, triggers, etc.); *Desvincular y limpiar*: desvincula el template y elimina todas sus entidades (items, triggers, etc.).
Grupos de templates	Grupos de templates a los que pertenece el template.
Descripción	Descripción del template.
Proveedor y versión	Proveedor y versión del template; se muestran únicamente al actualizar templates existentes (templates incluidos de fábrica proporcionados por Zabbix, templates importados o templates modificados mediante la API de Template) si la configuración del template contiene esta información. No se puede modificar en el frontend de Zabbix. Para los templates incluidos de fábrica, la versión se muestra de la siguiente manera: versión principal de Zabbix, delimitador (”-”), número de revisión (incrementado con cada nueva versión del template y restablecido con cada versión principal de Zabbix). Por ejemplo, 7.0-0, 7.0-5, 8.0-0, 8.0-3.

La pestaña **Etiquetas** permite definir etiquetas a nivel de template. Todos los problemas de los hosts vinculados a este template se etiquetarán con los valores introducidos aquí.



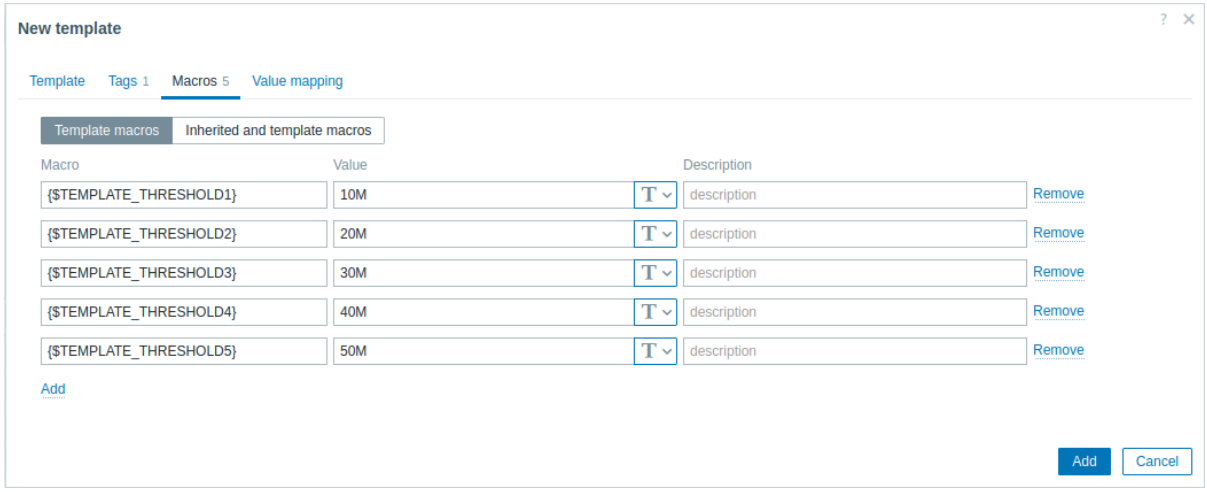
Si selecciona la opción *Etiquetas heredadas y del template*, también podrá ver las etiquetas de los templates vinculados que heredará el template.



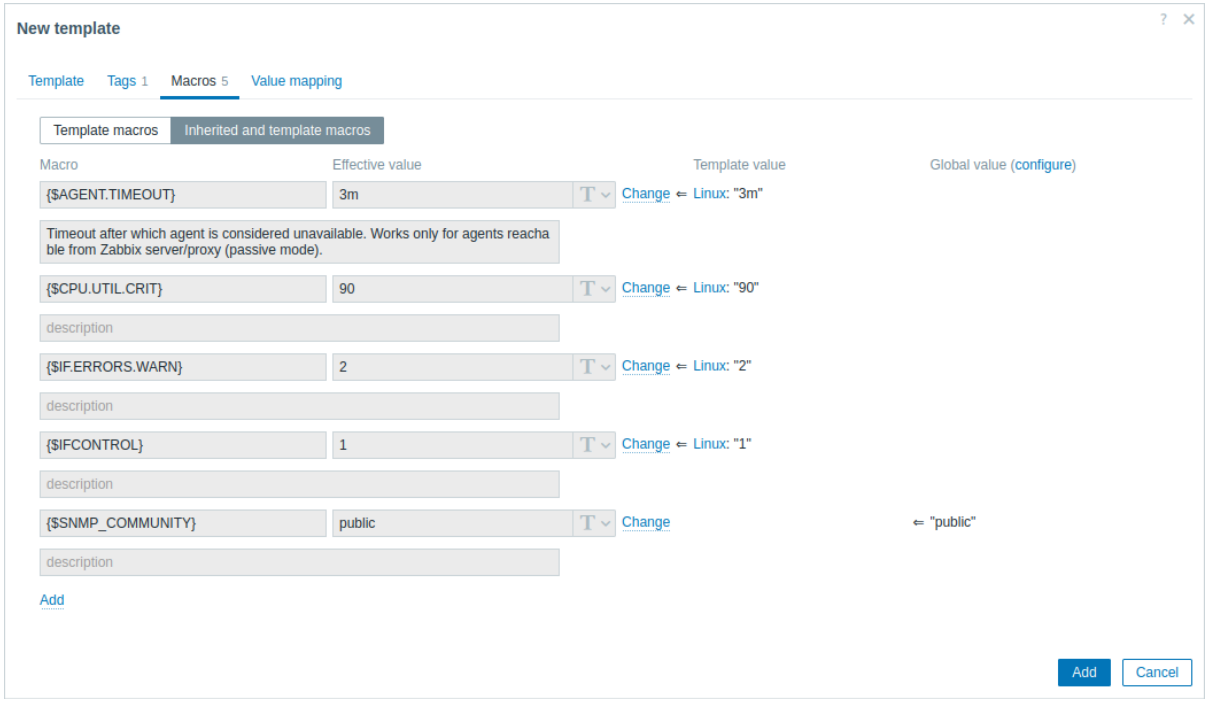
Para mayor comodidad, se proporcionan enlaces a los templates correspondientes.

Las macros de usuario, las macros {INVENTORY.*}, {HOST.HOST}, {HOST.NAME}, {HOST.CONN}, {HOST.DNS}, {HOST.IP}, {HOST.PORT} y {HOST.ID} son compatibles con las etiquetas.

La pestaña **Macros** permite definir macros de usuario a nivel de template como pares nombre-valor. Tenga en cuenta que los valores de las macros pueden conservarse como texto sin formato, texto secreto o secreto de Vault. También se admite añadir una descripción.



Si selecciona la opción *Macros heredadas y del template*, también podrá ver las macros de los templates vinculados y las macros globales que heredará el template, así como los valores a los que se resolverán las macros.



Para mayor comodidad, se proporcionan enlaces a los templates correspondientes, así como un enlace a la configuración de macros globales. También es posible editar una macro de un template vinculado o una macro global a nivel de template, creando efectivamente una copia de la macro en el template.

La pestaña **Asignación de valores** permite configurar una representación comprensible para las personas de los datos de los items en las asignaciones de valores.

Botones:

Add	Añade el template. El template añadido debería aparecer en la lista.
Update	Actualiza las propiedades de un template existente.
Clone	Crea otro template basándose en las propiedades del template actual. Esto **incluye** las entidades (items, triggers, etc.) heredadas de los templates vinculados y asociadas directamente al template actual, pero **excluye** el proveedor y la versión del template actual para que el template clonado se pueda distinguir del original.
Delete	Elimina el template; las entidades del template (items, triggers, etc.) permanecen en los hosts vinculados.
Delete and clear	Elimina el template y todas sus entidades de los hosts vinculados.
Cancel	Cancela la edición de las propiedades del template.

Añadir items, triggers y gráficos

> **Attention:**
> Los items deben añadirse primero a un template. Los triggers y los gráficos no se pueden añadir sin el item correspondiente.

Puede añadir items existentes a un template o crear nuevos items.

Para añadir items al template:

1. Vaya a *Data collection* > *Hosts* (o *Templates*).
2. Haga clic en *Items* en la fila del host/template requerido.
3. Marque las casillas de los items que desea añadir al template.
4. Haga clic en *Copy* debajo de la lista de items.
5. Seleccione el template (o grupo de templates) al que se deben copiar los items y haga clic en *Copy*.
 Todos los items seleccionados deben copiarse al template.

La adición de triggers y gráficos se realiza de forma similar (desde la lista de triggers y gráficos, respectivamente), teniendo en cuenta de nuevo que solo se pueden añadir si primero se añaden los items requeridos.

Añadir dashboards

Para añadir dashboards a un template en *Data collection* > *Templates*, haga lo siguiente:

1. Haga clic en *Dashboards* en la fila del template.
2. Configure un dashboard siguiendo las directrices de configurar dashboards.

Attention:
Al configurar widgets en un dashboard de template (en lugar de un dashboard global), los parámetros relacionados con el host no están disponibles, y algunos parámetros tienen una etiqueta diferente. Esto se debe a que los dashboards de template muestran datos solo del host al que está vinculado el template. Por ejemplo, los parámetros *Host groups*, *Exclude host groups* y *Hosts* en el widget *Problems* no están disponibles, el parámetro *Host groups* en el widget *Host availability* no está disponible, y el parámetro *Show hosts in maintenance* se renombra a *Show data in maintenance*, etc. Para obtener más información sobre la disponibilidad de parámetros en los widgets de dashboard de template, consulte los parámetros específicos de cada widget de dashboard.

Note:
Para obtener detalles sobre el acceso a los dashboards del host que se crean a partir de dashboards de template, consulte la sección dashboards del host.

Configuración de reglas de descubrimiento de bajo nivel

Consulte la sección descubrimiento de bajo nivel del manual.

Añadir escenarios web

Para añadir escenarios web a un template en *Data collection* > *Templates*, haga lo siguiente:

1. Haga clic en *Web* en la fila del template.
2. Configure un escenario web siguiendo el método habitual de configurar escenarios web.

Configuración de un grupo de template

Resumen

Los grupos de templates se utilizan para la agrupación lógica de templates y para asignarles permisos de usuario.

Cada template debe tener asignado al menos un grupo de templates. Un template puede pertenecer a varios grupos de templates, y cada grupo de templates puede contener varios templates.

Tenga en cuenta que, en Zabbix, todos los permisos se basan en grupos: grupos de usuarios, grupos de host y grupos de templates. Por lo tanto, incluso si un único usuario necesita acceso a un único template, este se concede agregando al usuario a un grupo de usuarios y asignando a ese grupo de usuarios permisos sobre el grupo de templates que contiene ese template.

Configuración

Attention:
Solo los usuarios Super admin pueden crear grupos de template.

Hay dos opciones para crear un grupo de template en el frontend de Zabbix.

Opción uno:

1. Vaya a *Data collection* > *Template groups*.
2. Haga clic en *Create template group* en la esquina superior derecha de la pantalla.
3. Introduzca el nombre del grupo en el formulario.



Opción dos: al configurar un template, introduzca un nombre de grupo que no exista en el campo de entrada *Template groups*.

Una vez creado el grupo de template, puede hacer clic en el nombre del template en la lista, en *Data collection* > *Template groups*, para editar el nombre del grupo, clonar el grupo o eliminarlo.

Eliminar un grupo de template solo elimina el grupo lógico, no los templates del grupo. No es posible eliminar un grupo de template que sea el único grupo de cualquier template existente.

Creación de subgrupos de plantillas

Un subgrupo de plantillas (o grupo de plantillas anidado) es un hijo del grupo de plantillas principal que lo contiene.

Un subgrupo se crea utilizando la barra inclinada '/' en el campo de entrada del nombre del grupo para denotar su relación con el grupo principal. Por ejemplo:

- al introducir `Servidores Linux/Bases de datos` se crea el subgrupo `Servidores Linux/Bases de datos` dentro del grupo principal `Servidores Linux`.
- al introducir `Servidores Linux/Bases de datos/MySQL/Tokio` se crea el subgrupo respectivo dentro de los grupos principales anidados `Servidores Linux`, `Servidores Linux/Bases de datos`, `Servidores Linux/Bases de datos/MySQL`.

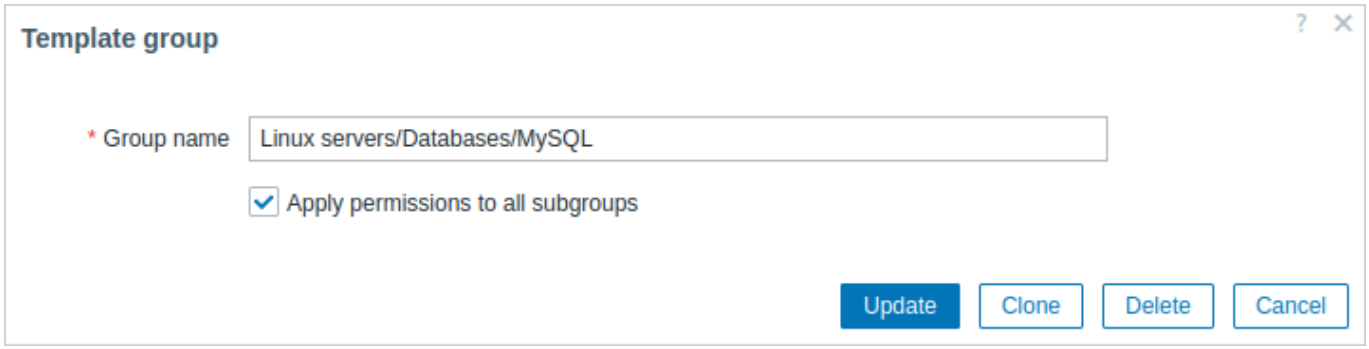
Al crear un subgrupo, no se permite utilizar barras inclinadas al principio o al final, ni varias barras inclinadas seguidas. No se admite el escape de '/'.

No es necesario crear ningún grupo de plantillas principal antes de crear un subgrupo. Puede elegir si desea comenzar creando un subgrupo (por ejemplo, `Servidores Linux/Bases de datos`) o cualquier grupo de plantillas principal (en nuestro ejemplo, `Servidores Linux`). Si comienza creando un subgrupo, los grupos de plantillas principales **no** se crearán automáticamente.

Permisos para grupos de template

- Al crear un subgrupo de un grupo de template principal existente (por ejemplo, al crear `Linux servers/Databases` cuando `Linux servers` ya existe), los permisos del grupo de usuarios para el subgrupo se heredan del grupo principal.
- Al crear un grupo de template principal para un subgrupo existente (por ejemplo, al crear `Linux servers` cuando `Linux servers/Databases` ya existe), no se establecen permisos para el grupo principal.

Al editar cualquier grupo de template, también puede configurar una opción adicional, *Aplicar permisos a todos los subgrupos*.



Al marcar esta casilla y hacer clic en *Update*, se aplicará el mismo nivel de permisos a todos los subgrupos actuales y futuros del grupo de template que se está editando.

Por lo tanto, si a algún grupo de usuarios se le han asignado permisos diferentes para los subgrupos del grupo de template que se está editando, al marcar la casilla todos los subgrupos actuales y futuros obtendrán los mismos permisos de usuario que el grupo que se está editando.

Tenga en cuenta que esta opción no se guarda en la base de datos y sobrescribirá los permisos existentes. Cualquier cambio realizado mediante esta opción solo puede revertirse manualmente.

Vinculación/desvinculación

Descripción general

La vinculación es un proceso mediante el cual las plantillas se aplican a los equipos, mientras que la desvinculación elimina la asociación con la plantilla de un equipo.

Vincular un template

Para vincular un template al host, haga lo siguiente:

1. Vaya a *Recopilación de datos* > *Hosts*.
2. Haga clic en el host requerido.
3. Empiece a escribir el nombre del template en el campo *Templates*. Aparecerá una lista de templates coincidentes; desplácese hacia abajo para seleccionar uno. Como alternativa, puede hacer clic en *Seleccionar* junto al campo y seleccionar uno o varios templates de la lista en una ventana emergente.
4. Haga clic en *Añadir/Actualizar* en el formulario de atributos del host.

El host tendrá ahora todas las entidades del template. Esto incluye items, triggers, gráficos, reglas de descubrimiento de bajo nivel, escenarios web y dashboards.

Attention:
La vinculación de varios templates al mismo host fallará si esos templates contienen items con la misma clave de item. Además, como los triggers y los gráficos utilizan items, tampoco se pueden vincular a un único host desde varios templates si utilizan claves de item idénticas.

Cuando se añaden entidades (items, triggers, etc.) desde el template:

- las entidades idénticas que ya existían en el host se actualizan como entidades del template, y **se pierden todas las personalizaciones existentes de nivel de host realizadas en la entidad**;
- se añaden las entidades del template;
- las entidades vinculadas directamente que, antes de vincular el template, solo existían en el host permanecen sin cambios.

En las listas, todas las entidades del template aparecen ahora precedidas por el nombre del template, lo que indica que pertenecen a ese template en particular. El propio nombre del template (mostrado en gris) es un enlace que permite acceder a la lista de esas entidades en el nivel del template.

Note:
Para algunos items, como las comprobaciones externas, las comprobaciones de HTTP agent, las comprobaciones simples, las comprobaciones de SSH y las comprobaciones de Telnet, una interfaz de host es opcional. Si, en el momento de vincular un template, el host no tiene una interfaz definida, estos items se añadirán sin una interfaz. Si añade una interfaz de host posteriormente, no se asignará automáticamente a los items ya existentes. Para asignar de una vez la interfaz de host recién añadida a todos los items del template, desvincule el template del host y vuelva a vincularlo. Para conservar el historial de los items, utilice la opción *Desvincular*; no utilice *Desvincular y borrar*.

Si alguna entidad no aparece precedida por el nombre del template, significa que ya existía en el host y que no fue añadida por el template.

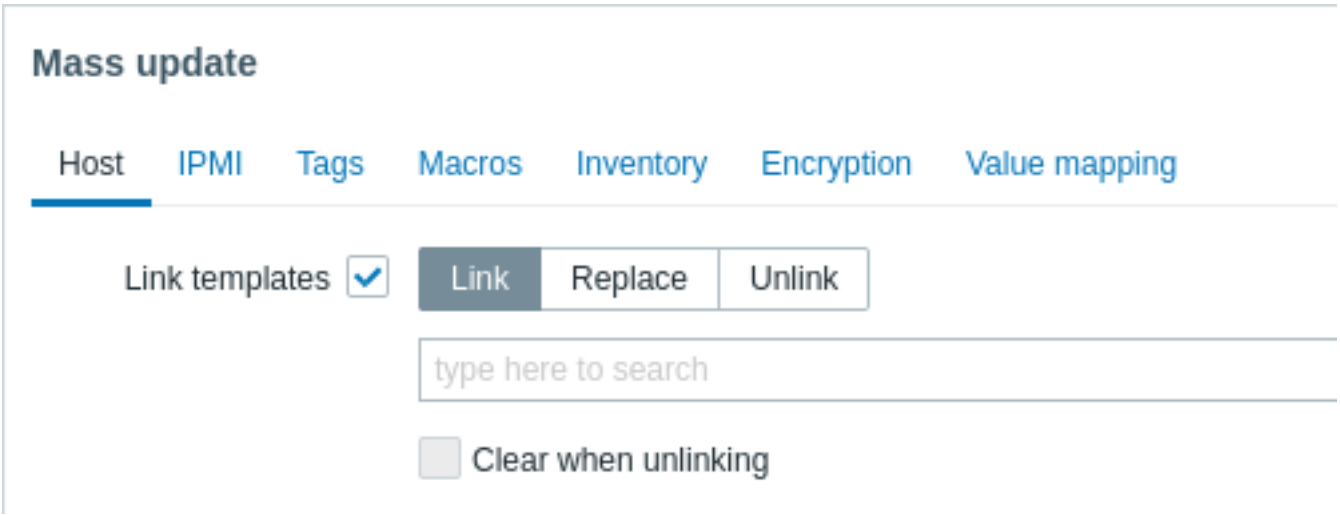
Criterios de unicidad de entidades

Al añadir entidades (métricas, disparadores, etc.) desde una plantilla, es importante saber cuáles de esas entidades ya existen en el equipo y necesitan ser actualizadas y cuáles entidades difieren. Los criterios de unicidad para decidir sobre la igualdad/diferencia son:

- para métricas: la clave de la métrica;
- para disparadores: el nombre del disparador y la expresión;
- para gráficos personalizados: el nombre del gráfico y sus métricas.

Vinculación de templates a varios hosts

Para actualizar la vinculación de templates de muchos hosts, en *Recopilación de datos* > *Hosts*, seleccione algunos hosts marcando sus casillas de verificación, haga clic en **Actualización masiva** debajo de la lista y, a continuación, seleccione *Vincular templates*:



Para vincular templates adicionales, comience a escribir el nombre del template en el campo de autocompletado hasta que aparezca una lista desplegable con los templates coincidentes. Desplácese hacia abajo para seleccionar el template que desea vincular.

La opción *Reemplazar* permite vincular un nuevo template y desvincular cualquier template que estuviera vinculado anteriormente a los hosts. La opción *Desvincular* permite especificar qué templates se deben desvincular. La opción *Limpiar al desvincular* permite no solo desvincular los templates vinculados anteriormente, sino también eliminar todas las entidades heredadas de ellos (items, triggers, etc.).

Note:
Zabbix ofrece un conjunto considerable de templates predefinidos. Puede utilizarlos como referencia, pero tenga cuidado al usarlos sin modificaciones en producción, ya que pueden contener demasiados items y consultar datos con demasiada frecuencia. Si desea utilizarlos, ajústelos para adaptarlos a sus necesidades reales.

Editando entidades vinculadas

Si intenta editar una métrica o un disparador que fue vinculado desde la plantilla, puede darse cuenta de que muchas opciones clave están deshabilitadas para la edición. Esto tiene sentido, ya que la idea de las plantillas es que las cosas se editen de manera centralizada a nivel de plantilla. Sin embargo, aún puede, por ejemplo, habilitar/deshabilitar una métrica en equipos individuales y establecer el intervalo de actualización, la longitud del historial y algunos otros parámetros.

Si desea editar completamente la entidad, debe hacerlo a nivel de plantilla (el acceso directo a nivel de plantilla se muestra en el nombre del formulario), teniendo en cuenta que estos cambios afectarán a todos los equipos que tengan esta plantilla vinculada.

Attention:
Cualquier personalización de las entidades implementada a nivel de plantilla sobrescribirá las personalizaciones anteriores de las entidades a nivel de equipo.

Desvincular un template

Para desvincular un template de un host, haga lo siguiente:

1. Vaya a *Data collection* > *Hosts*.
2. Haga clic en el host requerido y busque el campo *Templates*.
3. Haga clic en *Unlink* o *Unlink and clear* junto al template para desvincularlo.
4. Haga clic en *Update* en el formulario de atributos del host.

Al elegir la opción *Unlink*, solo se eliminará la asociación con el template, mientras que todas sus entidades permanecerán en el host. Esto incluye items, triggers, gráficos, reglas de descubrimiento de bajo nivel y escenarios web, pero excluye los dashboards. Tenga en cuenta que los value maps y las etiquetas heredadas de los templates vinculados también se eliminarán.

Al elegir la opción *Unlink and clear*, se eliminarán tanto la asociación con el template como todas sus entidades (items, triggers, etc.).

Anidamiento

Descripción general

La anidación es una forma de que una plantilla abarque una o más plantillas adicionales.

Como tiene sentido separar entidades en plantillas individuales para varios servicios, aplicaciones, etc., puede terminar con bastantes plantillas, todas las cuales pueden necesitar estar vinculadas a varios equipos. Para simplificar la situación, es posible vincular algunas plantillas entre sí en una sola plantilla.

El beneficio de la anidación es que solo tiene que vincular una plantilla al equipo, y el equipo heredará automáticamente todas las entidades de las plantillas que estén vinculadas a esa plantilla. Por ejemplo, si vinculamos *T1* y *T2* a *T3*, complementamos *T3* con todas las entidades de *T1* y *T2*, pero no al revés. Si vinculamos *T1* a *T2* y *T3*, complementamos *T2* y *T3* con las entidades de *T1*.

Configuración de templates anidados

Para vincular templates, debe tomar un template existente (o crear uno nuevo) y luego:

1. Abra el formulario de configuración del template.
2. Busque el campo *Templates*.
3. Haga clic en *Select* para abrir la ventana emergente *Templates*.
4. En la ventana emergente, elija los templates necesarios y luego haga clic en *Select* para agregar los templates a la lista.
5. Haga clic en *Add* o *Update* en el formulario de configuración del template.

De este modo, todas las entidades del template configurado, así como todas las entidades de los templates vinculados, aparecerán ahora en la configuración del template. Esto incluye items, triggers, gráficos, reglas de descubrimiento de bajo nivel y escenarios web, pero excluye los dashboards. Sin embargo, los dashboards de los templates vinculados, no obstante, serán heredados por los hosts.

Para desvincular cualquiera de los templates vinculados, haga clic en *Unlink* o *Unlink and clear* en el formulario de configuración del template y luego haga clic en *Update*.

La opción *Unlink* simplemente eliminará la asociación con el template vinculado, sin eliminar ninguna de sus entidades (items, triggers, etc.).

La opción *Unlink and clear* eliminará tanto la asociación con el template vinculado como todas sus entidades (items, triggers, etc.).

Actualización masiva

Descripción general

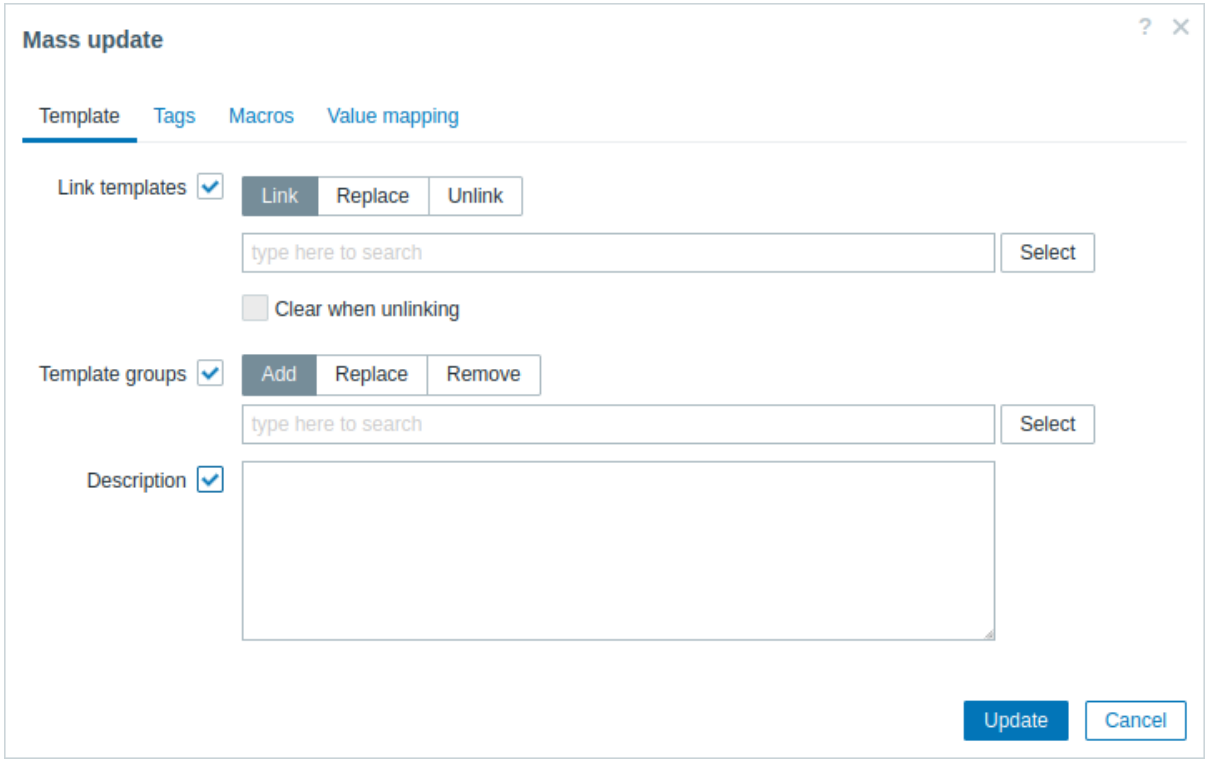
A veces puede que desee cambiar algún atributo para varias plantillas a la vez. En lugar de abrir cada plantilla individualmente para editarla, puede utilizar la función de actualización masiva para ello.

Uso de actualización masiva

Para actualizar masivamente algunas templates, haga lo siguiente:

1. Marque las casillas de verificación delante de las templates que desea actualizar en la lista de templates.
2. Haga clic en *Mass update* debajo de la lista.
3. Vaya a la pestaña con los atributos necesarios (*Template*, *Tags*, *Macros* o *Value mapping*).
4. Marque las casillas de verificación de cualquier atributo que desee actualizar e introduzca un nuevo valor para ellos.

La pestaña **Template** contiene opciones generales para la actualización masiva de templates.



Las siguientes opciones están disponibles al seleccionar el botón correspondiente para la actualización de *Vincular templates*:

- *Vincular* - especifique qué templates adicionales desea vincular;
- *Reemplazar* - especifique qué templates desea vincular y, al mismo tiempo, desvincule cualquier template vinculado anteriormente;
- *Desvincular* - especifique qué templates desea desvincular.

Para especificar los templates que desea vincular o desvincular, comience a escribir el nombre del template en el campo de autocompletado hasta que aparezca una lista desplegable con los templates coincidentes. Desplácese hacia abajo para seleccionar los templates necesarios.

La opción *Limpiar al desvincular* permite no solo desvincular los templates vinculados anteriormente, sino también eliminar todas las entidades heredadas de ellos (items, triggers, gráficos, etc.).

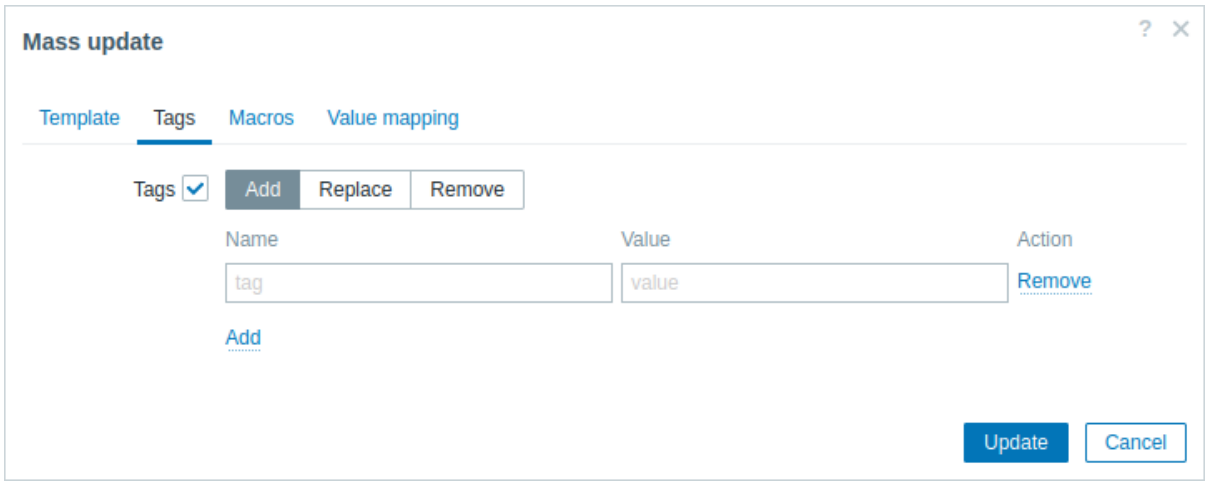
Las siguientes opciones están disponibles al seleccionar el botón correspondiente para la actualización de *Grupos de templates*:

- *Añadir* - especifique grupos de templates adicionales de los existentes o introduzca grupos de templates completamente nuevos para los templates;

- *Reemplazar* - elimine el template de todos los grupos de templates existentes y reemplácelos por los especificados en este campo (grupos de templates existentes o nuevos);
- *Eliminar* - elimine grupos de templates específicos de los templates.

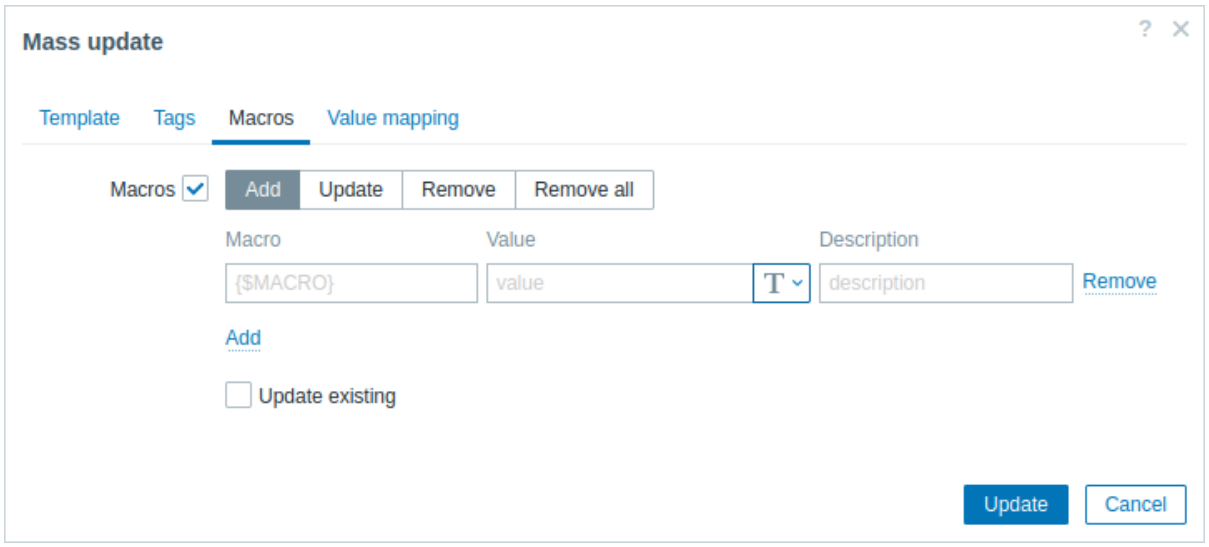
Estos campos utilizan autocompletado: al comenzar a escribir en ellos, aparece una lista desplegable con los grupos de templates coincidentes. Si el grupo de templates es nuevo, también aparece en la lista desplegable y se indica mediante *(nuevo)* después de la cadena. Desplácese hacia abajo para seleccionarlo.

La pestaña **Etiquetas** permite actualizar masivamente las etiquetas a nivel de plantilla.



En las etiquetas se admiten macros de usuario, macros {INVENTORY.*}, {HOST.HOST}, {HOST.NAME}, {HOST.CONN}, {HOST.DNS}, {HOST.IP}, {HOST.PORT} y {HOST.ID}. Tenga en cuenta que las etiquetas con el mismo nombre, pero con valores diferentes, no se consideran 'duplicadas' y pueden añadirse a la misma plantilla.

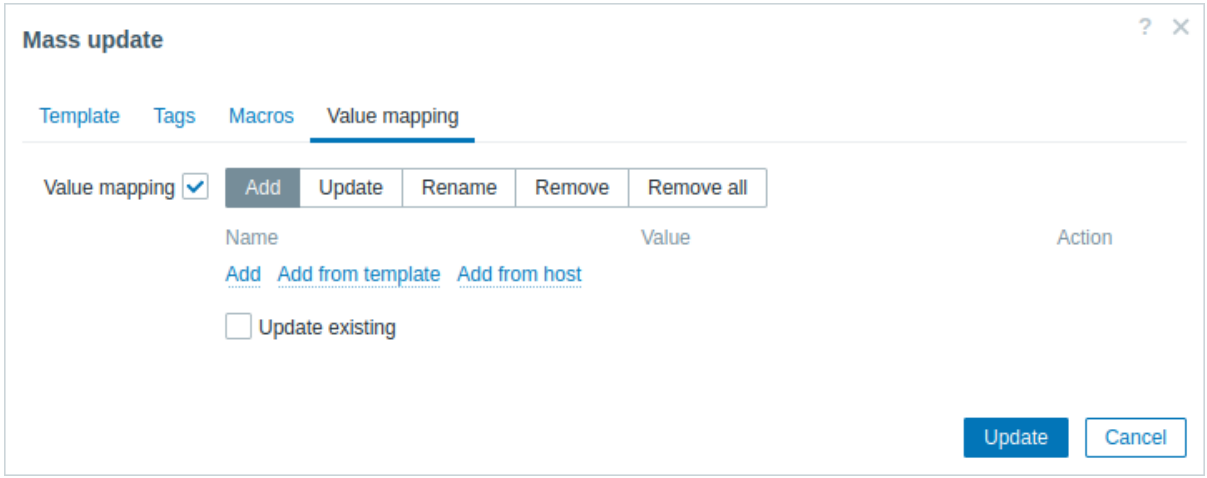
La pestaña **Macros** permite actualizar masivamente las macros a nivel de template.



Las siguientes opciones están disponibles al seleccionar el botón correspondiente para actualizar las macros:

- *Añadir*: especificar macros de usuario adicionales para los templates. Si la casilla *Actualizar existentes* está seleccionada, se actualizarán el valor, el tipo y la descripción del nombre de macro especificado. Si no está seleccionada, si ya existe una macro con ese nombre en el template o templates, no se actualizará.
- *Actualizar*: reemplazar los valores, tipos y descripciones de las macros especificadas en esta lista. Si la casilla *Añadir faltantes* está seleccionada, la macro que no existía previamente en un template se añadirá como una macro nueva. Si no está seleccionada, solo se actualizarán las macros que ya existen en un template.
- *Eliminar*: eliminar las macros especificadas de los templates. Si la casilla *Excepto las seleccionadas* está seleccionada, se eliminarán todas las macros excepto las especificadas en la lista. Si no está seleccionada, solo se eliminarán las macros especificadas en la lista.
- *Eliminar todas*: eliminar todas las macros de usuario de los templates. Si la casilla *Confirmo que deseo eliminar todas las macros* no está seleccionada, se abrirá una nueva ventana emergente para solicitar la confirmación de la eliminación de todas las macros.

La pestaña **Asignación de valores** permite actualizar masivamente las asignaciones de valores.



Para actualizar las asignaciones de valores, están disponibles botones con las siguientes opciones:

- *Añadir*: añade asignaciones de valores a los templates. Si marca *Actualizar existentes*, se actualizarán todas las propiedades de la asignación de valores con este nombre. De lo contrario, si ya existe una asignación de valores con ese nombre, no se actualizará.
- *Actualizar*: actualiza las asignaciones de valores existentes. Si marca *Añadir faltantes*, una asignación de valores que no existía anteriormente en un template se añadirá como una nueva asignación de valores. De lo contrario, solo se actualizarán las asignaciones de valores que ya existen en un template.
- *Renombrar*: asigna un nuevo nombre a una asignación de valores existente.
- *Eliminar*: elimina las asignaciones de valores especificadas de los templates. Si marca *Excepto las seleccionadas*, se eliminarán todas las asignaciones de valores **excepto** las especificadas.
- *Eliminar todas*: elimina todas las asignaciones de valores de los templates. Si la casilla *Confirmo eliminar todas las asignaciones de valores* no está marcada, se abrirá una nueva ventana emergente para solicitar confirmación de la eliminación.

Las opciones *Añadir desde template* y *Añadir desde host* están disponibles para las operaciones de añadir/actualizar asignaciones de valores. Permiten seleccionar asignaciones de valores de un template o de un host, respectivamente.

Cuando haya terminado con todos los cambios necesarios, haga clic en *Update*. Los atributos se actualizarán en consecuencia para todas las templates seleccionadas.

Hosts y grupos de hosts

¿Qué es un ”equipo”?

En Zabbix, un ”equipo” se refiere a cualquier dispositivo físico o virtual, aplicación, servicio o cualquier otra colección lógicamente relacionada de parámetros monitorizados.

Crear equipos es una de las primeras tareas de monitorización en Zabbix. Por ejemplo, si desea monitorizar algunos parámetros en un servidor ”x”, primero debe crear un equipo llamado, por ejemplo, ”Servidor X” y luego podrá añadirle métricas de monitorización.

Los equipos se organizan en grupos de equipos.

Proceda a:

- Asistente de equipos para una interfaz guiada, paso a paso, para crear y configurar un equipo nuevo o existente.
- Configurar un equipo para un enfoque clásico de creación y configuración de un equipo.

Asistente de host

Descripción general

El Host Wizard es una interfaz guiada, paso a paso, para configurar su objetivo de monitorización (dispositivo, aplicación, servicio, etc.) en Zabbix. Le guía a través de los siguientes pasos:

- Seleccionar un template
- Crear o seleccionar un host
- Instalar Zabbix agent o agent 2
- Añadir una interfaz de host
- Aplicar configuración adicional a su objetivo de monitorización o al host de Zabbix (si lo requiere el template)

Para acceder al Host Wizard en el frontend de Zabbix, haga lo siguiente:

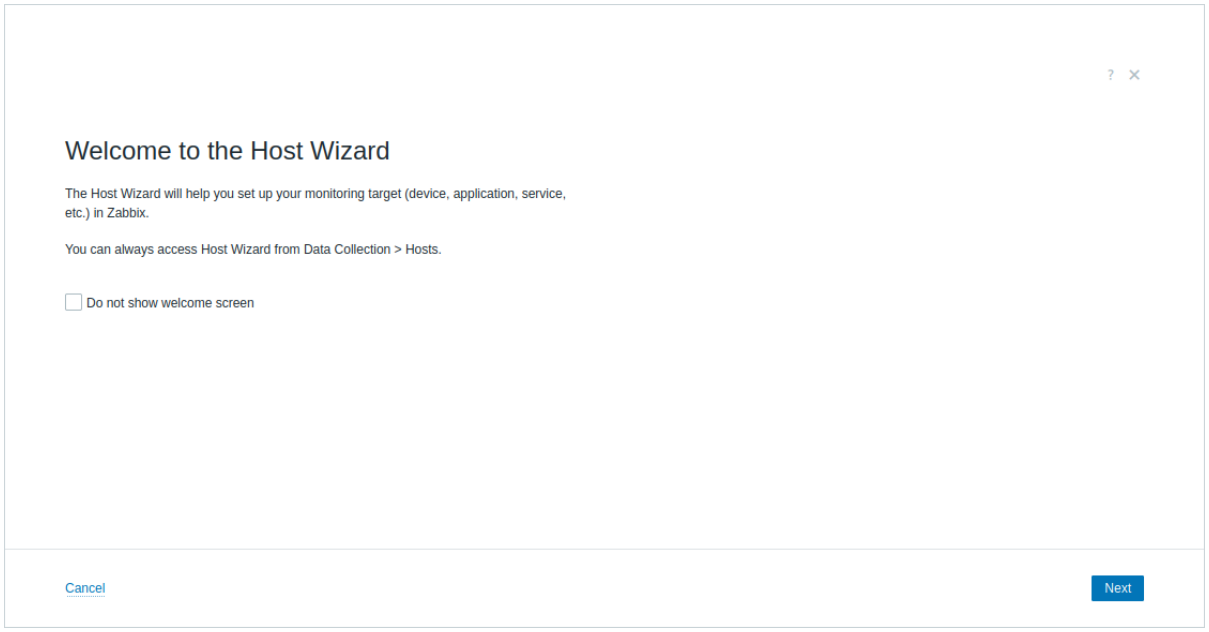
1. Vaya a: *Data collection* > *Hosts*.
2. Haga clic en *Host Wizard* en la esquina superior derecha de la pantalla.
3. Siga las instrucciones del asistente.

También puede editar un host existente haciendo clic en su nombre (o en el icono de tres puntos junto a él) en varias secciones del frontend para acceder al Host Wizard desde el menú del host. Tenga en cuenta que algunos pasos se omiten al editar un host existente si ya incluye la configuración requerida por el template seleccionado.

Como alternativa, puede usar el método clásico para crear y configurar un host.

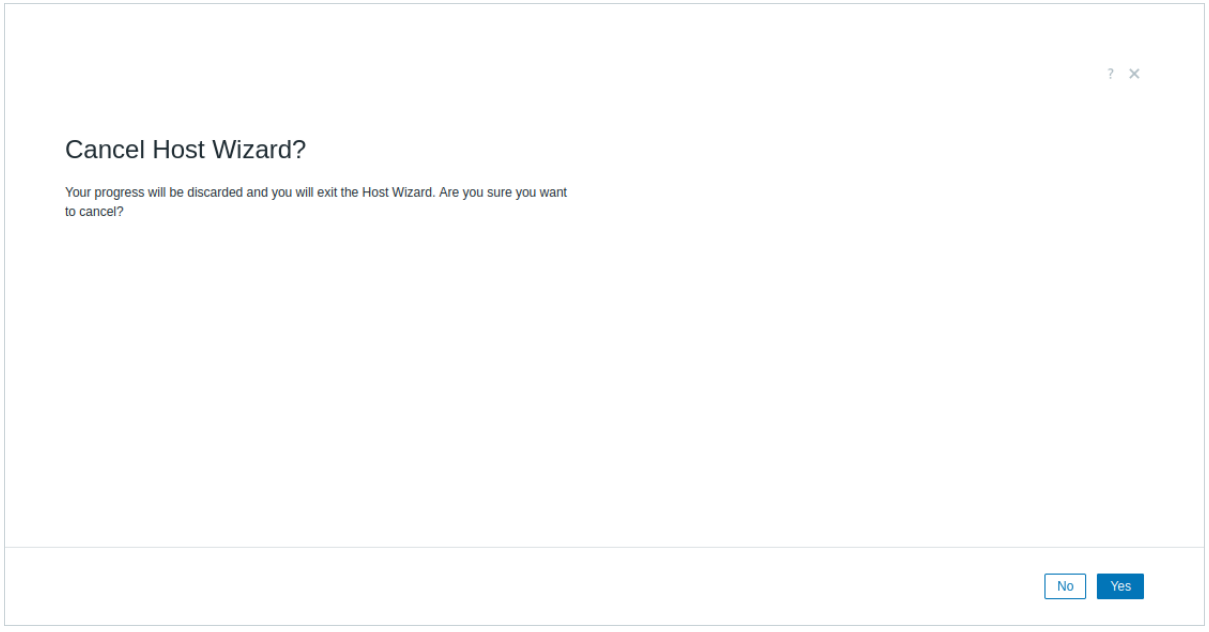
Bienvenido al asistente de host

Cuando inicia el asistente de host, se muestra una pantalla de bienvenida:



Para omitir esta pantalla en sesiones futuras, puede marcar la casilla *No mostrar la pantalla de bienvenida* y hacer clic en *Siguiente*.

Si empieza a configurar los ajustes en los siguientes pasos del asistente de host y trata de cerrar el asistente haciendo clic en el icono de cierre o presionando Esc, aparecerá una pantalla de confirmación:



Al hacer clic en *Sí*, se saldrá del asistente de host sin guardar el progreso. Al hacer clic en *No*, Esc o el icono de cierre, se volverá al último paso.

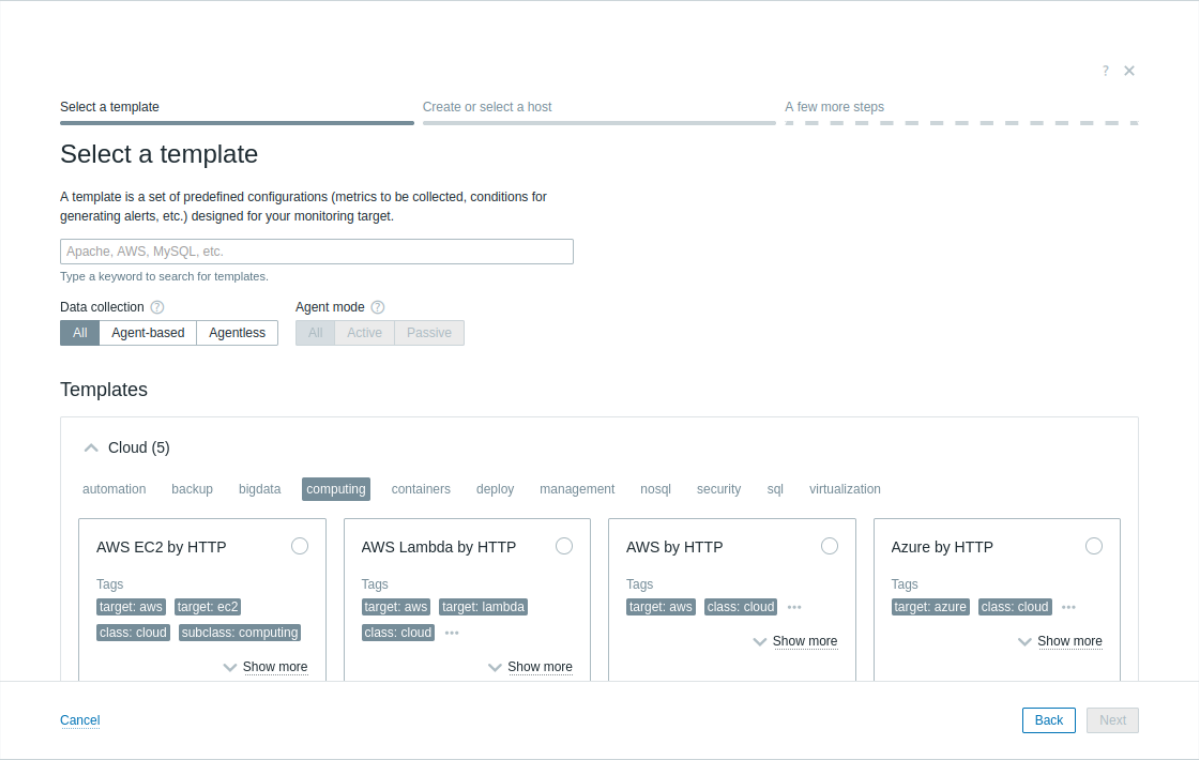
Seleccionar un template

El primer paso para configurar el objetivo de monitorización es seleccionar un template, que es un conjunto de configuraciones predefinidas (métricas que se recopilarán, condiciones para generar alertas, etc.) diseñado para el objetivo de monitorización.

Los templates incluidos de fábrica en Zabbix ofrecen una variedad de configuraciones de monitorización predefinidas para sistemas operativos, aplicaciones, bases de datos, dispositivos de red, servicios y mucho más.

En este paso, puede:

- Explorar los templates por clase (Cloud, Database, Network, etc.) y subclase (por ejemplo, automatización, copias de seguridad), que se basan en las etiquetas de los templates.
- Buscar templates por palabras clave en el nombre del template o en los nombres y valores de sus etiquetas.
- Filtrar templates por método de recopilación de datos (basado en agent o sin agent; los templates basados en agent incluyen al menos un item de Zabbix agent).
- Filtrar templates por modo de agent (activo o pasivo; los templates activos incluyen al menos un item de Zabbix agent (activo), mientras que los templates pasivos incluyen al menos un item pasivo; para obtener más información, consulte Comprobaciones de agent pasivas y activas).
- Al utilizar el Asistente de hosts para configurar un host existente, también puede filtrar los templates que ya están vinculados al host.



Attention:
Al seleccionar un template, puede aparecer el siguiente mensaje: *Algunos templates (n) no son compatibles con el Asistente de hosts. Consulte cómo actualizarlos. Los templates personalizados no son compatibles.* Esto indica que algunos templates incluidos de fábrica aún no son compatibles con el Asistente de hosts y deben actualizarse; consulte las instrucciones de actualización de templates.

El template que seleccione determinará los pasos siguientes del Asistente de hosts. Por ejemplo, si selecciona el template *MySQL by Zabbix agent*, que utiliza Zabbix agent para la recopilación de datos, el asistente le guiará durante el proceso de instalación del agent.

Crear o seleccionar un host

Un template debe vincularse a un host, que es una entidad de Zabbix que representa el objetivo de monitorización. Al vincularlo, el host recibe todas las entidades del template, como los items (métricas que se recopilarán) y los triggers (condiciones para generar alertas), necesarias para la monitorización.

Cada host también debe pertenecer al menos a un grupo de hosts, que se utiliza para organizar los hosts y asignarles permisos de usuario.

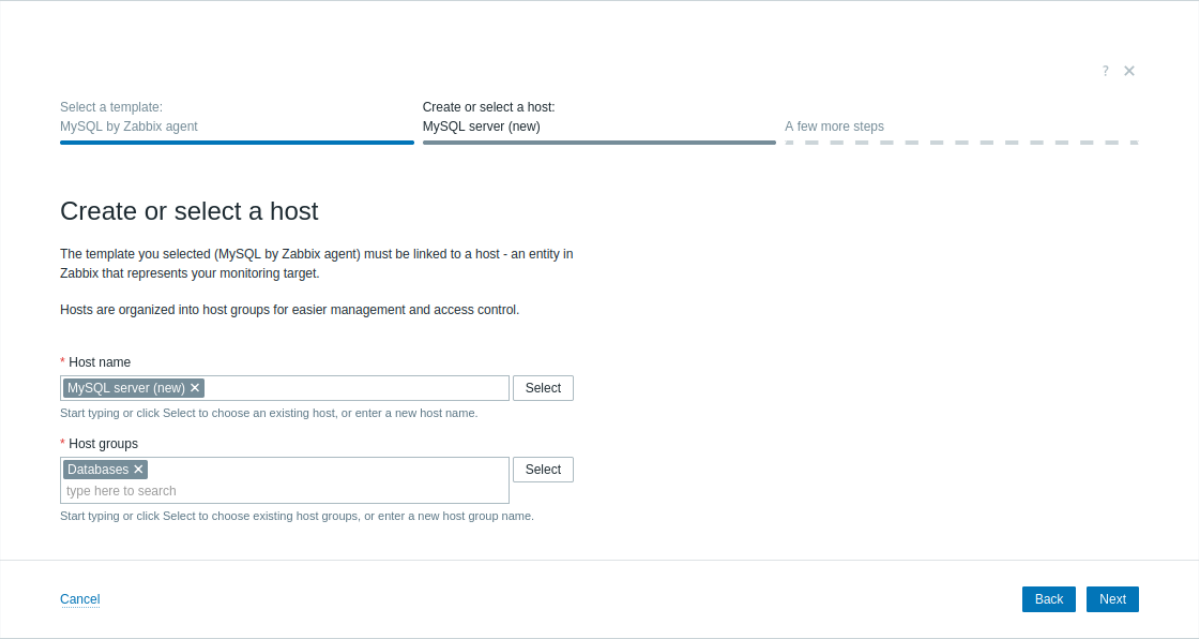
En este paso, puede:

- Crear un host y un grupo de hosts nuevos introduciendo un nombre para cada uno.
- Crear un host nuevo introduciendo su nombre y seleccionar un grupo de hosts existente al que se asignará el nuevo host.

- Seleccionar un host existente; también puede asignar este host a grupos de hosts adicionales sin eliminarlo de ninguno de sus grupos actuales.

Note:
Seleccionar un host existente no elimina ni sobrescribe su configuración actual, a menos que se indique explícitamente en pasos posteriores. Los hosts descubiertos no se pueden seleccionar.

Por ejemplo, podría crear un host nuevo llamado *MySQL server* para representar el server MySQL instalado localmente que desea monitorizar. El template *MySQL by Zabbix agent* seleccionado anteriormente se vinculará a este host. Además, podría seleccionar el grupo de hosts existente *Databases* (o crear uno nuevo, por ejemplo, *MySQL servers*) para organizar el host junto con otros hosts relacionados con bases de datos y simplificar posteriormente la gestión de permisos.



Instalar Zabbix agent

Las plantillas basadas en agent (como *MySQL by Zabbix agent*) requieren instalar Zabbix agent o agent 2: un proceso implementado en el destino de monitorización para supervisar activamente los recursos y las aplicaciones locales (si la plantilla lo requiere).

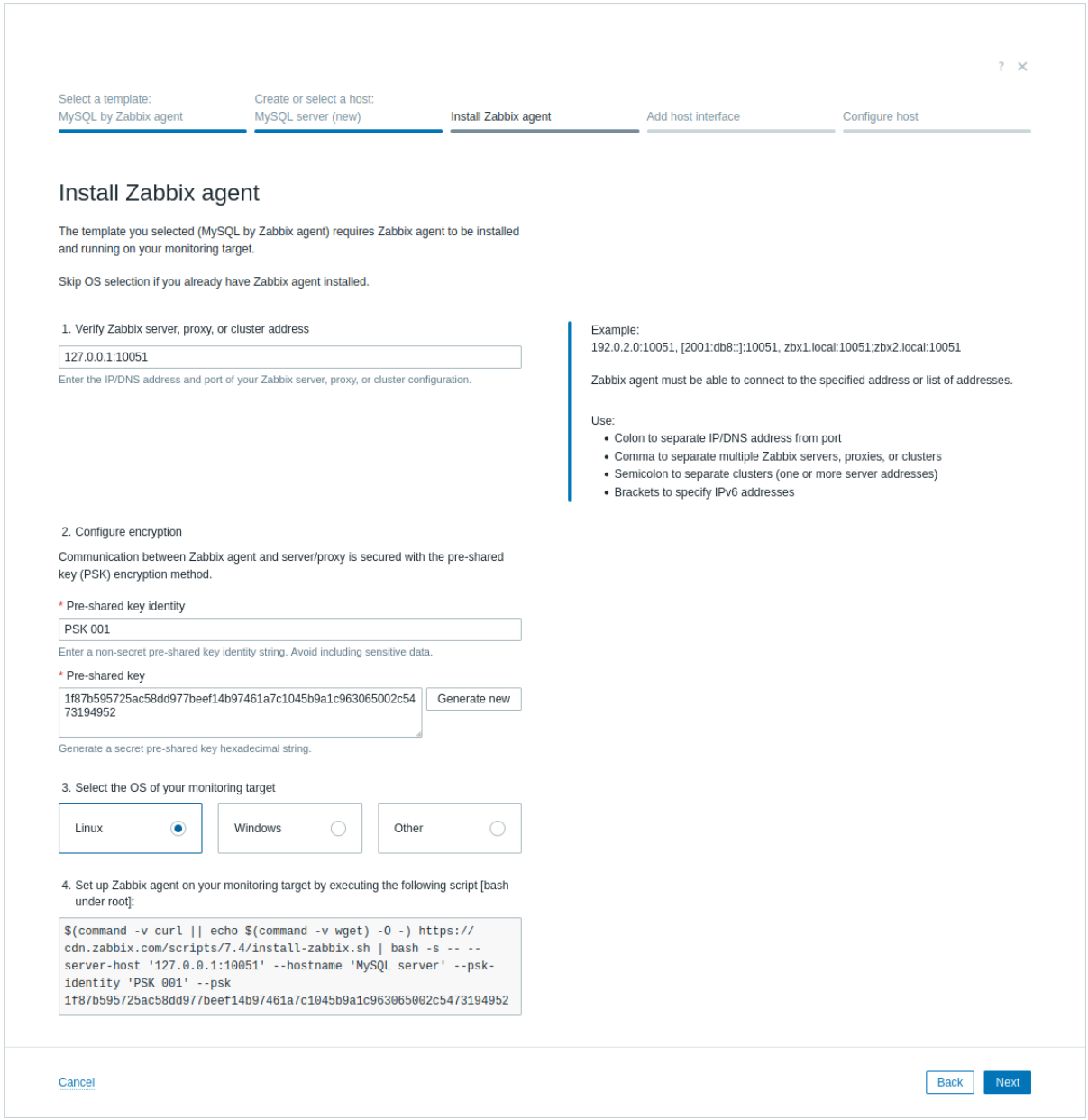
En este paso:

- Verifique la dirección del Zabbix server, proxy o clúster (por ejemplo, 192.0.2.1:10051). Este valor se utilizará para configurar los parámetros Server y ServerActive de Zabbix agent.
- Introduzca una identidad de clave precompartida nueva, única y no secreta (por ejemplo, *PSK 001* o *mysql-agent-psk1*) y genere la clave precompartida. Estos valores se utilizarán para configurar el cifrado mediante clave precompartida (PSK) para el host de Zabbix (por ejemplo, *MySQL server*) y Zabbix agent.

Attention:
Al editar un host existente, la configuración de PSK sobrescribirá todos los ajustes de cifrado existentes en el host.

- Seleccione el sistema operativo de la máquina que aloja el destino de monitorización.
- Instale Zabbix agent en ese sistema ejecutando el script proporcionado en él o siguiendo las instrucciones de instalación.

Por ejemplo, podría establecer *Identidad de la clave precompartida* como *PSK 001* y generar una nueva *Clave precompartida*. A continuación, podría seleccionar *Linux* como sistema operativo y ejecutar el script proporcionado en ese sistema. Después de instalar Zabbix agent, vuelva al Asistente de hosts.

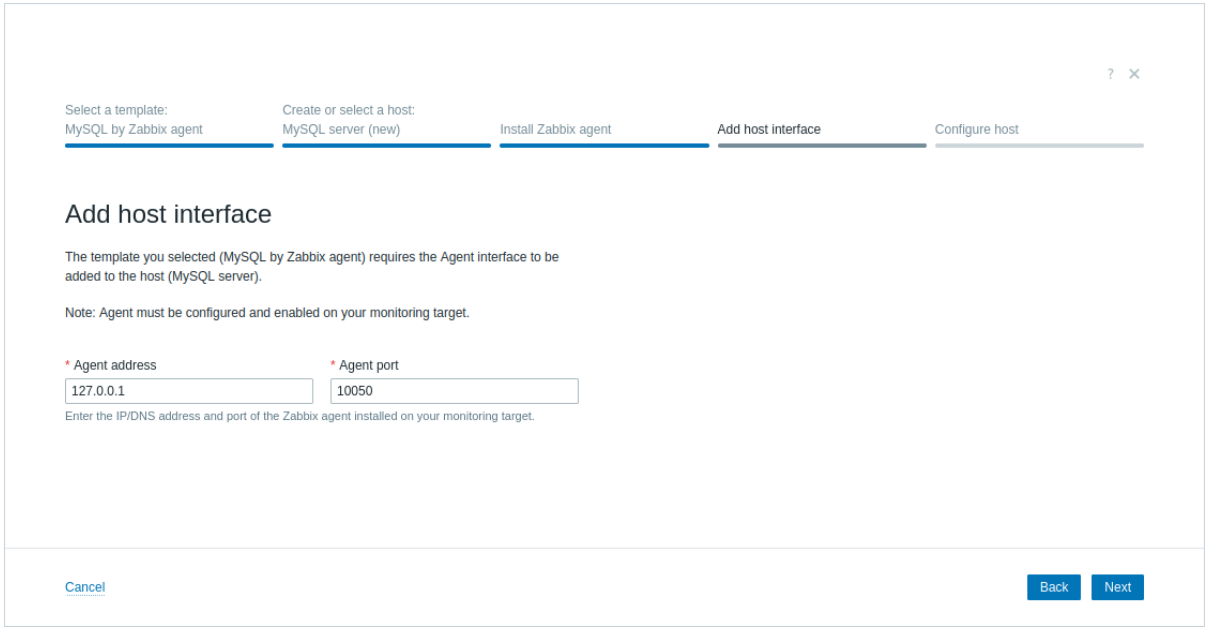


Agregar interfaz de host

La interfaz de host define cómo Zabbix server se conecta a su objetivo de monitoreo a través de la red. Define parámetros de conexión como la dirección IP, el nombre DNS, el puerto y el tipo de interfaz (Agent, SNMP, JMX o IPMI), según el método de recopilación de datos requerido por el template seleccionado.

En este paso, introduzca los detalles de la interfaz de host requeridos por el template seleccionado.

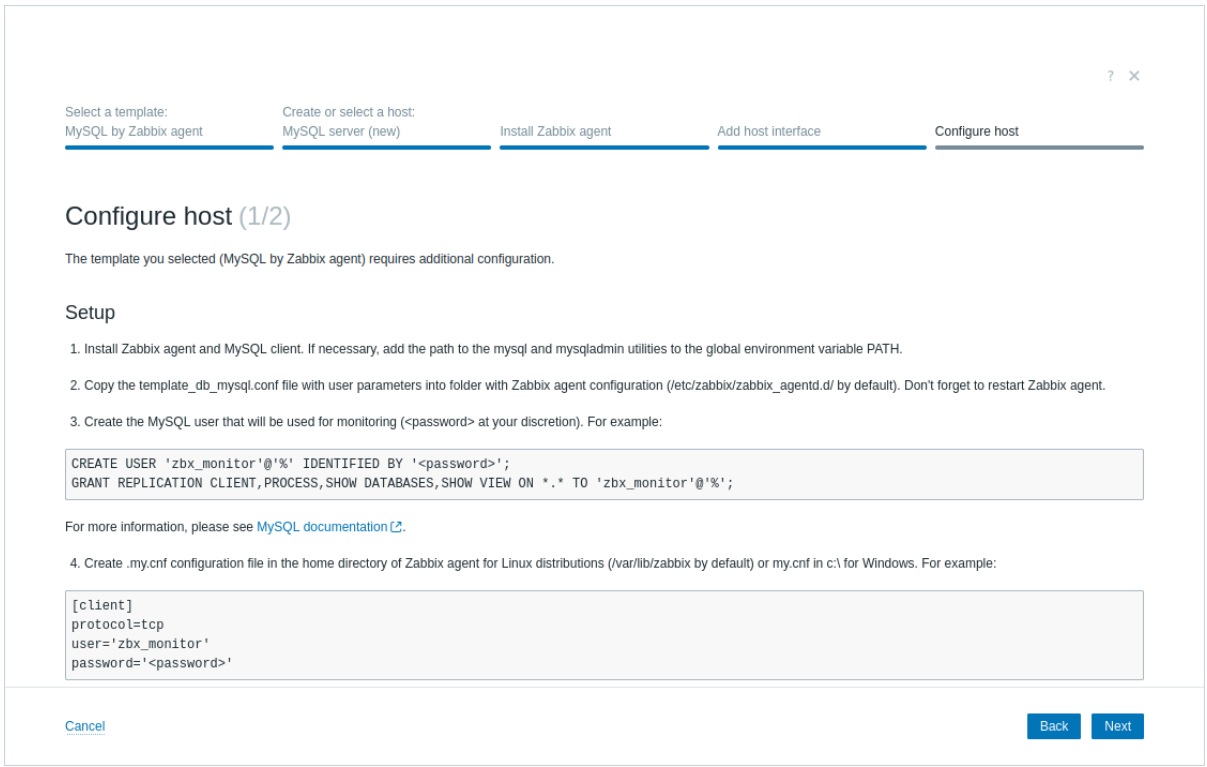
Por ejemplo, podría usar la dirección predeterminada de agent (`127.0.0.1`) y el puerto (10050) si Zabbix server, Zabbix agent y el servidor MySQL se ejecutan localmente en la misma máquina.



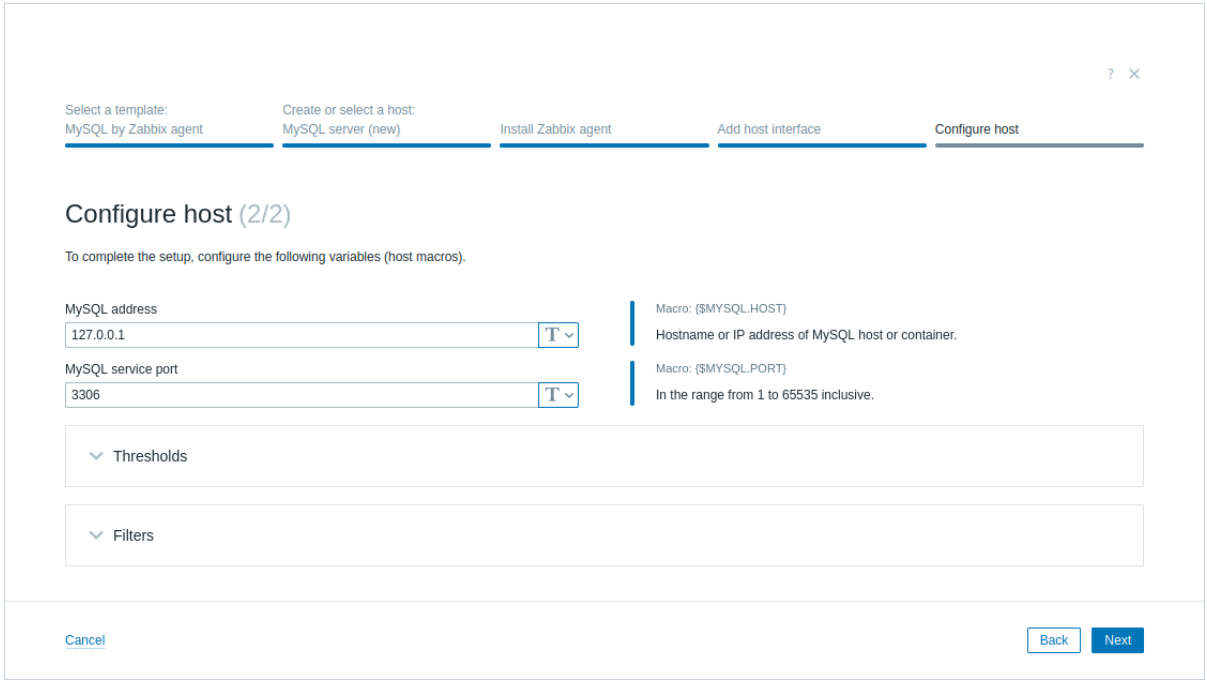
Configurar host

Algunos templates requieren configuración adicional antes de poder crear el host. Esto puede incluir:

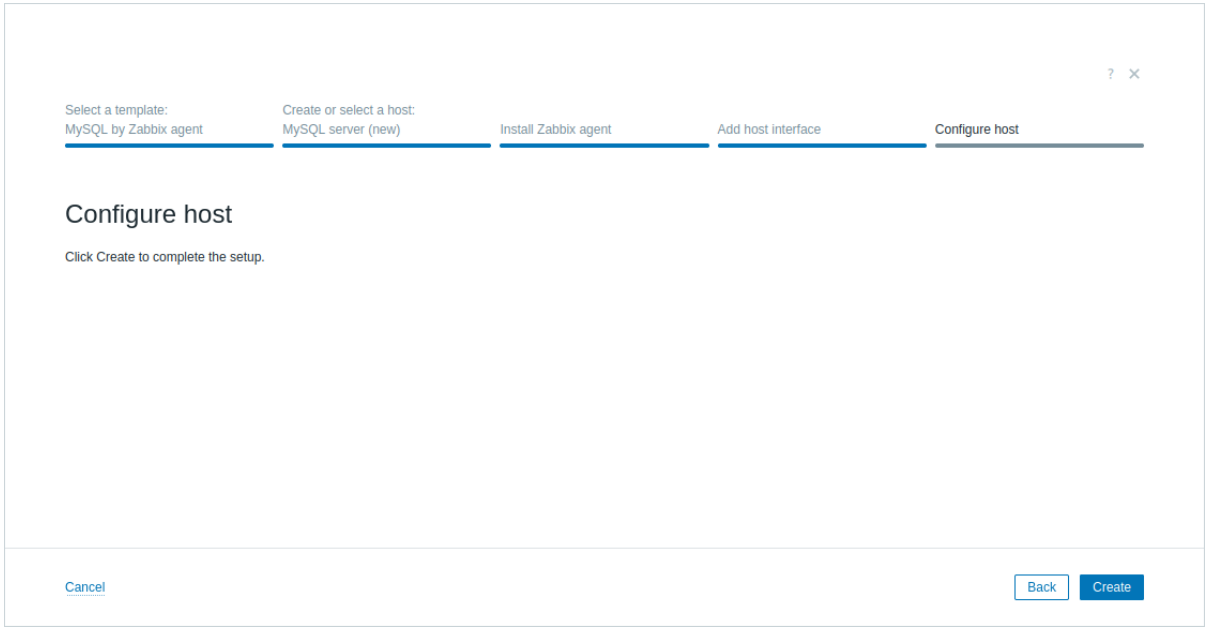
- Configuración manual del objetivo de monitorización (por ejemplo, habilitar servicios específicos, crear usuarios de servicio u otorgar permisos):



- Definir macros del host: variables que controlan el comportamiento de los item, la configuración de conexión, las credenciales de autenticación y otros parámetros:



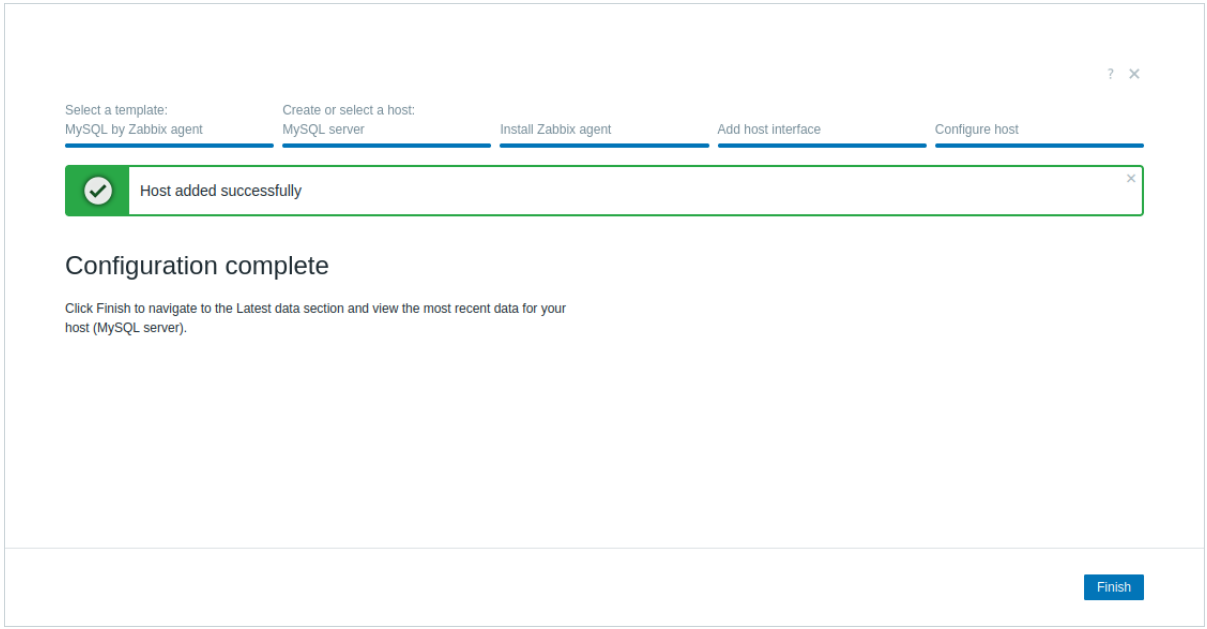
Después de completar todos los pasos de configuración necesarios, haga clic en *Crear* para añadir el host a Zabbix (o en *Actualizar* para actualizar la configuración de un host existente):



Configuración completa

En este punto, Zabbix ya está monitorizando su objetivo.

Para nuevos equipos, haga clic en *Finalizar* para navegar a la sección de Datos recientes y ver los datos más recientes de su equipo. Para equipos existentes, haga clic en *Finalizar* para cerrar el Asistente de equipos y volver a la pantalla desde la que se abrió.



Configuración de un host

Descripción general

Para configurar un host en Zabbix frontend, haga lo siguiente:

1. Vaya a *Data collection* > *Hosts* o *Monitoring* > *Hosts*.
2. Haga clic en *Create host* en la esquina superior derecha de la pantalla (o en el nombre del host para editar un host existente).
3. Introduzca los parámetros del host en el formulario.

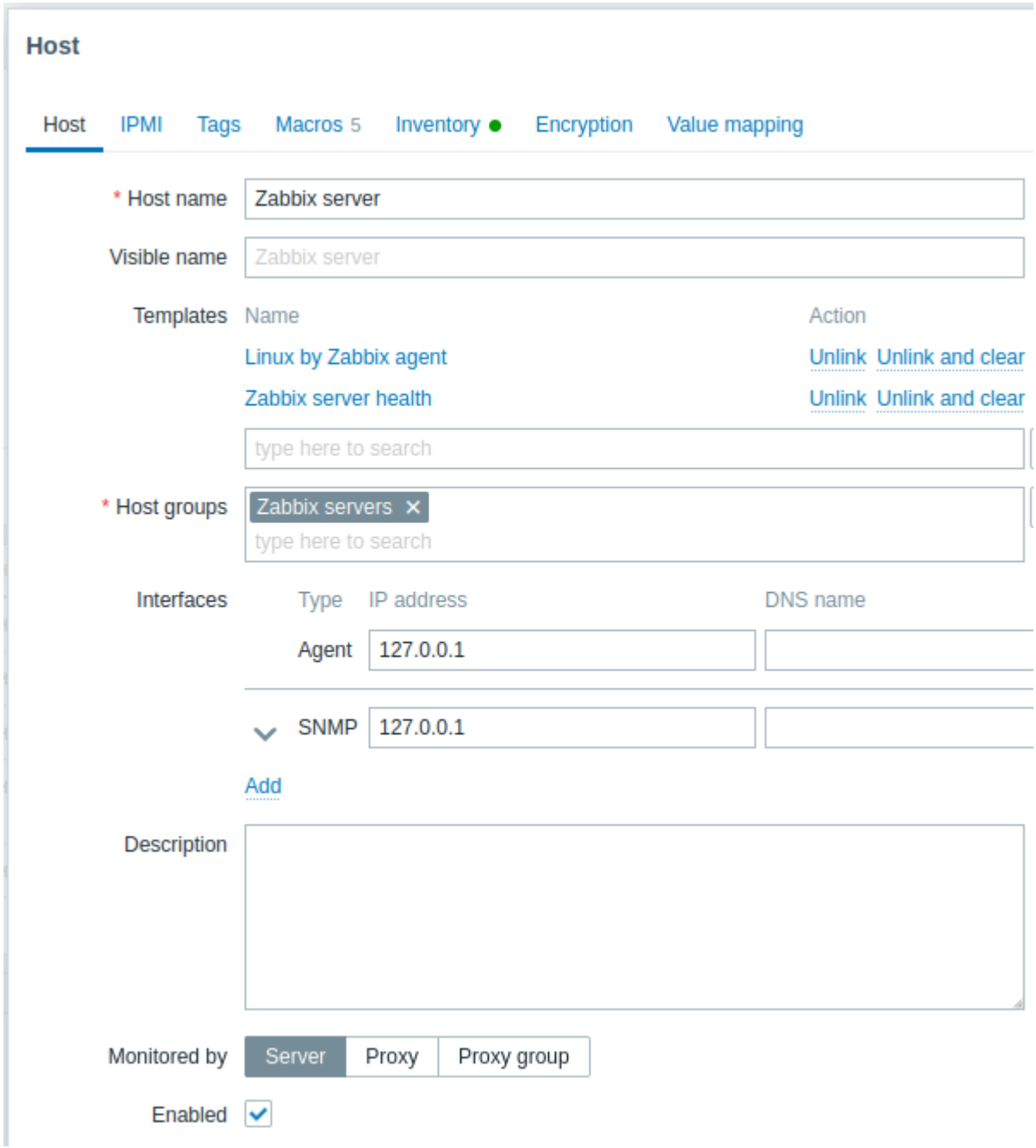
También puede usar el botón *Clone* en el formulario de configuración de un host existente para crear un nuevo host. Este host tendrá todas las propiedades del host existente, incluidas las templates vinculadas, las entidades (items, triggers, etc.) de esas templates, así como las entidades adjuntas directamente al host existente.

Tenga en cuenta que, cuando se clona un host, conservará todas las entidades de las templates tal como están originalmente en la template. Cualquier cambio realizado en esas entidades a nivel del host existente (como un intervalo de item modificado, una expresión regular modificada o prototipos añadidos a la regla de descubrimiento de bajo nivel) no se clonará al nuevo host; en su lugar, quedarán como en la template.

Como alternativa, puede usar el Host Wizard para configurar un host mediante una interfaz guiada paso a paso.

Configuración

La pestaña **Host** contiene los atributos generales del host:



Todos los campos de entrada obligatorios están marcados con un asterisco rojo.

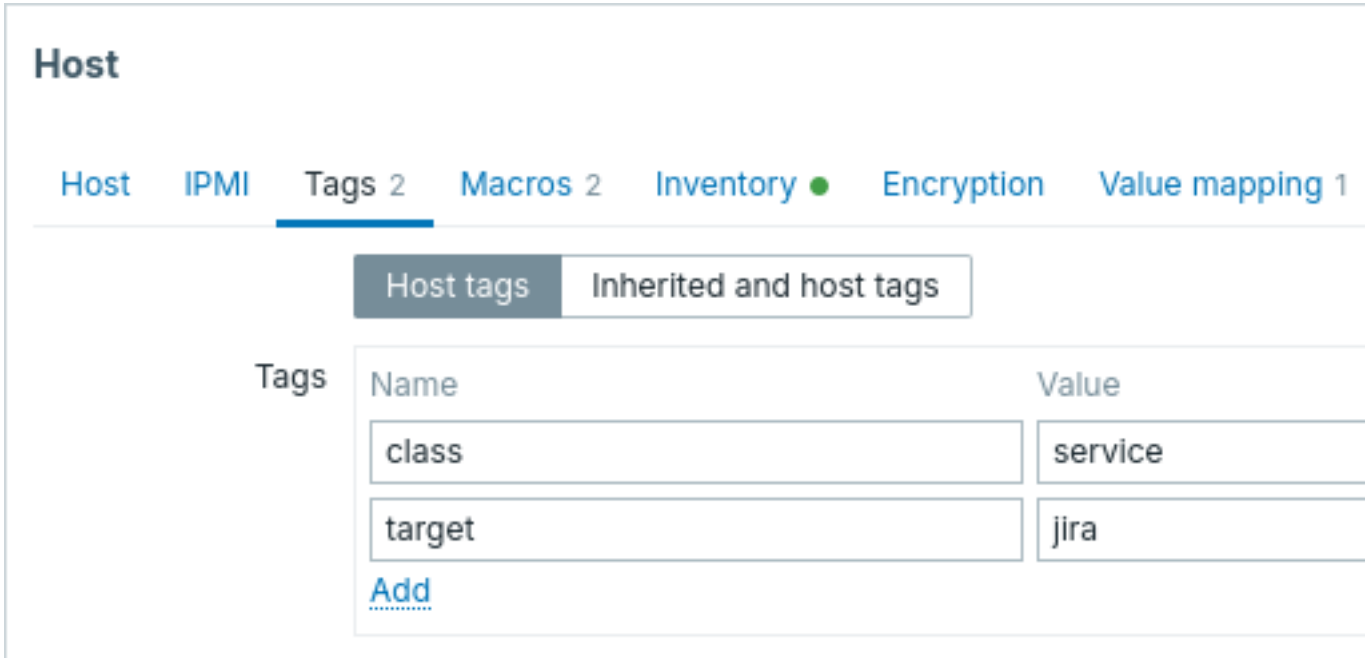
Parámetro	Descripción
Nombre del host	Introduzca un nombre de host único. Se permiten caracteres alfanuméricos, espacios, puntos, guiones y guiones bajos. Sin embargo, no se permiten espacios al principio ni al final. *Nota:* Si Zabbix agent se está ejecutando en el host que está configurando, el parámetro *Hostname* del archivo de configuración del agent debe tener el mismo valor que el nombre del host introducido aquí. El nombre especificado en el parámetro es necesario para procesar las comprobaciones activas.
Nombre visible	Introduzca un nombre visible único para el host. Si establece este nombre, será el que se muestre en listas, mapas, etc., en lugar del nombre técnico del host. Este atributo admite UTF-8.

Parámetro	Descripción
Templates	Vincule templates al host. Todas las entidades (items, triggers, etc.) se heredarán del template. Para vincular un template nuevo, comience a escribir su nombre en el campo de entrada de texto. Aparecerá una lista de templates coincidentes; desplácese hacia abajo para seleccionar uno. También puede hacer clic en *Seleccionar*, junto al campo, y seleccionar templates de la lista en una ventana emergente. Todos los templates seleccionados se vincularán al host cuando se guarde o actualice el formulario de configuración del host. Para desvincular un template, utilice una de las dos opciones del bloque *Templates vinculados*: *Desvincular*: desvincula el template, pero conserva sus entidades (items, triggers, etc.); *Desvincular y borrar*: desvincula el template y elimina todas sus entidades (items, triggers, etc.). Los nombres de los templates mostrados son enlaces en los que se puede hacer clic y que llevan al formulario de configuración del template.
Grupos de hosts	Seleccione los grupos de hosts a los que pertenece el host. Un host debe pertenecer al menos a un grupo de hosts. Puede crear un grupo nuevo y vincularlo al host añadiendo un nombre de grupo que no exista.
Interfaces	Un host admite varios tipos de interfaces: *Agent*, *SNMP*, *JMX* e *IPMI*. De forma predeterminada, no se define ninguna interfaz. Para añadir una interfaz nueva, haga clic en *Añadir* en el bloque *Interfaces*, seleccione el tipo de interfaz e introduzca la información de *IP/DNS*, *Conectar a* y *Puerto*. *Nota:* Las interfaces utilizadas en algún item no se pueden eliminar y el enlace *Eliminar* aparece deshabilitado para ellas. La "IP" o el "DNS" de una interfaz SNMP también se utiliza para las traps SNMP. Durante la coincidencia, solo se utiliza la "IP" o el "DNS" seleccionado en la interfaz del host. Consulte Configuración de la monitorización SNMP para obtener información adicional sobre la configuración de una interfaz SNMP (v1, v2 y v3).
Dirección IP	Dirección IP del host (opcional).
Nombre DNS	Nombre DNS del host (opcional).
Conectar a	Al hacer clic en el botón correspondiente, indicará a Zabbix server qué utilizar para recuperar datos de los agents: **IP**: conectarse a la dirección IP del host (recomendado) **DNS**: conectarse al nombre DNS del host
Puerto	Número de puerto TCP/UDP. Los valores predeterminados son: 10050 para Zabbix agent, 161 para SNMP agent, 12345 para JMX y 623 para IPMI.
Predeterminado	Marque el botón de opción para establecer la interfaz predeterminada.
Descripción	Introduzca la descripción del host.
Monitorizado por	Seleccione si el host es monitorizado por: **Server**: el host es monitorizado por Zabbix server; **Proxy**: el host es monitorizado por un proxy independiente; **Grupo de proxies**: el host es monitorizado por un grupo de proxies.
Proxy	Se muestra el nombre del proxy asignado (solo si Zabbix server ha asignado uno del grupo de proxies seleccionado). Este campo solo se muestra si el host es monitorizado por un grupo de proxies.
Habilitado	Cuando la casilla está marcada, el host está habilitado y listo para la monitorización. Cuando la casilla no está marcada, el host está deshabilitado y no se monitoriza: Para las solicitudes de datos pasivas iniciadas por Zabbix server/proxy (por ejemplo, Zabbix agent, SNMP agent, comprobaciones simples), la monitorización se deshabilita después de la sincronización de la configuración. Los triggers y las acciones vinculados al host también se deshabilitan solo después de volver a cargar la configuración. Para las comprobaciones activas de Zabbix agent, la monitorización se detiene dentro del intervalo de tiempo (aproximadamente 5 segundos) en el que Zabbix agent recibe información de que el host se ha deshabilitado. Durante este breve intervalo, el host seguirá recopilando datos localmente para las comprobaciones activas e intentará enviarlos al server/proxy; sin embargo, como el host está marcado como *Deshabilitado*, el server/proxy rechazará los datos. Al deshabilitar el host, sus items se eliminan inmediatamente de la caché del historial (excepto sus últimos valores, que se conservan para los registros).

La pestaña **IPMI** contiene los atributos de gestión de IPMI.

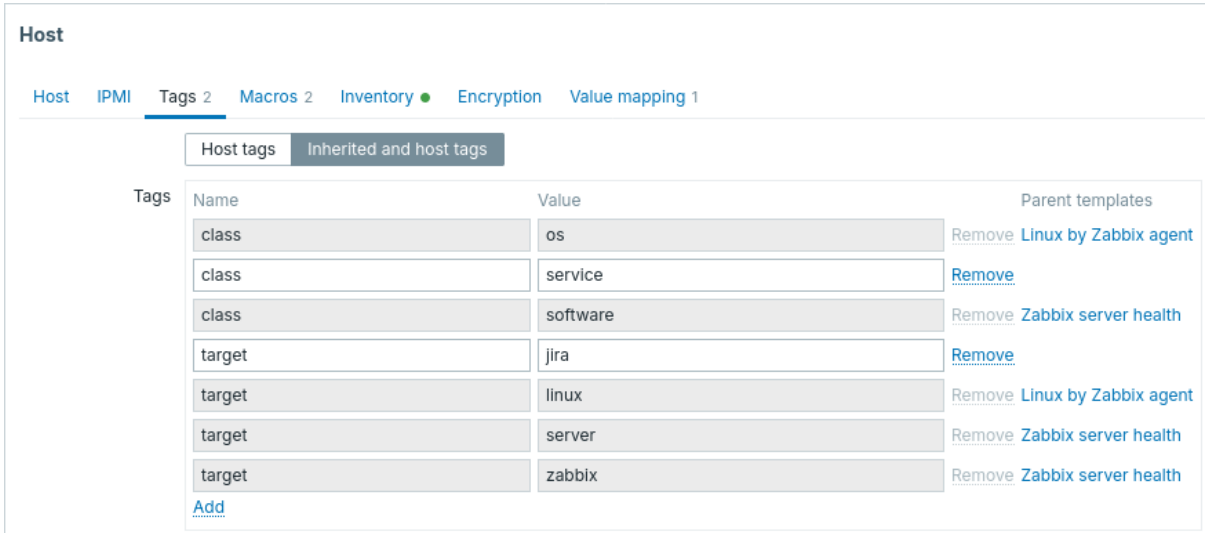
Parámetro	Descripción
Algoritmo de autenticación	Seleccione el algoritmo de autenticación.
Nivel de privilegios	Seleccione el nivel de privilegios.
Nombre de usuario	Nombre de usuario para la autenticación. Se pueden utilizar macros de usuario.
Contraseña	Contraseña para la autenticación. Se pueden utilizar macros de usuario.

La pestaña **Etiquetas** permite definir etiquetas a nivel de host. Todos los problemas de este host se etiquetarán con los valores introducidos aquí.



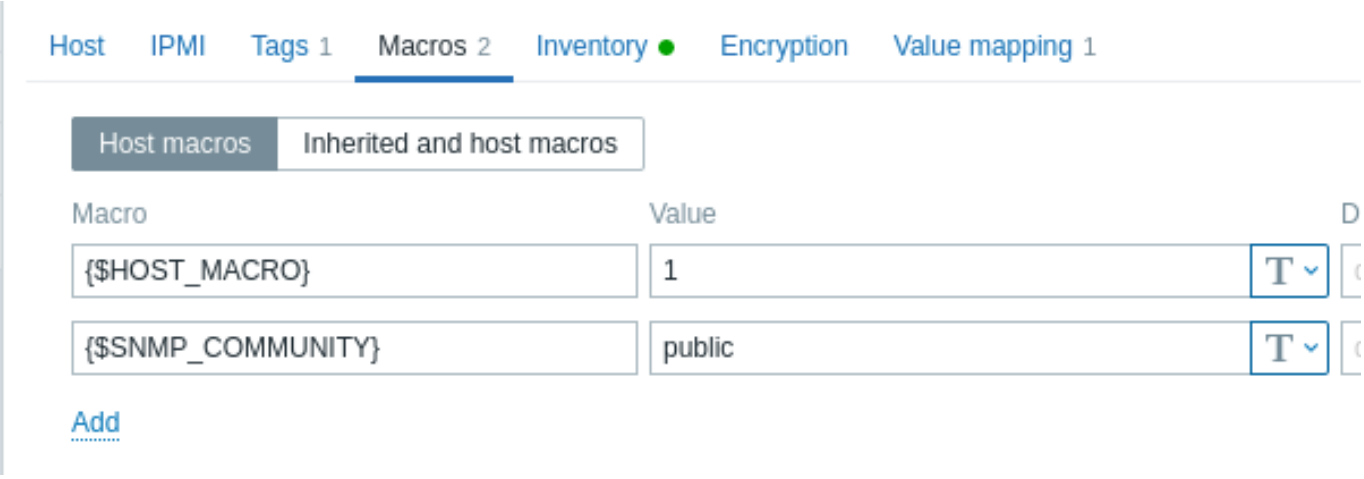
Las etiquetas admiten macros de usuario, macros `{INVENTORY.*}`, `{HOST.HOST}`, `{HOST.NAME}`, `{HOST.CONN}`, `{HOST.DNS}`, `{HOST.IP}`, `{HOST.PORT}` y `{HOST.ID}`.

También puede ver aquí las etiquetas a nivel de template si selecciona la opción *Etiquetas heredadas y del host*. Aquí se muestran todas las etiquetas de usuario definidas para el host, así como su origen.

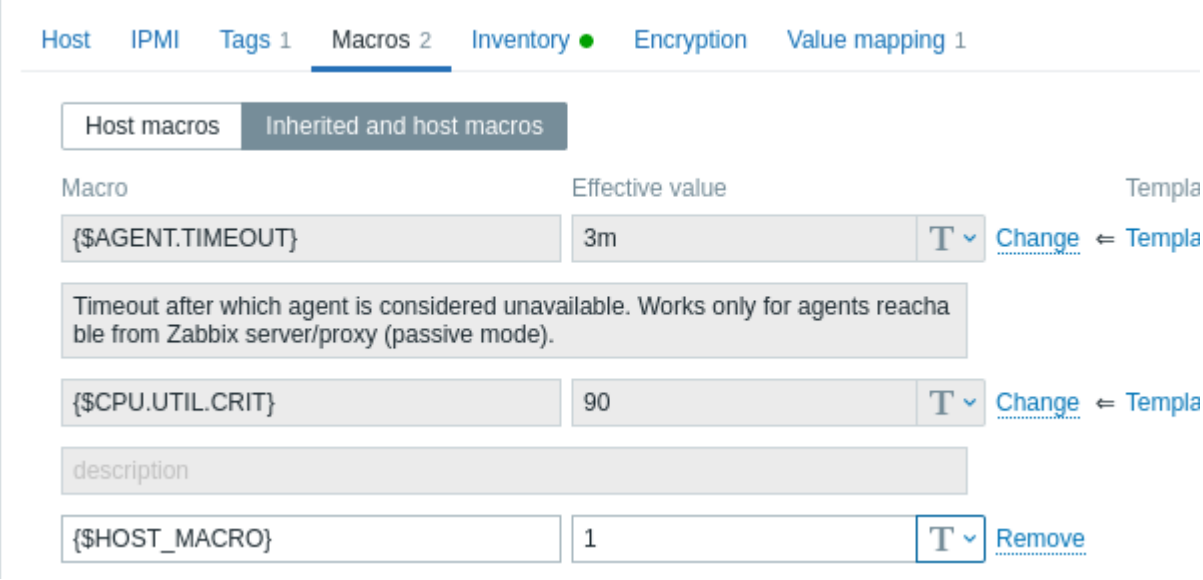


Para mayor comodidad, se proporcionan enlaces a los templates correspondientes.

La pestaña **Macros** permite definir macros de usuario a nivel de host como pares nombre-valor. Tenga en cuenta que los valores de las macros pueden conservarse como texto sin formato, texto secreto o secreto de Vault. También se admite añadir una descripción.

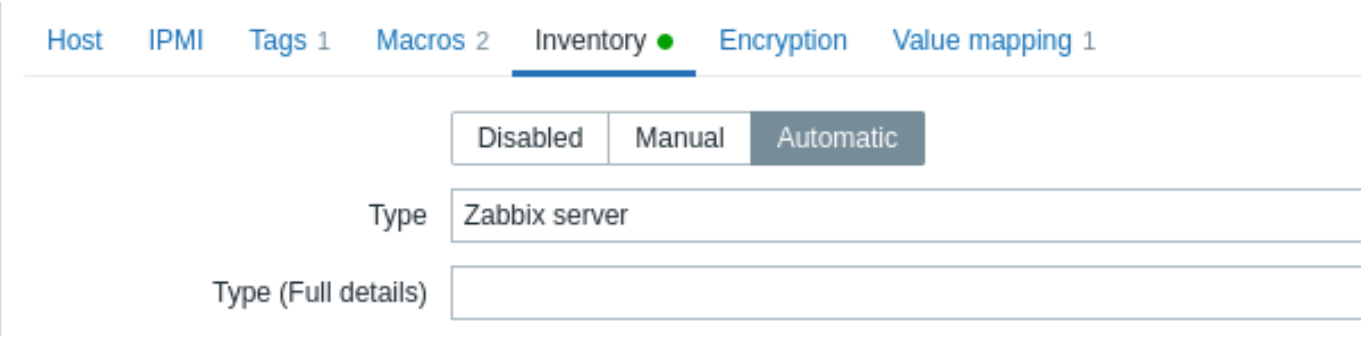


También puede ver aquí las macros de usuario a nivel de template y globales si selecciona la opción *Macros heredadas y del host*. Aquí se muestran todas las macros de usuario definidas para el host, el valor al que se resuelven y su origen.



Para mayor comodidad, se proporcionan enlaces a los templates correspondientes y a la configuración de macros globales. También es posible editar una macro de template o global a nivel de host, creando efectivamente una copia de la macro en el host.

La pestaña **Inventario** permite introducir manualmente información de inventario para el host. También puede seleccionar la opción de habilitar la recopilación de inventario *Automática* o deshabilitarla para este host.



Si el inventario está habilitado (manual o automático), se muestra un punto verde junto al nombre de la pestaña.

Cifrado

La pestaña **Cifrado** permite exigir conexiones cifradas con el host.

Parameter	Description
Connections to host	Cómo se conecta el server o proxy de Zabbix al agent de Zabbix en un host: sin cifrado (predeterminado), usando PSK (clave previamente compartida) o certificado.

Parameter	Description
Connections from host	Seleccione qué tipo de conexiones se permiten desde el host (es decir, desde el agent de Zabbix y Zabbix sender). Se pueden seleccionar varios tipos de conexión al mismo tiempo (útil para pruebas y para cambiar a otro tipo de conexión). El valor predeterminado es "Sin cifrado".
Issuer	Emisor permitido del certificado. El certificado se valida primero con la CA (autoridad de certificación). Si es válido y está firmado por la CA, entonces el campo *Issuer* puede usarse para restringir aún más la CA permitida. Este campo está pensado para usarse si su instalación de Zabbix utiliza certificados de varias CAs. Si este campo está vacío, se acepta cualquier CA.
Subject	Asunto permitido del certificado. El certificado se valida primero con la CA. Si es válido y está firmado por la CA, entonces el campo *Subject* puede usarse para permitir solo un valor de la cadena *Subject*. Si este campo está vacío, se acepta cualquier certificado válido firmado por la CA configurada.
PSK identity	Cadena de identidad de la clave previamente compartida. No coloque información sensible en la identidad de la PSK; se transmite sin cifrar por la red para informar al receptor qué PSK debe usar.
PSK	Clave previamente compartida (cadena hexadecimal). Longitud máxima: 512 dígitos hexadecimales (PSK de 256 bytes) si Zabbix usa la biblioteca GnuTLS u OpenSSL, 64 dígitos hexadecimales (PSK de 32 bytes) si Zabbix usa la biblioteca mbed TLS (PolarSSL). Ejemplo: 1f87b595725ac58dd977beef14b97461a7c1045b9a1c963065002c5473194952

Asignación de valores

La pestaña **Asignación de valores** permite configurar una representación comprensible para los datos de los item en las asignaciones de valores.

Configuración de un grupo de hosts

Resumen

Los grupos de hosts se utilizan para la agrupación lógica de hosts y para asignarles permisos de usuario.

Cada host debe tener asignado al menos un grupo de hosts. Un host puede pertenecer a varios grupos de hosts, y cada grupo de hosts puede contener varios hosts.

Tenga en cuenta que, en Zabbix, todos los permisos se basan en grupos: grupos de usuarios, grupos de hosts y grupos de template. Por lo tanto, incluso si un único usuario necesita acceso a un único host, este se concede agregando al usuario a un grupo de usuarios y asignando a ese grupo de usuarios permisos sobre el grupo de hosts que contiene ese host.

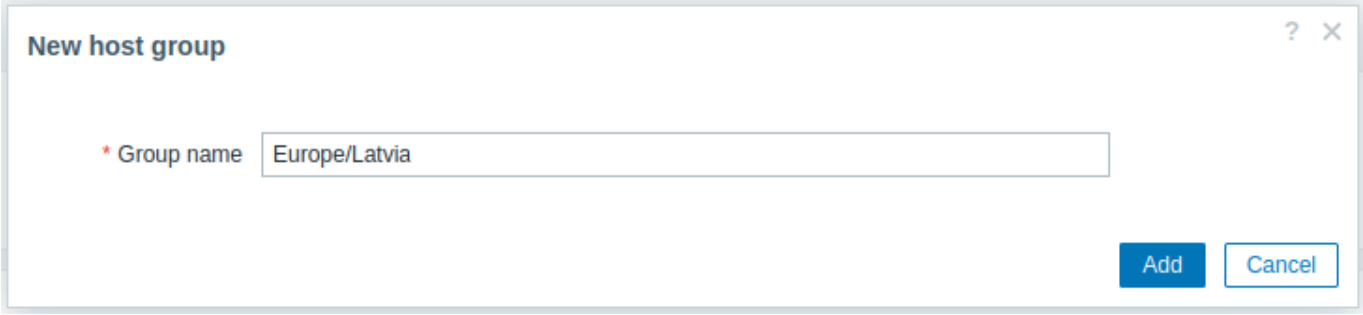
Configuración

> **Attention:**
> Solo los usuarios Super admin pueden crear grupos de hosts.

Hay dos opciones para crear un grupo de hosts en el frontend de Zabbix.

Opción uno:

1. Vaya a *Data collection* > *Host groups*.
2. Haga clic en *Create host group* en la esquina superior derecha de la pantalla.
3. Introduzca el nombre del grupo en el formulario.



Opción dos: al configurar un host, introduzca un nombre de grupo inexistente en el campo de entrada *Host groups*.

Una vez creado el grupo de hosts, puede hacer clic en el nombre del grupo en la lista, en *Data collection* > *Host groups*, para editar el nombre del grupo, clonar el grupo o eliminarlo.

Eliminar un grupo de hosts solo elimina el grupo lógico, no los hosts del grupo. No es posible eliminar un grupo de hosts que sea el único grupo de cualquier host existente.

Creación de subgrupos de equipos

Un subgrupo de equipos (o grupo de equipos anidado) es un hijo del grupo de equipos principal que lo contiene.

Un subgrupo se crea utilizando la barra inclinada '/' en el campo de entrada del nombre del grupo para denotar su relación con el/los grupo(s) principal(es). Por ejemplo:

- al introducir `Europe/Latvia` se crea el subgrupo `Europe/Latvia` dentro del grupo principal `Europe`.
- al introducir `Europe/Latvia/Riga/Zabbix servers` se crea el subgrupo correspondiente dentro de los grupos principales anidados `Europe`, `Europe/Latvia`, `Europe/Latvia/Riga`.

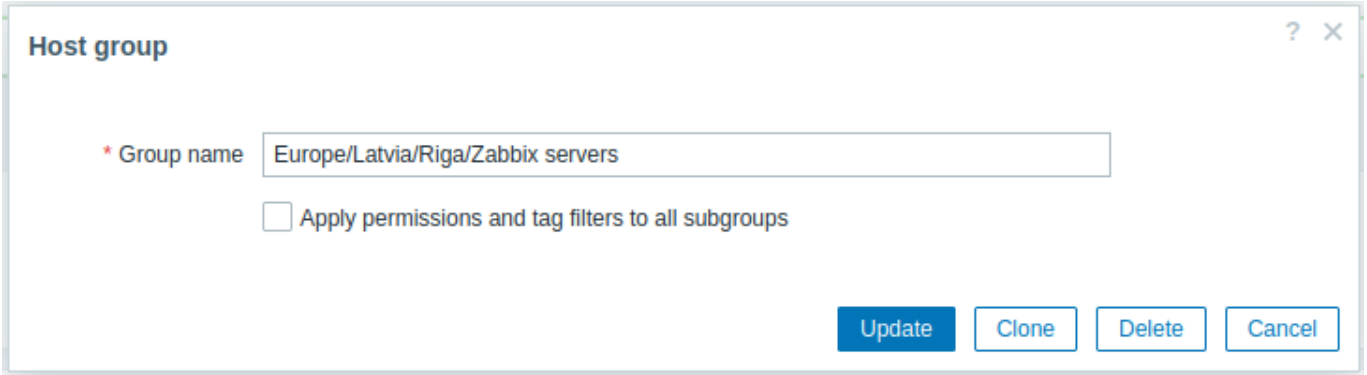
Al crear un subgrupo, no se permite el uso de barras inclinadas al principio o al final, ni varias barras inclinadas seguidas. No se admite el escape de '/'.

No es necesario crear ningún grupo de equipos principal antes de crear un subgrupo. Puede elegir si desea comenzar creando un subgrupo (por ejemplo, `Europe/Latvia`) o cualquier grupo de equipos principal (en nuestro ejemplo, `Europe`). Si comienza creando un subgrupo, el/los grupo(s) de equipos principal(es) **no** se crearán automáticamente.

Permisos para grupos de hosts

- Al crear un subgrupo de un grupo de hosts principal existente (por ejemplo, al crear `Europe/Latvia` cuando `Europe` ya existe), los permisos del grupo de usuarios para el subgrupo se heredan del grupo principal.
- Al crear un grupo de hosts principal para un subgrupo existente (por ejemplo, al crear `Europe` cuando `Europe/Latvia` ya existe), no se establecen permisos para el grupo principal.

Al editar cualquier grupo de hosts, también puede configurar una opción adicional, *Aplicar permisos y filtros de etiquetas a todos los subgrupos*.



Al marcar esta casilla y hacer clic en *Update*, se aplicará el mismo nivel de permisos y filtros de etiquetas a todos los subgrupos actuales y futuros del grupo de hosts que se está editando.

Por lo tanto, si a algún grupo de usuarios se le han asignado permisos distintos a los subgrupos del grupo de hosts que se está editando, al marcar la casilla se concederán a todos los subgrupos actuales y futuros los mismos permisos de usuario y permisos basados en etiquetas que el grupo que se está editando.

Tenga en cuenta que esta opción no se guarda en la base de datos y sobrescribirá los permisos existentes. Cualquier cambio realizado mediante esta opción solo puede revertirse manualmente.

Inventario

Descripción general

Puede mantener el inventario de dispositivos en red en Zabbix.

Hay un menú especial *Inventario* en la interfaz de Zabbix. Sin embargo, inicialmente no verá ningún dato allí y no es donde se ingresan los datos. La creación de datos de inventario se realiza manualmente al configurar un equipo o automáticamente utilizando algunas opciones de población automática.

Creando inventario

Modo manual

Al configurar un host, en la pestaña *Inventario* puede introducir detalles como el tipo de dispositivo, el número de serie, la ubicación, la persona responsable, las URL, etc.; estos datos se utilizarán para completar la información del inventario.

Si se incluye una URL en la información del inventario del host y comienza por «http» o «https», se mostrará como un enlace en el que se puede hacer clic en la sección *Inventario*.

Modo automático

El inventario del host también se puede completar automáticamente. Para que esto funcione, al configurar un host, el modo de inventario en la pestaña *Inventario* debe establecerse en *Automático*.

A continuación, puede configurar items del host para completar cualquier campo del inventario del host con su valor, indicando el campo de destino mediante el atributo correspondiente (denominado *El item completará el campo de inventario del host*) en la configuración del item.

Items especialmente útiles para la recopilación automatizada de datos del inventario:

- system.hw.chassis[full|type|vendor|model|serial] - el valor predeterminado es [full], se necesitan permisos de root
- system.hw.cpu[all|cpunum,full|maxfreq|vendor|model|curfreq] - el valor predeterminado es [all,full]
- system.hw.devices[pci|usb] - el valor predeterminado es [pci]
- system.hw.macaddr[interface,short|full] - el valor predeterminado es [all,full], interface es una expresión regular
- system.sw.arch
- system.sw.os[name|short|full] - el valor predeterminado es [name]
- system.sw.packages[regexp,manager,short|full] - el valor predeterminado es [all,all,full].

Selección del modo de inventario

El modo de inventario se puede seleccionar en el formulario de configuración del host.

El modo de inventario predeterminado para los hosts nuevos se selecciona según la configuración *Default host inventory mode* en *Administration* > *General* > *Other*.

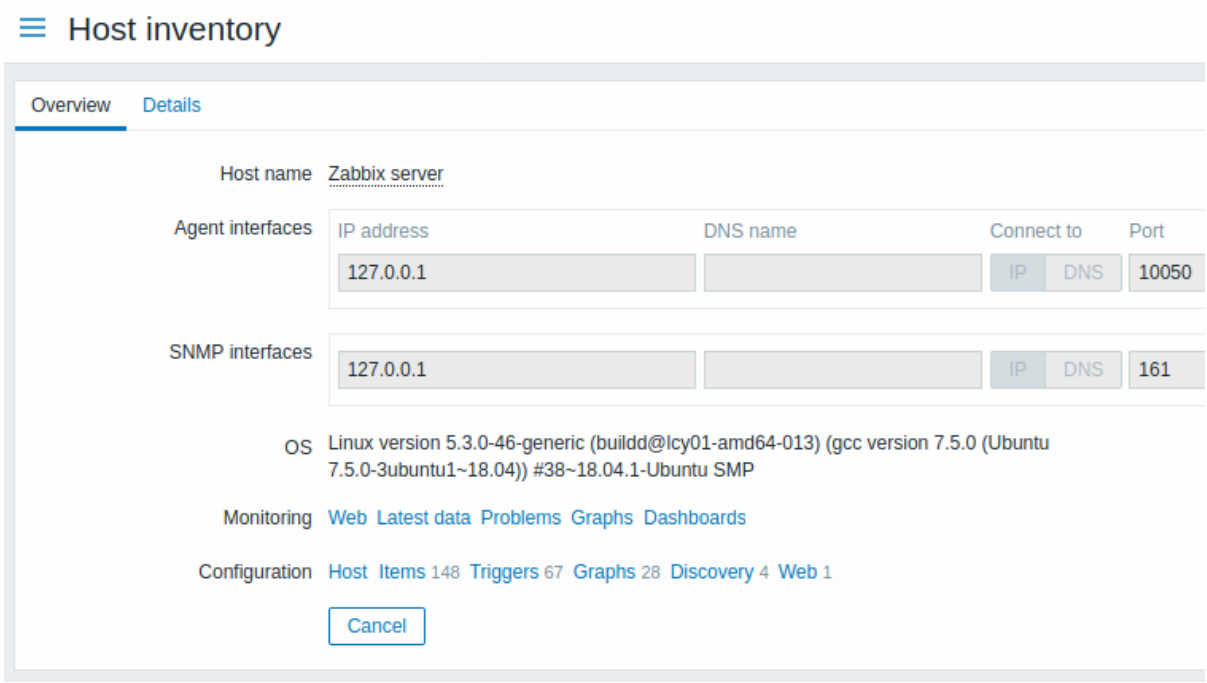
Para los hosts añadidos mediante acciones de descubrimiento de red o autorregistro, es posible definir una operación *Set host inventory mode* seleccionando el modo manual o automático. Esta operación anula la configuración *Default host inventory mode*.

Resumen del inventario

Los detalles de todos los datos de inventario existentes están disponibles en el menú *Inventory*.

En *Inventory* > *Overview* puede obtener un recuento de hosts por varios campos del inventario.

En *Inventory* > *Hosts* puede ver todos los hosts que tienen información de inventario. Al hacer clic en el nombre del host, se mostrarán los detalles del inventario en un formulario.



La pestaña **Overview** muestra:

Parameter	Description
Host name	Nombre del host. Al hacer clic en el nombre se abre un menú con los scripts definidos para el host. El nombre del host se muestra con un icono naranja si el host está en mantenimiento.

Parameter	Description
Visible name	Nombre visible del host (si está definido).
Host (Agent, SNMP, JMX, IPMI) interfaces	Este bloque proporciona detalles de las interfaces configuradas para el host.
OS	Campo de inventario del sistema operativo del host (si está definido).
Hardware	Campo de inventario del hardware del host (si está definido).
Software	Campo de inventario del software del host (si está definido).
Description	Descripción del host.
Monitoring	Enlaces a secciones de monitorización con datos de este host: *Web*, *Latest data*, *Problems*, *Graphs*, *Dashboards*.
Configuration	Enlaces a secciones de configuración para este host: *Host*, *Items*, *Triggers*, *Graphs*, *Discovery*, *Web*. La cantidad de entidades configuradas se muestra después de cada enlace.

La pestaña **Details** muestra todos los campos de inventario que están rellenados (no están vacíos).

Macros de inventario

Hay macros de inventario de host `{INVENTORY.*}` disponibles para su uso en notificaciones, por ejemplo:

"El server en {INVENTORY.LOCATION1} tiene un problema, la persona responsable es {INVENTORY.CONTACT1}, el número de teléfono es {INVENTORY.POC.PRIMARY.PHONE.A1}."

Para más detalles, consulte la página Macros compatibles.

Actualización masiva

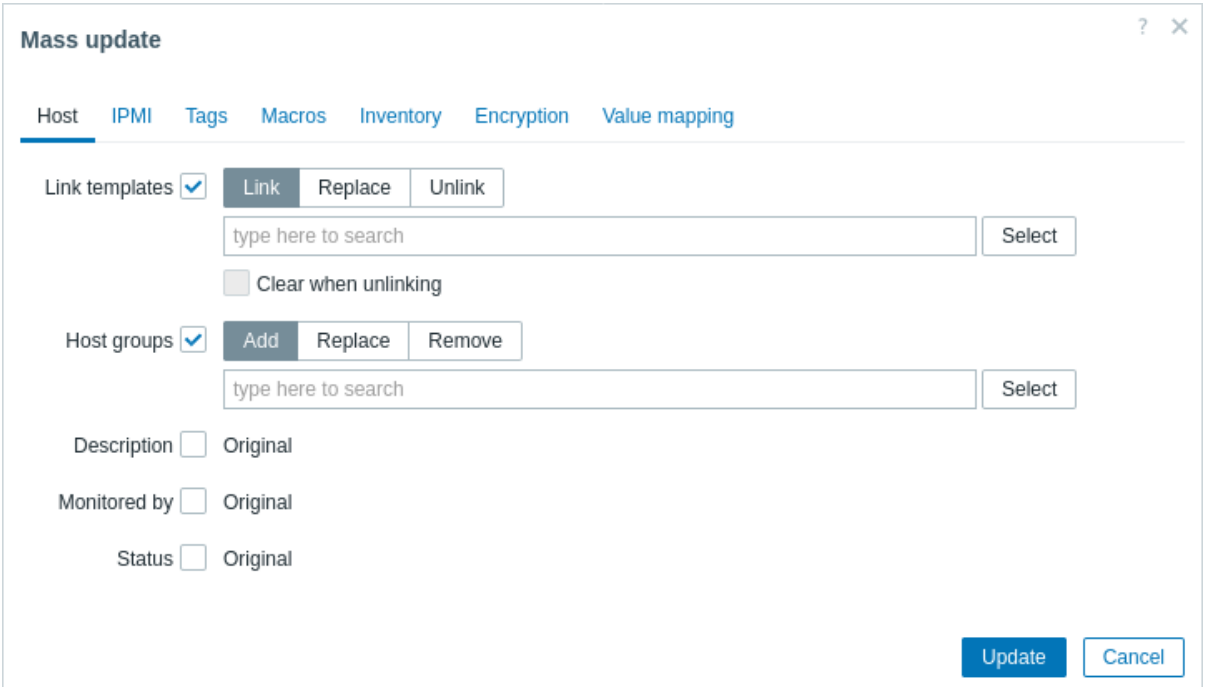
Descripción general

A veces puede que desee cambiar algún atributo para varios equipos a la vez. En lugar de abrir cada equipo individualmente para editarlo, puede utilizar la función de actualización masiva para ello.

Uso de la actualización masiva

Para actualizar varios hosts de forma masiva, haga lo siguiente:

- Marque las casillas situadas delante de los hosts que desea actualizar en la lista de hosts
- Haga clic en *Actualización masiva* debajo de la lista
- Vaya a la pestaña con los atributos necesarios (*Host*, *IPMI*, *Tags*, *Macros*, *Inventory*, *Encryption* o *Value mapping*)
- Marque las casillas de los atributos que desea actualizar e introduzca un nuevo valor para ellos



Las siguientes opciones están disponibles al seleccionar el botón correspondiente para actualizar la vinculación de **template**:

- *Vincular* - especifique qué templates adicionales desea vincular

- *Reemplazar* - especifique qué templates desea vincular y desvincule cualquier template que estuviera vinculado anteriormente a los hosts
- *Desvincular* - especifique qué templates desea desvincular

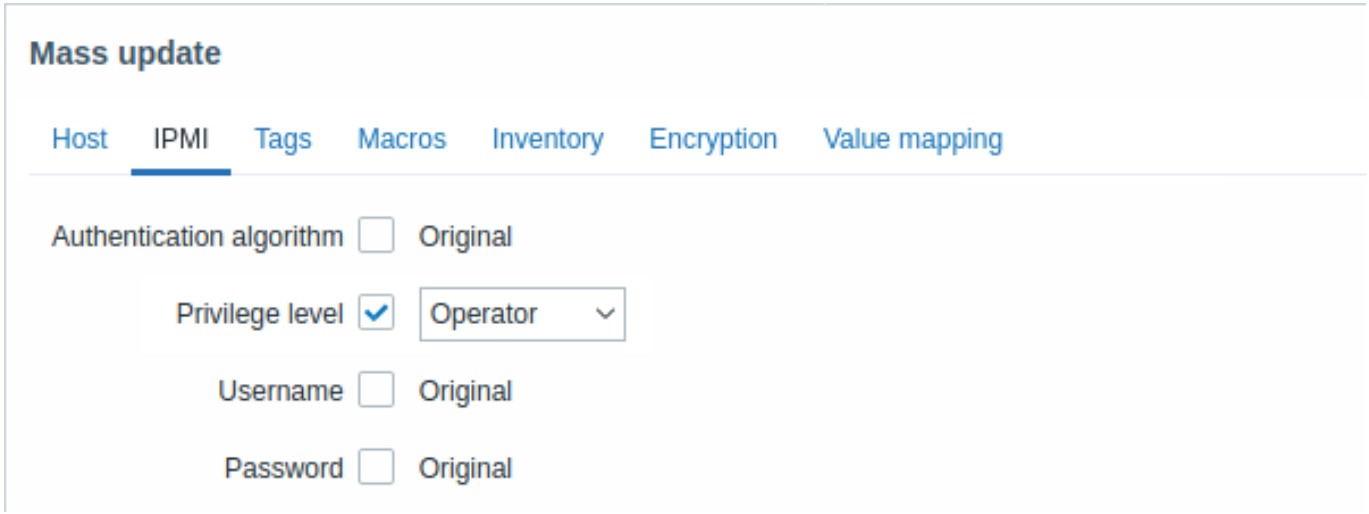
Para especificar los templates que desea vincular o desvincular, comience a escribir el nombre del template en el campo de autocompletado hasta que aparezca una lista desplegable con los templates coincidentes. Desplácese hacia abajo para seleccionar el template necesario.

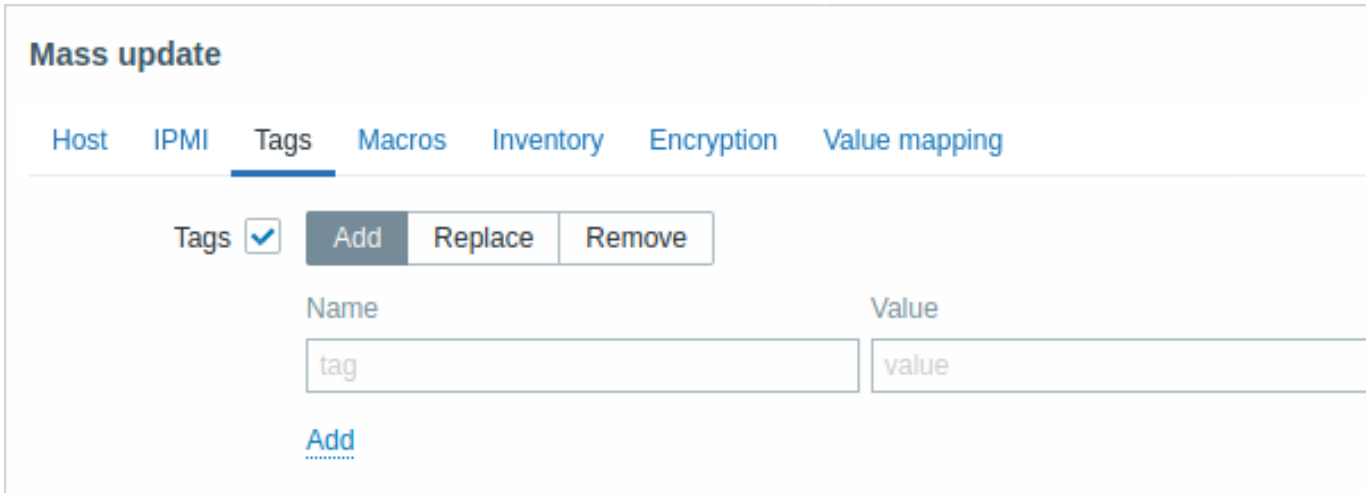
La opción *Borrar al desvincular* permite no solo desvincular los templates vinculados anteriormente, sino también eliminar todas las entidades heredadas de ellos (items, triggers, gráficos, etc.).

Las siguientes opciones están disponibles al seleccionar el botón correspondiente para actualizar el **grupo de hosts**:

- *Añadir* - especifique grupos de hosts adicionales de los existentes o introduzca grupos de hosts completamente nuevos para los hosts;
- *Reemplazar* - elimine el host de todos los grupos de hosts existentes y reemplácelos por los especificados en este campo (grupos de hosts existentes o nuevos);
- *Eliminar* - elimine grupos de hosts específicos de los hosts.

Estos campos disponen de autocompletado: al comenzar a escribir en ellos, aparece una lista desplegable con los grupos de hosts coincidentes. Si el grupo de hosts es nuevo, también aparece en la lista desplegable y se indica mediante *(nuevo)* después de la cadena. Desplácese hacia abajo para seleccionarlo.

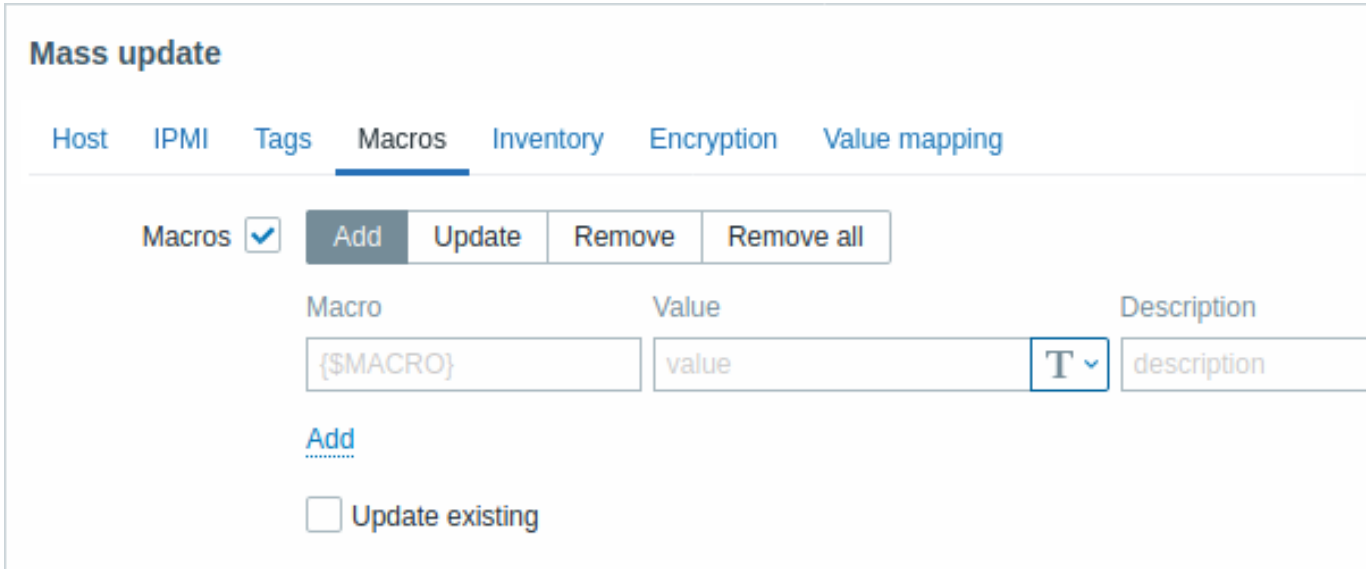




Las siguientes opciones están disponibles al seleccionar el botón correspondiente para actualizar las etiquetas:

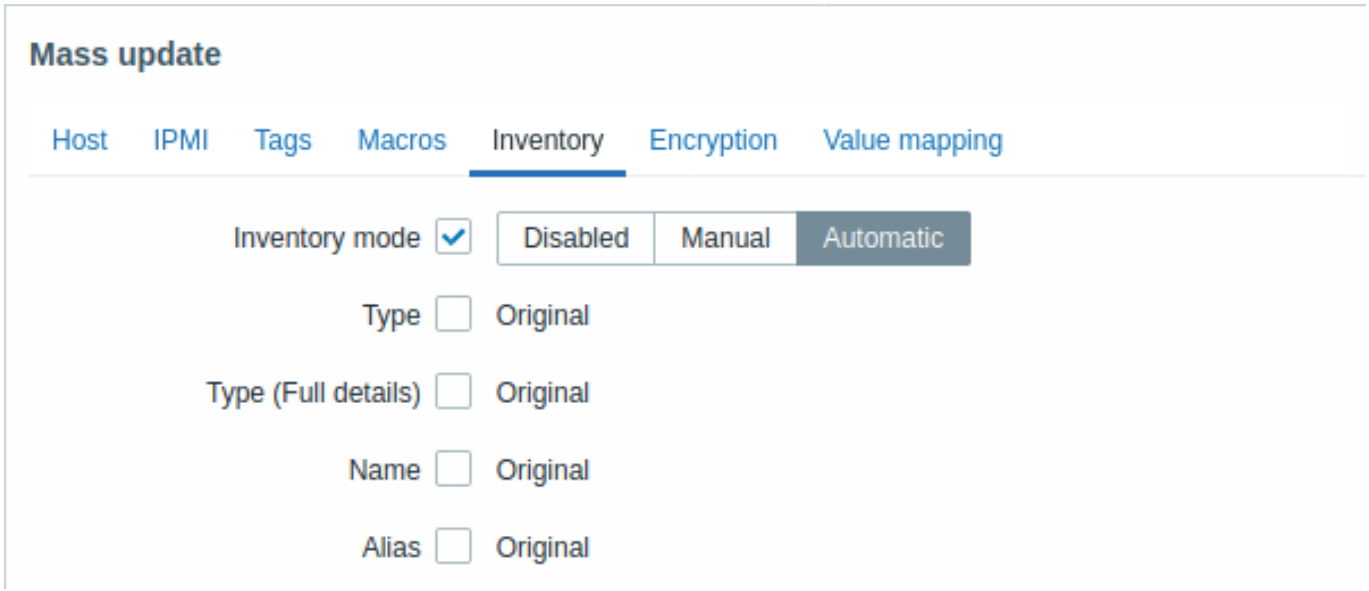
- *Añadir* - añada nuevas etiquetas para los hosts;
- *Reemplazar* - elimine todas las etiquetas existentes del host y reemplácelas por las especificadas a continuación;
- *Eliminar* - elimine las etiquetas especificadas de los hosts.

Las macros de usuario, las macros `{INVENTORY.*}`, `{HOST.HOST}`, `{HOST.NAME}`, `{HOST.CONN}`, `{HOST.DNS}`, `{HOST.IP}`, `{HOST.PORT}` y `{HOST.ID}` son compatibles con las etiquetas. Tenga en cuenta que las etiquetas con el mismo nombre pero con valores diferentes no se consideran «duplicadas» y pueden añadirse al mismo host.

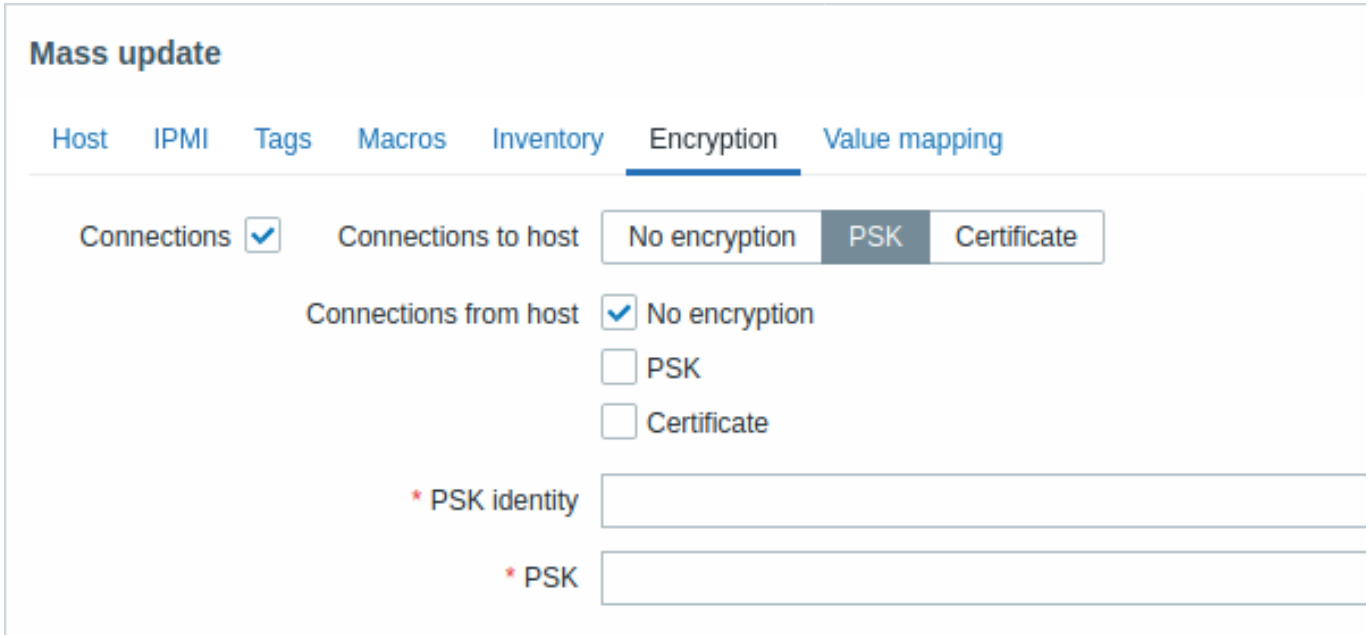


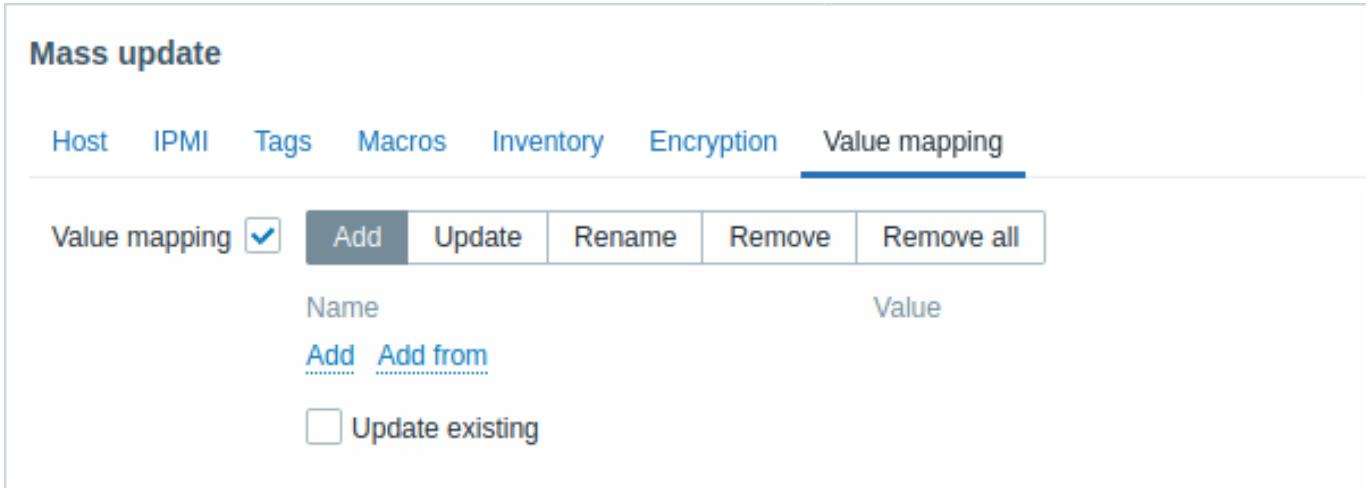
Las siguientes opciones están disponibles al seleccionar el botón correspondiente para actualizar las macros:

- *Añadir* - especifique macros de usuario adicionales para los hosts. Si la casilla *Actualizar existentes* está marcada, se actualizarán el valor, el tipo y la descripción del nombre de macro especificado. Si no está marcada, si ya existe una macro con ese nombre en el host o hosts, no se actualizará.
- *Actualizar* - reemplace los valores, tipos y descripciones de las macros especificadas en esta lista. Si la casilla *Añadir faltantes* está marcada, la macro que no existía anteriormente en un host se añadirá como una macro nueva. Si no está marcada, solo se actualizarán las macros que ya existan en un host.
- *Eliminar* - elimine las macros especificadas de los hosts. Si la casilla *Excepto las seleccionadas* está marcada, se eliminarán todas las macros excepto las especificadas en la lista. Si no está marcada, solo se eliminarán las macros especificadas en la lista.
- *Eliminar todas* - elimine todas las macros de usuario de los hosts. Si la casilla *Confirmo eliminar todas las macros* no está marcada, se abrirá una nueva ventana emergente solicitando confirmar la eliminación de todas las macros.



Para poder actualizar masivamente los campos de inventario, el *Modo de inventario* debe estar establecido en «Manual» o «Automático».





Los botones con las siguientes opciones están disponibles para actualizar el mapa de valores:

- *Añadir* - añada mapas de valores a los hosts. Si marca *Actualizar existentes*, se actualizarán todas las propiedades del mapa de valores con este nombre. De lo contrario, si ya existe un mapa de valores con ese nombre, no se actualizará.
- *Actualizar* - actualice los mapas de valores existentes. Si marca *Añadir faltantes*, se añadirá como un mapa de valores nuevo un mapa de valores que no existía anteriormente en un host. De lo contrario, solo se actualizarán los mapas de valores que ya existan en un host.
- *Renombrar* - asigne un nombre nuevo a un mapa de valores existente
- *Eliminar* - elimine los mapas de valores especificados de los hosts. Si marca *Excepto los seleccionados*, se eliminarán todos los mapas de valores **excepto** los especificados.
- *Eliminar todos* - elimine todos los mapas de valores de los hosts. Si la casilla *Confirmo eliminar todos los mapas de valores* no está marcada, se abrirá una nueva ventana emergente solicitando confirmar la eliminación.

Cuando haya terminado de realizar todos los cambios necesarios, haga clic en *Actualizar*. Los atributos se actualizarán según corresponda para todos los hosts seleccionados.

Items

Descripción general

Una métrica es una métrica individual.

Las métricas se utilizan para recopilar datos. Una vez que haya configurado un equipo, debe añadir métricas para obtener datos reales. Una forma de añadir rápidamente muchas métricas es adjuntar una de las plantillas predefinidas a un equipo. Sin embargo, para optimizar el rendimiento del sistema, puede que necesite ajustar las plantillas para tener tantas métricas y una monitorización tan frecuente como sea necesario.

Para especificar qué tipo de datos recopilar de un equipo, utilice la clave de la métrica. Por ejemplo, una métrica con el nombre de clave **system.cpu.load** recopilará datos de carga del procesador, mientras que una métrica con el nombre de clave **net.if.in** recopilará información de tráfico entrante.

Se pueden especificar parámetros adicionales entre corchetes después del nombre de la clave. Por ejemplo, system.cpu.load**[avg5]** devolverá la carga promedio del procesador durante los últimos 5 minutos, mientras que net.if.in**[eth0]** mostrará el tráfico entrante en la interfaz "eth0".

> **Note:**
> Consulte las secciones individuales de tipos de métricas para ver todos los tipos de métricas y claves de métricas compatibles.

Continúe con crear y configurar una métrica.

Creación de un item

Descripción general

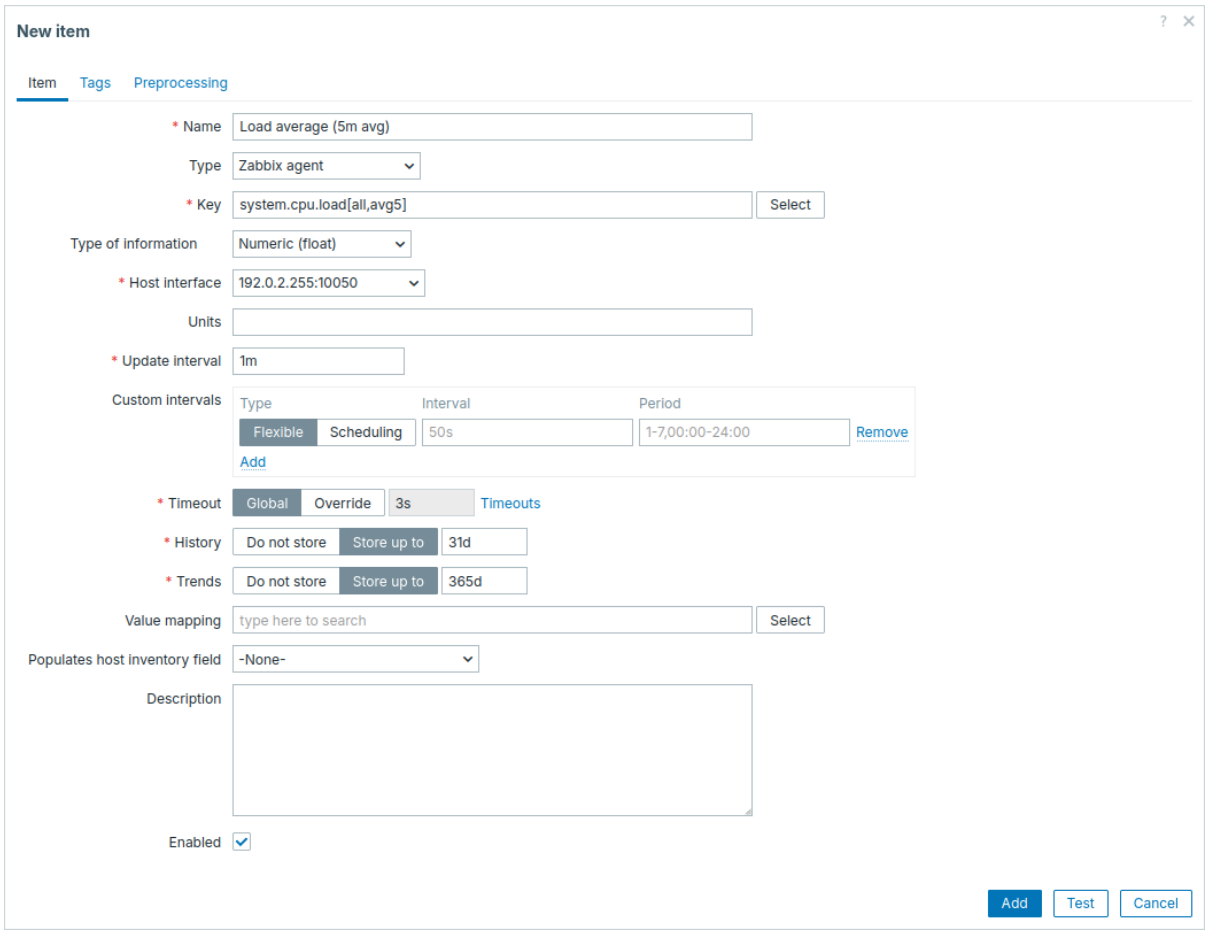
Para crear un item en el frontend de Zabbix, haga lo siguiente:

1. Vaya a *Recopilación de datos* > *Hosts*.
2. Haga clic en *Items* en la fila del host.
3. Haga clic en *Crear item* en la esquina superior derecha de la pantalla.
4. Introduzca los parámetros del item en el formulario.

Como alternativa, puede crear un item abriendo uno existente, haciendo clic en el botón *Clonar* y guardándolo con otro nombre.

Configuración

La pestaña **Item** contiene los atributos generales del item.



Todos los campos de entrada obligatorios están marcados con un asterisco rojo.

Parámetro	Descripción
Nombre	Nombre del item. Se admiten macros de usuario.
Tipo	Tipo de item. Consulte las secciones individuales sobre tipos de item.
Clave	Clave del item (hasta 2048 caracteres). Las claves de item admitidas se encuentran en las secciones individuales sobre tipos de item. La clave debe ser única dentro de un solo host. Si el tipo de clave es 'Zabbix agent', 'Zabbix agent (active)' o 'Simple check', el valor de la clave debe ser compatible con Zabbix agent o Zabbix server. Consulte también el formato correcto de la clave.
Tipo de información	Tipo de datos utilizado para validar el valor del item y almacenarlo en la base de datos, tanto antes como después de cualquier conversión: **Numérico (sin signo)**: entero sin signo de 64 bits; **Numérico (flotante)**: número de coma flotante de 64 bits; **Carácter**: texto corto; **Registro**: texto largo con o sin propiedades de registro (marca de tiempo, origen, gravedad, ID del evento de registro); **Texto**: texto largo; **Binario**: número binario (admitido únicamente para items dependientes); **JSON**: datos JSON estructurados, almacenados de forma nativa en la base de datos (no admitido para items calculados). Los items que devuelven valores con el tipo de datos binario o JSON no se admiten en fórmulas de items calculados ni en expresiones de trigger. Consulte límites de datos de los items para conocer los límites de almacenamiento por tipo de datos y backend de base de datos. Para los items que devuelven datos en un único formato específico, se selecciona automáticamente un tipo de datos coincidente; por ejemplo, `system.cpu.load` y `system.cpu.util` devuelven un número de coma flotante de 64 bits, por lo que *Numérico (flotante)* se selecciona automáticamente.
Interfaz del host	Seleccione la interfaz del host. Este campo está disponible al editar un item en el nivel del host.
Unidades	Si se establece un símbolo de unidad, Zabbix aplica un posprocesamiento al valor recibido del item y lo muestra con el sufijo de unidad especificado. Símbolos de unidad admitidos con formato especial (y ejemplos de valor recibido → valor mostrado): **B**: bytes (1024 → 1 KB) **Bps**: bytes por segundo (1024 → 1 KBps) **s**: segundos, mostrados utilizando hasta las tres unidades de tiempo más grandes distintas de cero (881764 → 10d 4h 56m) **uptime**: tiempo transcurrido en hh:mm:ss o N días, hh:mm:ss (881764 → 10 días, 04:56:04) **unixtime**: marca de tiempo Unix, con formato yyyy.mm.dd hh:mm:ss (881764 → 1970-01-11 04:56:04 AM); para obtener el formato correcto, el valor recibido del item debe ser *Numérico (sin signo)*. Para otras unidades (como Hz, W, etc.), si el valor recibido supera 1000, se divide entre 1000 y se muestra con el prefijo correspondiente (5000 → 5 KHz, 881764 → 881.76 KW). Si el símbolo de unidad lleva el prefijo `!`, se deshabilitan la conversión y la aplicación de prefijos (1024 !B → 1024 B, 61 !s → 61 s). Para consultar más ejemplos y detalles sobre los símbolos de unidad y la conversión de unidades, consulte Sufijos de los valores de los items.

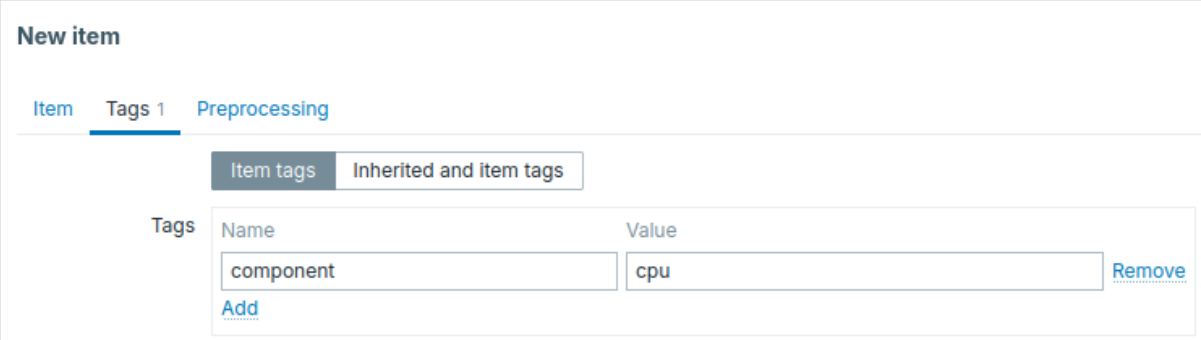
Parámetro	Descripción
Intervalo de actualización	Recupere un nuevo valor para este item cada N segundos. El intervalo de actualización máximo permitido es de 86400 segundos (1 día). Se admiten sufijos de tiempo, por ejemplo, 30s, 1m, 2h, 1d. Se admiten macros de usuario. Una sola macro debe ocupar todo el campo. No se admiten varias macros en un campo ni macros combinadas con texto. *Nota*: El intervalo de actualización solo puede establecerse en '0' si existen intervalos personalizados con un valor distinto de cero. Si se establece en '0' y existe un intervalo personalizado (flexible o programado) con un valor distinto de cero, el item se sondeará durante la duración del intervalo personalizado. *Tenga en cuenta* que el primer sondeo del item después de que este se active o después de cambiar el intervalo de actualización puede producirse antes del valor configurado. Los nuevos items se comprobarán en un plazo de 60 segundos desde su creación, a menos que tengan un intervalo de actualización programado o flexible y *Intervalo de actualización* esté establecido en 0. Un item pasivo existente puede sondearse inmediatamente para obtener su valor pulsando el botón *Ejecutar ahora*.
Intervalos personalizados	Puede crear reglas personalizadas para comprobar el item: **Flexible**: crea una excepción al *Intervalo de actualización* (intervalo con una frecuencia diferente). **Programación**: crea una programación de sondeo personalizada. Para obtener información detallada, consulte Intervalos personalizados. Se admiten sufijos de tiempo en el campo *Intervalo*, por ejemplo, 30s, 1m, 2h, 1d. Se admiten macros de usuario. Una sola macro debe ocupar todo el campo. No se admiten varias macros en un campo ni macros combinadas con texto.
Tiempo de espera	Establezca el tiempo de espera de la comprobación del item (disponible para los tipos de item admitidos). Seleccione la opción de tiempo de espera: **Global**: se utiliza el tiempo de espera del proxy/global (se muestra en el campo *Tiempo de espera* atenuado). **Anulación**: se utiliza un tiempo de espera personalizado (establecido en el campo *Tiempo de espera*; rango permitido: 1 - 600s). Se admiten sufijos de tiempo, por ejemplo, 30s, 1m, y macros de usuario. Al hacer clic en el enlace *Tiempos de espera*, puede configurar los tiempos de espera del proxy o los tiempos de espera globales (si no se utiliza un proxy). Tenga en cuenta que el enlace *Tiempos de espera* solo está visible para los usuarios de tipo *Super admin* con permisos para las secciones del frontend *Administración* > *General* o *Administración* > *Proxies*.
Historial	Seleccione una de las siguientes opciones: **No almacenar**: no se almacena el historial del item. Es útil para items maestros si solo los items dependientes necesitan conservar el historial. Esta configuración no puede anularse mediante las configuraciones globales del housekeeper. **Almacenar hasta**: especifique durante cuánto tiempo se conservará el historial detallado en la base de datos (de 1 hora a 25 años). El housekeeper eliminará los datos más antiguos. Se almacena en segundos. Se admiten sufijos de tiempo, por ejemplo, 2h, 1d. Se admiten macros de usuario. El valor de *Almacenar hasta* puede anularse globalmente en *Administración* > *Mantenimiento*. Si existe una configuración global de anulación, se muestra un icono de información naranja . Al colocar el cursor sobre él, se muestra un mensaje de advertencia, por ejemplo, *Anulado por la configuración global del housekeeper (1d)*. Se recomienda conservar los valores registrados durante el menor tiempo posible para reducir el tamaño del historial de valores en la base de datos. En lugar de almacenar un historial de valores extenso, puede almacenar datos de tendencias durante más tiempo. Consulte también Historial y tendencias.

Parámetro	Descripción
Tendencias	Seleccione una de las siguientes opciones: **No almacenar**: no se almacenan las tendencias. Esta configuración no puede anularse mediante las configuraciones globales del housekeeper. **Almacenar hasta**: especifique durante cuánto tiempo se conservará en la base de datos el historial agregado (mínimo, máximo, promedio y cantidad por hora) (de 1 día a 25 años). El housekeeper eliminará los datos más antiguos. Se almacena en segundos. Se admiten sufijos de tiempo, por ejemplo, 24h, 1d. Se admiten macros de usuario. El valor de *Almacenar hasta* puede anularse globalmente en *Administración* > *Mantenimiento*. Si existe una configuración global de anulación, se muestra un icono de información naranja . Al colocar el cursor sobre él, se muestra un mensaje de advertencia, por ejemplo, *Anulado por la configuración global del housekeeper (7d)*. *Nota:* La conservación de tendencias no está disponible para datos no numéricos: carácter, registro y texto. Consulte también Historial y tendencias.
Asignación de valores	Aplique una asignación de valores a este item. La asignación de valores no cambia los valores recibidos; solo sirve para mostrar los datos. Funciona con items *Numérico (sin signo)*, *Numérico (flotante)* y *Carácter*. Por ejemplo, ”Estados de los servicios de Windows”.
Formato de hora del registro	Disponible únicamente para items de tipo **Registro**. Marcadores de posición admitidos: **y**: *Año (1970-2038)*. **M**: *Mes (01-12)*. **d**: *Día (01-31)*. **h**: *Hora (00-23)*. **m**: *Minuto (00-59)*. **s**: *Segundo (00-59)*. Si se deja en blanco, la marca de tiempo se establecerá en 0 en tiempo Unix, lo que representa el 1 de enero de 1970. Por ejemplo, considere la siguiente línea del archivo de registro de Zabbix agent: ” 23480:20100328:154718.045 Zabbix agent started. Zabbix 1.8.2 (revision 11211).” Comienza con seis posiciones de caracteres para el PID, seguidas de la fecha, la hora y el resto del mensaje. El formato de hora del registro para esta línea sería ”pppppp:yyyyMMdd:hhmmss”. Tenga en cuenta que los caracteres ”p” y ”:” son marcadores de posición y pueden ser cualquier carácter excepto ”yMdhms”.
Rellena el campo de inventario del host	Puede seleccionar un campo del inventario del host que se rellenará con el valor del item. Esto funcionará si la población automática del inventario está habilitada para el host y con items *Numérico (sin signo)*, *Numérico (flotante)*, *Carácter* y *Texto*.
Descripción	Introduzca una descripción del item. Se admiten macros de usuario.
Habilitado	Marque la casilla para habilitar el item y que pueda procesarse. Al deshabilitar el item, se elimina inmediatamente de la caché del historial (excepto su último valor, que se conserva para los registros).
Datos más recientes	Haga clic en el enlace para ver los datos más recientes del item. Este enlace solo está disponible al editar un item ya existente.

Note:
Los campos específicos del tipo de item se describen en las páginas correspondientes.

Note:
Al editar en el nivel de un host un item existente de nivel de template, varios campos son de solo lectura. Puede utilizar el enlace del encabezado del formulario para ir al nivel del template y editarlos allí, teniendo en cuenta que los cambios realizados en el nivel del template modificarán el item para todos los hosts a los que esté vinculado el template.

La pestaña **Etiquetas** permite definir etiquetas a nivel del item.



Preprocesamiento de valores de item

La pestaña **Preprocesamiento** permite definir reglas de transformación para los valores recibidos.

Prueba de item

Attention:
Para realizar la prueba de un item, asegúrese de que la hora del sistema en el server y el proxy esté sincronizada. En caso de que la hora del server esté atrasada, la prueba del item puede devolver el mensaje de error "The task has been expired." Sin embargo, haber configurado distintas zonas horarias en el server y el proxy no afectará el resultado de la prueba.

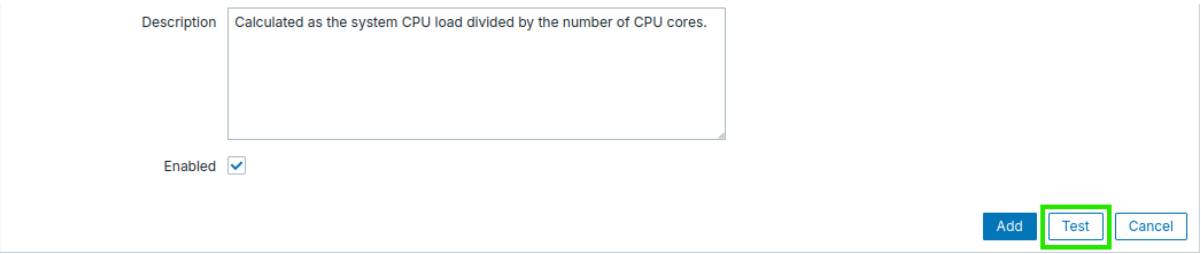
Es posible probar un item y, si está configurado correctamente, obtener un valor real como resultado. La prueba puede realizarse incluso antes de guardar un item.

La prueba está disponible para items de host y template, prototipos de item y reglas de descubrimiento de bajo nivel. La prueba no está disponible para items activos.

La prueba de item está disponible para los siguientes tipos de item pasivos:

- Zabbix agent
- SNMP agent (v1, v2, v3)
- IPMI agent
- SSH checks
- Telnet checks
- JMX agent
- Simple checks (except `icmpping*`, `vmware.*` items)
- Zabbix internal
- Calculated items
- External checks
- Database monitor
- HTTP agent
- Script
- Browser

Para probar un item, haga clic en el botón *Test* en la parte inferior del formulario de configuración del item. Tenga en cuenta que el botón *Test* estará deshabilitado para los items que no pueden probarse (como las comprobaciones activas, los simple checks excluidos).



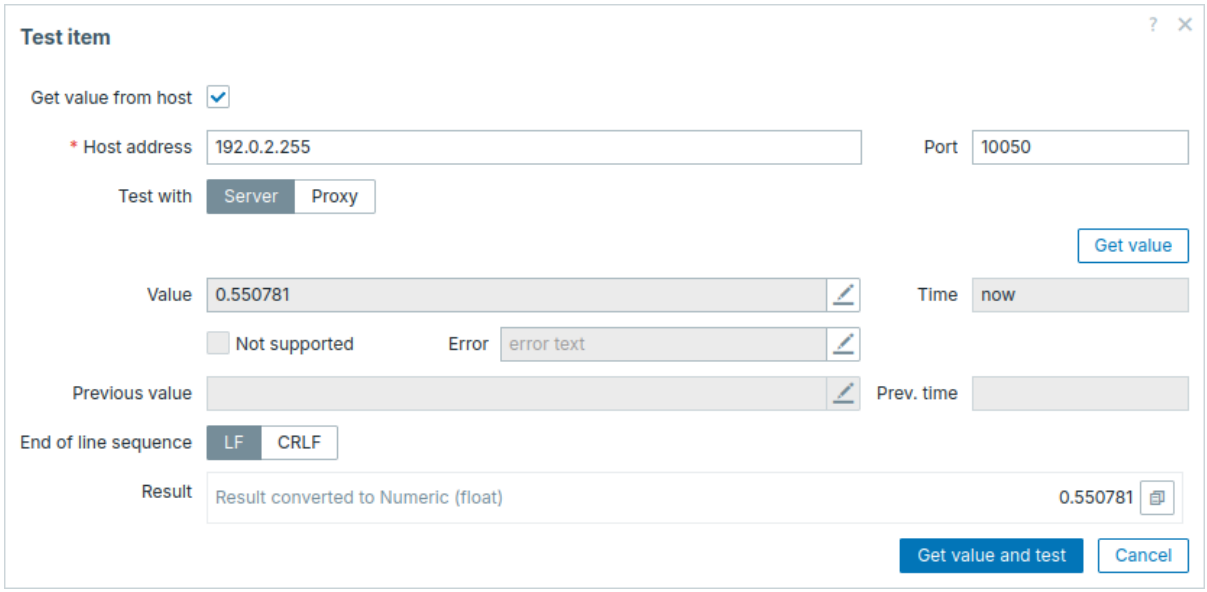
El formulario de prueba del item tiene campos para los parámetros requeridos del host (dirección del host, puerto, prueba con server/proxy (nombre del proxy)) y detalles específicos del item (como la comunidad SNMPv2 o las credenciales de seguridad SNMPv3). Estos campos son context aware:

- Los valores se rellenan previamente cuando es posible; es decir, para items que requieren un agent, tomando la información de la interfaz de agent seleccionada del host.
- Los valores deben rellenarse manualmente para los items de template.
- Los valores de macros en texto plano se resuelven.

- Los campos cuyo valor (o parte del valor) sea un secreto o una macro de Vault quedan vacíos y deben introducirse manualmente. Si algún parámetro del item contiene un valor de macro secreto, se muestra el siguiente mensaje de advertencia: "Item contains user-defined macros with secret values. Values of these macros should be entered manually."
- Los campos se deshabilitan cuando no son necesarios en el contexto del tipo de item (por ejemplo, el campo de dirección del host y el campo de proxy se deshabilitan para los calculated items)

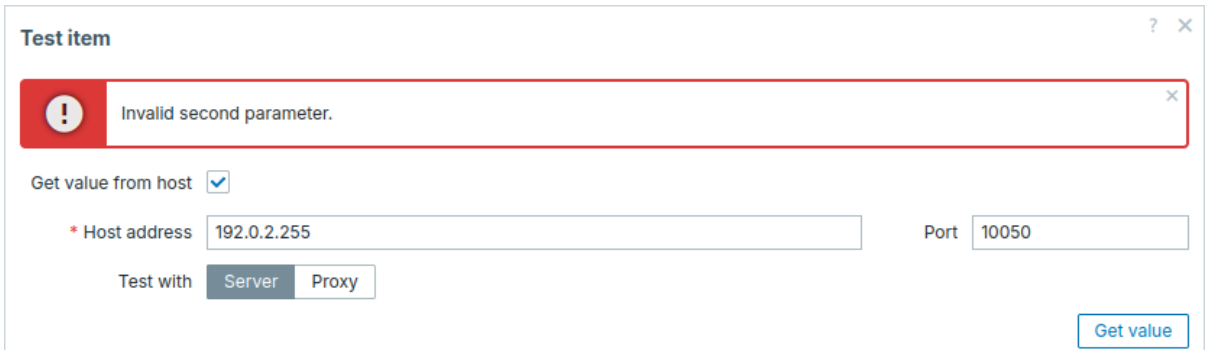
Para probar el item, haga clic en *Get value*. Si el valor se recupera correctamente, se rellenará el campo *Value*, moviendo el valor actual (si existe) al campo *Previous value* y calculando también el campo *Prev. time*, es decir, la diferencia de tiempo entre los dos valores (clics), e intentando detectar una secuencia EOL y cambiar a CRLF si detecta "\n\r" en el valor recuperado.

Haga clic en *Get value and test* para probar el preprocessing.



Los valores recuperados de un host y los resultados de la prueba se truncan a un tamaño máximo de 512 KB cuando se envían al frontend. Si un resultado de prueba se trunca, se muestra un icono de advertencia; pase el cursor sobre él para ver los detalles. Si un valor se trunca, puede fallar la validación del tipo de datos en pruebas consecutivas que comparan valores grandes (por ejemplo, JSON); en tales casos, se muestra un mensaje de error. Tenga en cuenta que Zabbix server sigue procesando completamente los datos de más de 512 KB.

Si la configuración es incorrecta, se muestra un mensaje de error que describe la posible causa.



Un valor recuperado correctamente desde el host también puede utilizarse para probar los pasos de preprocessing.

Botones del formulario

Los botones situados en la parte inferior del formulario permiten realizar varias operaciones.

Add	Añadir un item. Este botón solo está disponible para items nuevos.
Update	Actualizar las propiedades de un item.
Clone	Crear otro item basándose en las propiedades del item actual.
Execute now	Ejecutar inmediatamente una comprobación para obtener un nuevo valor del item. Solo se admite para comprobaciones **pasivas** (consulte más información). *Nota*: al comprobar un valor inmediatamente, la caché de configuración no se actualiza, por lo que el valor no reflejará los cambios más recientes en la configuración del item.
Test	Comprobar si la configuración del item es correcta obteniendo un valor.
Clear history and trends	Eliminar el historial y las tendencias del item.
Delete	Eliminar el item.

Cancel	Cancelar la edición de las propiedades del item.

Límites de datos de item

Los límites de datos de item dependen del tipo de datos del item y del backend de la base de datos.

Los valores numéricos (sin signo) se almacenan tal cual. Si se recibe un valor de punto flotante, solo se almacena la parte entera (por ejemplo, 1.23 → 1).

Los valores numéricos (float) se almacenan tal cual y admiten aproximadamente de 15 a 17 dígitos de precisión, con un rango aproximado de -1.79E+308 a 1.79E+308. También se admiten valores en notación científica (por ejemplo, 1.23E+7, 1e308, 1.1E-4).

Los valores de texto se truncan antes de almacenarse para ajustarse al límite del tipo de valor de la base de datos:

Database	Type of information		
	Character	**Log**	**Text**
MySQL	255 caracteres	65536 bytes	65536 bytes
PostgreSQL	255 caracteres	65536 caracteres	65536 caracteres
SQLite (only Zabbix proxy)	255 caracteres	65536 caracteres	65536 caracteres

Los valores binarios están limitados a 16MiB (16777216 bytes). Los valores que superan este límite se descartan y se muestra un error correspondiente en el frontend.

Los valores con tipo de datos JSON están limitados a 128MiB (134217728 bytes). Los valores que superan este límite se descartan y se muestra un error correspondiente en el frontend. Para MySQL/MariaDB, considere configurar particiones diarias para la tabla `history_json` (usada para almacenar valores JSON), ya que puede crecer rápidamente y bloquear el housekeeper durante un período prolongado. Si su caso de uso implica valores JSON grandes (1MiB o más), consulte cómo configurar su sistema para compatibilidad con valores JSON grandes.

Cuando Zabbix escribe cualquier dato de item en la base de datos, utiliza consultas INSERT y las registra (si `DebugLevel` está establecido en 4 o 5). Las entradas de registro grandes se truncan a 64KB.

Límite de script personalizado

La longitud disponible para scripts personalizados depende de la base de datos utilizada:

Base de datos	Límite en caracteres	Límite en bytes
MySQL	65535	65535
PostgreSQL	65535	sin límite
SQLite (solo proxy Zabbix)	65535	sin límite

Tiempo de espera del item

El tiempo de espera del item especifica cuánto tiempo debe esperar Zabbix antes de abortar la comprobación como fallida.

Cuando se alcanza el tiempo de espera, la comprobación se aborta incluso si la obtención de datos no ha finalizado. Si los datos se reciben parcialmente, el item pasa a estar no compatible y se registra un mensaje de error (por ejemplo, si solo se recopilan correctamente datos de uno de varios OID en una comprobación SNMP).

Para muchos tipos de item, puede establecer tiempos de espera de item **flexibles**:

- por item individual
- por tipo de item (a nivel de proxy)
- por tipo de item (a nivel global global)

Un tiempo de espera personalizado *por item individual* es útil si desea especificar un tiempo de espera más largo para un item concreto, manteniendo bajos los tiempos de espera de otros items.

Precedencia de los tiempos de espera

1. El tiempo de espera de un item individual anula cualquier otro tiempo de espera.
2. Los tiempos de espera a nivel de proxy anulan los tiempos de espera globales.

El tiempo de espera de la configuración del server o del proxy de Zabbix **no** influye en las comprobaciones para las que se han establecido tiempos de espera flexibles.

Compatibilidad con tiempos de espera flexibles

Los tiempos de espera flexibles de los item son compatibles con estos tipos de item:

- Zabbix agent (tanto comprobaciones pasivas como activas)
- SNMP agent (excepto las comprobaciones SNMP heredadas[1])
- Simple check (excepto `icmpping*` y los item de VMware[2])
- SSH agent
- Telnet agent
- External check
- Database monitor
- Script
- HTTP agent
- Browser

[1] Para las comprobaciones SNMP heredadas (sondeo de un solo OID), se aplican los ajustes de tiempo de espera del server o del proxy.
 [2] Para los item `icmpping*`, el valor de tiempo de espera se especifica directamente en la clave del item; además, se aplica un tiempo de espera máximo fijo posible de 600 segundos. Los item de supervisión de VMware usan su propio parámetro VMwareTimeout de la configuración del server o del proxy.

Items no soportados

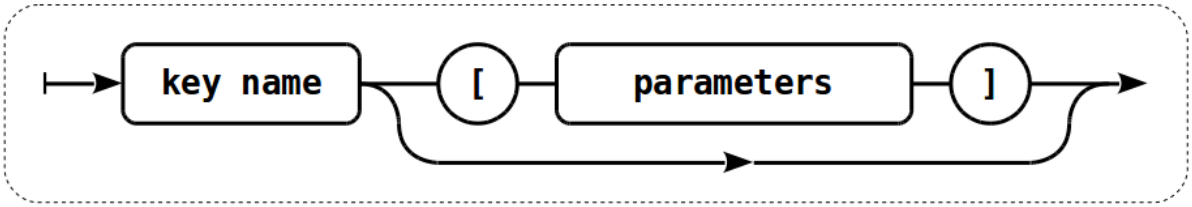
Los items no soportados se informan con el estado *No soportado* y se vuelven a comprobar en su *Intervalo de actualización* estándar.

Un item se vuelve no soportado si su valor no puede recuperarse por alguna razón (por ejemplo, error de conexión, no hay pollers configurados para procesar el item). Además, los items que no reciben datos históricos permanecen en el estado *No soportado*. Este es el comportamiento esperado: un item pasa al estado *Normal* solo cuando recibe nuevos datos históricos válidos.

Al igual que los problemas, los items no soportados solo se reevalúan cuando se reciben nuevos datos, incluso si ya no hay datos históricos disponibles para ese item. En otras palabras, los items y triggers cambian su estado únicamente al recibir nuevos datos. Sin embargo, si una expresión de trigger incluye una función de fecha y hora y/o nodata(), el trigger se recalculará cada 30 segundos (consulte tiempo de cálculo de triggers para más detalles).

Formato de la clave de item

El formato de la clave de item, incluidos los parámetros de la clave, debe seguir las reglas de sintaxis. Las siguientes ilustraciones muestran la sintaxis admitida. Los elementos y caracteres permitidos en cada punto se pueden determinar siguiendo las flechas: si se puede llegar a un bloque a través de la línea, está permitido; si no, no está permitido.



Para construir una clave de item válida, se comienza especificando el nombre de la clave; luego hay una opción de incluir parámetros o no, como se muestra en las dos líneas que se pueden seguir.

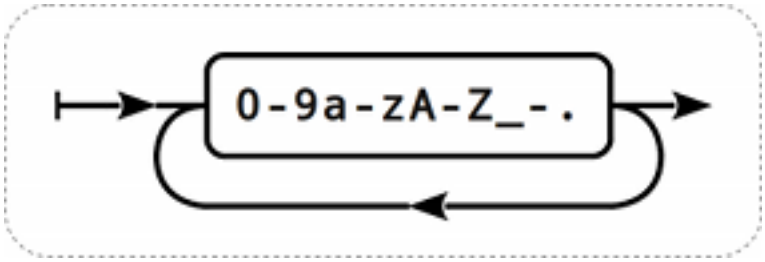
Nombre de clave

El propio nombre de la clave tiene un rango limitado de caracteres permitidos, que simplemente se suceden unos a otros. Los caracteres permitidos son:

`0-9a-zA-Z_-.`

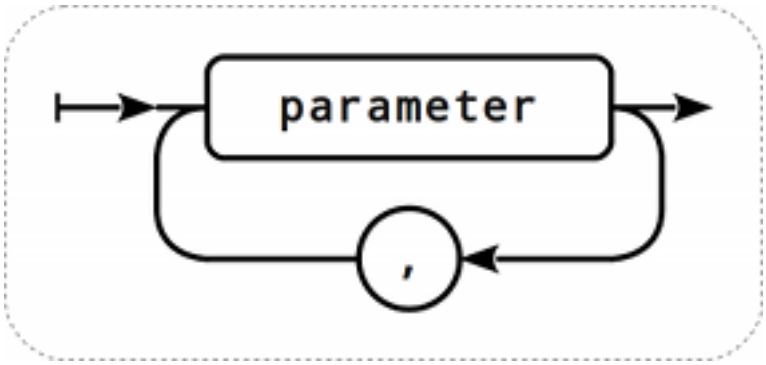
Lo que significa:

- todos los números;
- todas las letras minúsculas;
- todas las letras mayúsculas;
- guion bajo;
- guion;
- punto.

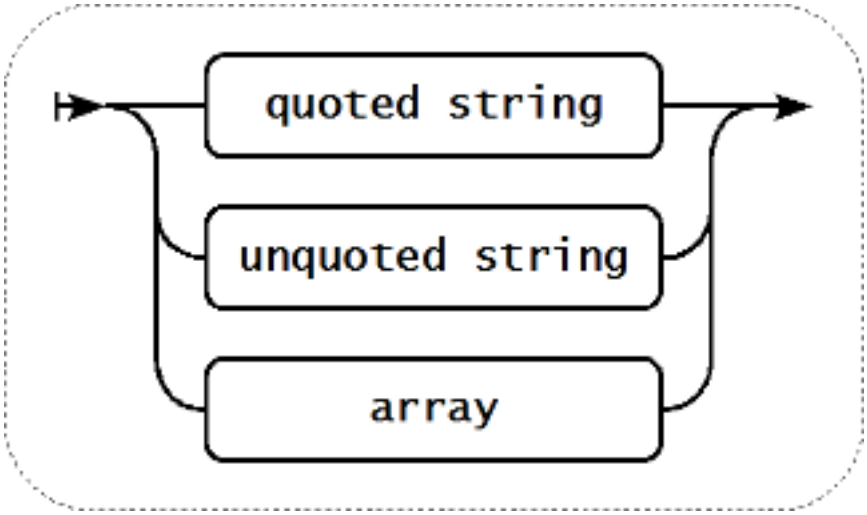


Parámetros clave

Una clave de métrica puede tener múltiples parámetros que se separan por comas.



Cada parámetro de la clave puede ser una cadena entre comillas, una cadena sin comillas o un array.

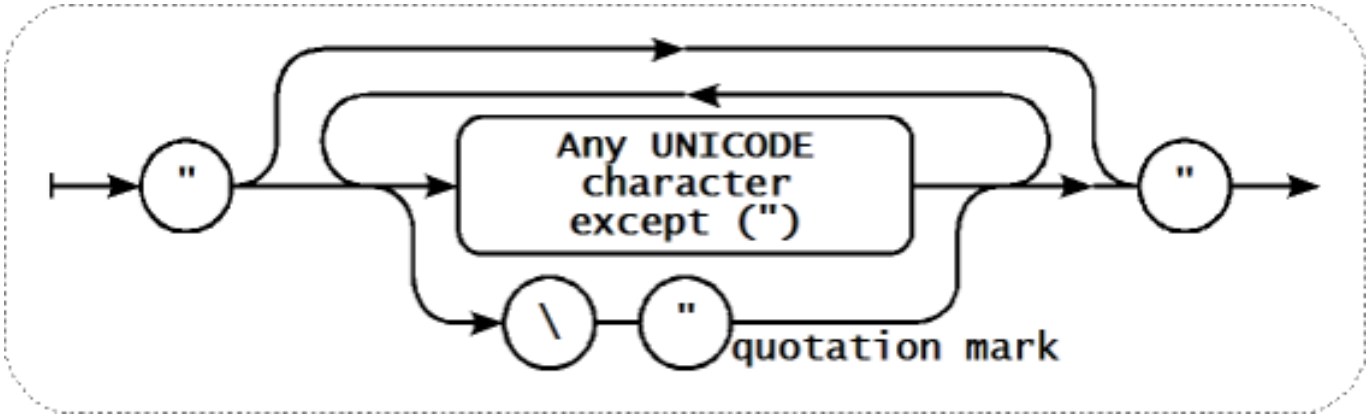


El parámetro también puede dejarse vacío, utilizando así el valor por defecto. En ese caso, se debe añadir el número adecuado de comas si se especifican más parámetros. Por ejemplo, la clave de métrica **icmpping[„200„500]** indicaría que el intervalo entre pings individuales es de 200 milisegundos, el tiempo de espera - 500 milisegundos, y todos los demás parámetros se dejan con sus valores por defecto.

Es posible incluir macros en los parámetros. Estas pueden ser macros de usuario o algunas de las macros integradas. Para ver qué macros integradas se admiten en los parámetros de la clave de métrica, busque en la página Macros soportadas ”parámetros de la clave de métrica”.

Parámetro - cadena entre comillas

Si el parámetro de la clave es una cadena entre comillas, se permite cualquier carácter Unicode. Si la cadena del parámetro de la clave contiene una comilla, este parámetro debe ir entre comillas, y cada comilla que forme parte de la cadena del parámetro debe escaparse con un carácter de barra invertida (\). Si la cadena del parámetro de la clave contiene una coma, este parámetro debe ir entre comillas.

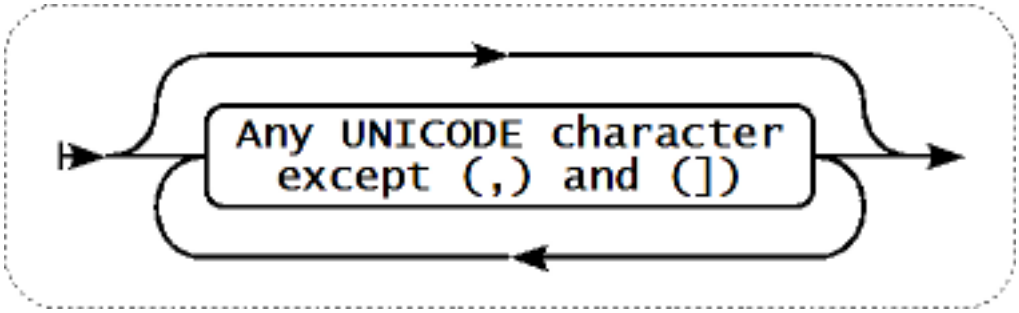


Warning:
Para entrecomillar los parámetros de la clave de la métrica, utilice únicamente comillas dobles. No se admiten comillas simples.

Attention:
No se permiten matrices de parámetros multinivel, por ejemplo, `[a,[b,[c,d]],e]`.

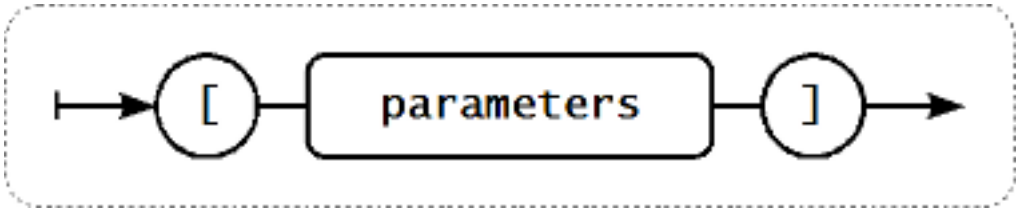
Parámetro - cadena sin comillas

Si el parámetro de la clave es una cadena sin comillas, se permite cualquier carácter Unicode excepto la coma y el corchete derecho (]). El parámetro sin comillas no puede comenzar con un corchete izquierdo ([).



Parámetro - array

Si el parámetro de la clave es un array, se encierra nuevamente entre corchetes, donde los parámetros individuales se ajustan a las reglas y sintaxis de especificar múltiples parámetros.



Intervalos personalizados

Descripción general

Es posible crear reglas personalizadas sobre los momentos en que se comprueba un item.
Los dos métodos para ello son los *intervalos flexibles*, que permiten redefinir el intervalo de actualización predeterminado, y la *programación*, mediante la cual una comprobación del item puede ejecutarse en un momento específico o en una secuencia de momentos.

Intervalos flexibles

Los intervalos flexibles permiten redefinir el intervalo de actualización predeterminado para períodos específicos.

Un intervalo flexible se define mediante *Interval* y *Period*, donde:

- *Interval* - el intervalo de actualización para el período especificado. Se admiten sufijos de tiempo, por ejemplo, 30s, 1m, 2h, 1d.

- *Period* - el período durante el cual está activo el intervalo flexible (consulte los períodos de tiempo para obtener una descripción detallada del formato de *Period*)

Si se superponen varios intervalos flexibles, se utiliza el valor más pequeño de *Interval* para el período superpuesto. Tenga en cuenta que, si el valor más pequeño de los intervalos flexibles superpuestos es '0', no se realizará ningún sondeo. Fuera de los intervalos flexibles, se utiliza el intervalo de actualización predeterminado.

Tenga en cuenta que, si el intervalo flexible es igual a la duración del período, el item se comprobará exactamente una vez. Si el intervalo flexible es mayor que el período, el item podría comprobarse una vez o no comprobarse en absoluto (por lo que no se recomienda esta configuración). Si el intervalo flexible es menor que el período, el item se comprobará al menos una vez.

Si el intervalo flexible se establece en '0', no se sondeará el item durante el período del intervalo flexible y el sondeo se reanudará según el *Update interval* predeterminado una vez finalizado el período. Ejemplos:

Interval	Period	Description
10	1-5,09:00-18:00	El item se comprobará cada 10 segundos durante el horario laboral.
0	1-7,00:00-7:00	El item no se comprobará durante la noche.
0	7-7,00:00-24:00	El item no se comprobará los domingos.
60	1-7,12:00-12:01	El item se comprobará a las 12:00 todos los días. Tenga en cuenta que esto se utilizaba como solución alternativa para las comprobaciones programadas y se recomienda utilizar intervalos de programación para este tipo de comprobaciones.

Intervalos de planificación

Los intervalos de planificación se utilizan para comprobar items en momentos específicos. Mientras que los intervalos flexibles están diseñados para redefinir el intervalo predeterminado de actualización del item, los intervalos de planificación se utilizan para especificar una programación de comprobación independiente, que se ejecuta en paralelo.

Un intervalo de planificación se define como: `md<filter>wd<filter>h<filter>m<filter>s<filter>` donde:

- **md** - días del mes
- **wd** - días de la semana
- **h** - horas
- **m** - minutos
- **s** - segundos

`<filter>` se utiliza para especificar valores para su prefijo (días, horas, minutos, segundos) y se define como: `[<from>[-<to>]][/<step>]` donde:

- `<from>` y `<to>` definen el rango de valores coincidentes (incluidos). Si se omite `<to>`, entonces el filtro coincide con un rango `<from> - <from>`. Si también se omite `<from>`, entonces el filtro coincide con todos los valores posibles.
- `<step>` define los saltos del valor numérico dentro del rango. De forma predeterminada, `<step>` tiene el valor 1, lo que significa que coinciden todos los valores del rango definido.

Aunque las definiciones de filtro son opcionales, debe utilizarse al menos un filtro. Un filtro debe tener un rango o el valor de *<step>* definido.

Un filtro vacío coincide con '0' si no se define ningún filtro de nivel inferior, o con todos los valores posibles en caso contrario. Por ejemplo, si se omite el filtro de horas, entonces solo coincidirá la hora '0', siempre que también se omitan los filtros de minutos y segundos; de lo contrario, un filtro de horas vacío coincidirá con todos los valores de hora.

Los valores válidos de `<from>` y `<to>` para su respectivo prefijo de filtro son:

Prefix	Description	*<from>*	*<to>*
md	Días del mes	1-31	1-31
wd	Días de la semana	1-7	1-7
h	Horas	0-23	0-23
m	Minutos	0-59	0-59
s	Segundos	0-59	0-59

El valor `<from>` debe ser menor o igual que el valor `<to>`. El valor `<step>` debe ser mayor o igual que 1 y menor o igual que `<to> - <from>`.

Los valores de un solo dígito para días del mes, horas, minutos y segundos pueden llevar el prefijo 0. Por ejemplo, `md01-31` y `h/02` son intervalos válidos, pero `md01-031` y `wd01-07` no lo son.

En el frontend de Zabbix, varios intervalos de planificación se introducen en filas separadas. En la API de Zabbix, se concatenan en una sola cadena con un punto y coma `;` como separador.

Si una hora coincide con varios intervalos, se ejecuta solo una vez. Por ejemplo, `wd1h9;h9` se ejecutará solo una vez el lunes a las 9:00.

Ejemplos:

Interval	Will be executed
m0-59	cada minuto
h9-17/2	cada 2 horas comenzando a las 9:00 (9:00, 11:00 ...)
m0,30 or m/30	cada hora a hh:00 y hh:30
m0,5,10,15,20,25,30,35,40,45,50,55 or m/5	cada cinco minutos
wd1-5h9	cada lunes a viernes a las 9:00
wd1-5h9-18	cada lunes a viernes a las 9:00,10:00,...,18:00
h9,10,11 or h9-11	cada día a las 9:00, 10:00 y 11:00
md1h9m30	cada día 1 de cada mes a las 9:30
md1wd1h9m30	cada día 1 de cada mes a las 9:30 si es lunes
h9m/30	cada día a las 9:00, 9:30
h9m0-59/30	cada día a las 9:00, 9:30
h9,10m/30	cada día a las 9:00, 9:30, 10:00, 10:30
h9-10m30	cada día a las 9:30, 10:30
h9m10-40/30	cada día a las 9:10, 9:40
h9,10m10-40/30	cada día a las 9:10, 9:40, 10:10, 10:40
h9-10m10-40/30	cada día a las 9:10, 9:40, 10:10, 10:40
h9m10-40	cada día a las 9:10, 9:11, 9:12, ... 9:40
h9m10-40/1	cada día a las 9:10, 9:11, 9:12, ... 9:40
h9-12,15	cada día a las 9:00, 10:00, 11:00, 12:00, 15:00
h9-12,15m0	cada día a las 9:00, 10:00, 11:00, 12:00, 15:00
h9-12,15m0s30	cada día a las 9:00:30, 10:00:30, 11:00:30, 12:00:30, 15:00:30
h9-12s30	cada día a las 9:00:30, 9:01:30, 9:02:30 ... 12:58:30, 12:59:30
h9m/30;h10 (*sintaxis específica de la API*)	cada día a las 9:00, 9:30, 10:00
h9m/30 h10 (*añada esto como otra fila en el frontend*)	cada día a las 9:00, 9:30, 10:00

Alineación de zonas horarias para proxies y agentes

Tenga en cuenta que los proxies y agentes de Zabbix utilizan sus zonas horarias locales al procesar los intervalos de programación.

Por esta razón, cuando los intervalos de programación se aplican a elementos monitorizados por un proxy de Zabbix o a elementos activos de un agente, se recomienda establecer la zona horaria de los respectivos proxies o agentes igual que la del servidor Zabbix; de lo contrario, la cola puede informar retrasos en los elementos de forma incorrecta.

La zona horaria para el proxy o agente de Zabbix se puede establecer utilizando la variable de entorno `TZ` en el archivo de unidad `systemd`:

```
[Service]
...
Environment="TZ=Europe/Amsterdam"
```

Preprocesamiento del valor del item

Resumen

El preprocesamiento le permite aplicar transformaciones a los valores de item recibidos antes de guardarlos en la base de datos. Estas transformaciones/pasos de preprocesamiento son realizados por el server o el proxy de Zabbix (si los items son monitorizados por proxy).

Esta función admite diversos casos de uso, como:

- convertir bytes a bits (por ejemplo, multiplicando los valores de tráfico de red por ”8”);
- calcular estadísticas por segundo para valores que aumentan de forma incremental;

- aplicar expresiones regulares para extraer o modificar valores;
- ejecutar scripts personalizados sobre los valores;
- descartar valores sin cambios para optimizar el almacenamiento en la base de datos.

Se puede configurar uno o varios pasos de preprocesamiento para un item. Estos pasos se ejecutan en el orden en que están configurados.

Attention:
Si falla un paso de preprocesamiento, un item pasa a ser no soportado. Esto puede evitarse mediante el manejo de errores *Custom on fail* (disponible para la mayoría de las transformaciones), que permite descartar valores o establecer valores personalizados.

 Para los items de log, los metadatos del log (sin valor) siempre restablecerán el estado de no soportado del item, haciendo que vuelva a estar soportado. Esto sucede incluso si el error inicial ocurrió después de recibir un valor de log desde agent.

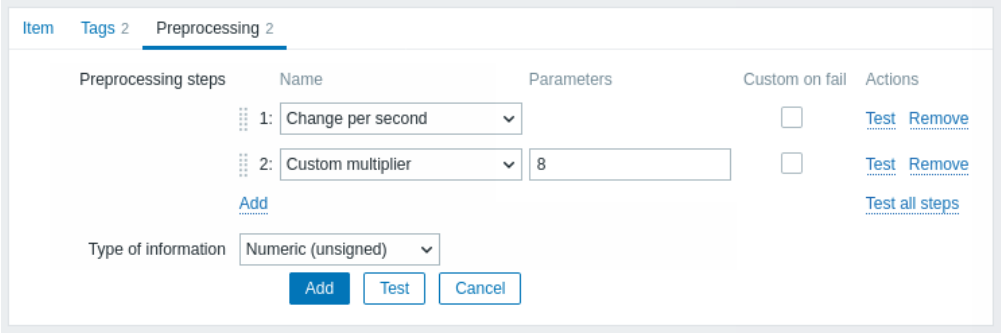
Todos los valores pasados al preprocesamiento se tratan inicialmente como cadenas. La conversión al tipo de valor deseado (tal como se define en la configuración del item) se realiza al final de la canalización de preprocesamiento. Sin embargo, algunos pasos específicos de preprocesamiento pueden activar conversiones antes, si es necesario. Para obtener información técnica detallada, consulte Detalles del preprocesamiento.

Para asegurarse de que su configuración de preprocesamiento funcione como se espera, puede probarla.

Vea también: Ejemplos de preprocesamiento

Configuración

Los pasos de preprocesamiento se definen en la pestaña **Preprocesamiento** del formulario de configuración del item.



Haga clic en *Añadir* para seleccionar una transformación compatible.

El campo *Tipo de información* se muestra en la parte inferior de la pestaña cuando se define al menos un paso de preprocesamiento. Si es necesario, es posible cambiar el tipo de información sin salir de la pestaña *Preprocesamiento*. Consulte Creación de un item para obtener una descripción detallada de los parámetros.

Transformaciones compatibles

Todas las transformaciones compatibles se enumeran a continuación.
Haga clic en el nombre de una transformación para consultar todos sus detalles.

Nombre	Descripción	Tipo
Expresión regular	Compara el valor con la expresión regular y lo reemplaza por el resultado requerido.	Texto
Reemplazar	Busca la cadena de búsqueda y la reemplaza por otra cadena (o por nada).	
Recortar	Elimina los caracteres especificados del principio y del final del valor.	
Recortar por la derecha	Elimina los caracteres especificados del final del valor.	
Recortar por la izquierda	Elimina los caracteres especificados del principio del valor.	
XML XPath	Extrae un valor o fragmento de datos XML mediante la funcionalidad XPath.	Datos estructurados
Ruta JSON	Extrae un valor o fragmento de JSON mediante la funcionalidad JSONPath.	
CSV a JSON	Convierte los datos de un archivo CSV al formato JSON.	
XML a JSON	Convierte los datos en formato XML a JSON.	
Valor de SNMP walk	Extrae un valor mediante el nombre de OID/MIB especificado y aplica opciones de formato.	SNMP
SNMP walk a JSON	Convierte los valores SNMP a JSON.	
Obtener valor de SNMP	Aplica opciones de formato al valor obtenido mediante SNMP.	

Nombre	Descripción	Tipo
Multiplicador personalizado	Multiplica el valor por el valor entero o de coma flotante especificado.	Aritmética
Cambio simple	Calcula la diferencia entre el valor actual y el anterior.	Cambio
Cambio por segundo	Calcula la velocidad de cambio del valor (la diferencia entre el valor actual y el anterior) por segundo.	
Booleano a decimal	Convierte el valor del formato booleano al formato decimal.	Sistemas numéricos
Octal a decimal	Convierte el valor del formato octal al formato decimal.	
Hexadecimal a decimal	Convierte el valor del formato hexadecimal al formato decimal.	
JavaScript	Introduce código JavaScript.	Scripts personalizados
En rango	Define un rango en el que debe encontrarse el valor.	Validación
Coincide con una expresión regular	Especifica una expresión regular con la que debe coincidir el valor.	
No coincide con una expresión regular	Especifica una expresión regular con la que no debe coincidir el valor.	
Comprobar si hay un error en JSON	Comprueba si hay un mensaje de error a nivel de aplicación ubicado en JSONPath.	
Comprobar si hay un error en XML	Comprueba si hay un mensaje de error a nivel de aplicación ubicado en XPath.	
Comprobar si hay un error mediante una expresión regular	Comprueba si hay un mensaje de error a nivel de aplicación mediante una expresión regular.	
Comprobar si hay un valor no compatible	Comprueba si no se pudo recuperar ningún valor del item.	
Descartar si no ha cambiado	Descarta un valor si no ha cambiado.	Limitación
Descartar si no ha cambiado con latido	Descarta un valor si no ha cambiado durante el período definido.	
Patrón de Prometheus	Utiliza la siguiente consulta para extraer los datos requeridos de las métricas de Prometheus.	Prometheus
Prometheus a JSON	Convierte las métricas de Prometheus requeridas a JSON.	

Si la casilla *Personalizado en caso de error* está habilitada para la transformación seleccionada, puede marcarla y especificar opciones personalizadas para el manejo de errores: *Descartar valor* (descartar el valor), *Establecer valor en* (establecer un valor especificado) o *Establecer error en* (establecer un mensaje de error especificado).
Si falla un paso de preprocesamiento, el item no pasará a ser no compatible cuando se seleccione la opción *Descartar valor* o *Establecer valor en*.

Tenga en cuenta que, para los pasos de preprocesamiento *Cambio* y *Limitación*, Zabbix debe recordar el último valor para calcular o comparar el nuevo valor según sea necesario.
Estos valores anteriores son gestionados por el administrador de preprocesamiento.
Si se reinicia Zabbix server o proxy, o se realiza cualquier cambio en los pasos de preprocesamiento, se restablece el último valor del item correspondiente, lo que da como resultado:

- para los pasos *Cambio simple* y *Cambio por segundo*: se ignorará el siguiente valor porque no existe un valor anterior a partir del cual calcular el cambio;
- para los pasos *Descartar si no ha cambiado* y *Descartar si no ha cambiado con latido*: el siguiente valor nunca se descartará, aunque debería haberse descartado según las reglas de descarte.

Expresión regular

Compara el valor con la expresión regular y lo reemplaza por el resultado requerido.

Parámetros:

- **pattern** - la expresión regular;
- **output** - la plantilla de formato de salida. Una secuencia de escape \N (donde N=1…9) se reemplaza por el grupo coincidente N. Una secuencia de escape \0 se reemplaza por el texto coincidente.

Comentarios:

- Si no se encuentra ninguna coincidencia con el valor de entrada, el item dejará de ser compatible;
- La expresión regular admite la extracción de un máximo de 10 grupos capturados mediante la secuencia \N;
- Consulte la sección expresiones regulares para ver algunos ejemplos existentes.

Reemplazar

Busca la cadena de búsqueda y la reemplaza por otra cadena (o por nada).

Parámetros:

- **search string** - cadena que se debe buscar y reemplazar, distingue entre mayúsculas y minúsculas (obligatorio);
- **replacement** - cadena con la que se reemplazará la cadena de búsqueda. La cadena de reemplazo también puede estar vacía, lo que permite eliminar la cadena de búsqueda cuando se encuentra.

Comentarios:

- Se reemplazarán todas las apariciones de la cadena de búsqueda;
- Es posible utilizar secuencias de escape para buscar o reemplazar saltos de línea, retornos de carro, tabulaciones y espacios ”\n \r \t \s”; la barra inversa se puede escapar como ”\\” y las secuencias de escape se pueden escapar como ”\\n”;
- El escape de los saltos de línea, retornos de carro y tabulaciones se realiza automáticamente durante el descubrimiento de bajo nivel.

Recortar

Elimina los caracteres especificados del principio y final del valor.

Recorte derecho

Elimina los caracteres especificados del final del valor.

Recorte izquierdo

Elimina los caracteres especificados del principio del valor.

XML XPath

Extrae un valor o fragmento de datos XML mediante la funcionalidad XPath.

Comentarios:

- Para que esta opción funcione, el server de Zabbix (o el proxy de Zabbix) debe estar compilado con compatibilidad con libxml;
- No se admiten espacios de nombres.

Ejemplos:

```
number(/document/item/value) #extraerá '10' de <document><item><value>10</value></item></document>
number(/document/item/@attribute) #extraerá '10' de <document><item attribute="10"></item></document>
/document/item #extraerá '<item><value>10</value></item>' de <document><item><value>10</value></item></doc
```

JSON Path

Extrae un valor o fragmento de datos JSON mediante la funcionalidad JSONPath.

CSV a JSON

Convierta los datos de un archivo CSV al formato JSON.

Para más información, consulte: Preprocesamiento de CSV a JSON.

XML a JSON

Convierte los datos del formato XML a JSON.

Para obtener más información, consulta: Reglas de serialización.

Valor de SNMP walk

Extrae el valor mediante el OID/nombre MIB especificado y aplica las opciones de formato:

- **Sin cambios**: devuelve la cadena hexadecimal como una cadena hexadecimal sin escape (*nota*: las sugerencias de visualización se siguen aplicando);
- **UTF-8 desde hex-STRING**: convierte la cadena hexadecimal en una cadena UTF-8;
- **MAC desde hex-STRING**: valida la cadena hexadecimal como una dirección MAC y devuelve una cadena de dirección MAC adecuada (donde `' '` se reemplaza por `':'`);
- **Entero desde BITS**: convierte los primeros 8 bytes de una cadena de bits expresada como una secuencia de caracteres hexadecimales (por ejemplo, ”1A 2B 3C 4D”) en un entero sin signo de 64 bits. En las cadenas de bits de más de 8 bytes, los bytes posteriores se ignorarán.

SNMP walk a JSON

Convierte los valores SNMP a JSON.

Especifique un nombre de campo en el JSON y la ruta OID SNMP correspondiente.
Los valores de los campos se completarán con los valores de la ruta OID SNMP especificada.

Comentarios:

- Hay disponibles opciones de formato de valores similares a las del paso *Valor de SNMP walk*;
- Puede utilizar este paso de preprocesamiento para el descubrimiento de OID SNMP.

Obtener valor SNMP

Aplicar opciones de formato al valor obtenido mediante SNMP:

- **UTF-8 desde Hex-STRING**: convierte la cadena hexadecimal en una cadena UTF-8;
- **MAC desde Hex-STRING**: valida la cadena hexadecimal como una dirección MAC y devuelve una cadena de dirección MAC válida (donde `' '` se reemplaza por `':'`);
- **Entero desde BITS**: convierte los primeros 8 bytes de una cadena de bits expresada como una secuencia de caracteres hexadecimales (por ejemplo, ”1A 2B 3C 4D”) en un entero sin signo de 64 bits. En las cadenas de bits de más de 8 bytes, los bytes posteriores se ignorarán.

Multiplicador personalizado

Multiplica el valor por el valor entero o de coma flotante especificado.

Comentarios:

- Utilice esta opción para convertir los valores recibidos en KB, MBps, etc., a B, Bps. De lo contrario, Zabbix no podrá establecer correctamente los prefijos (K, M, G, etc.).
- Tenga en cuenta que, si el tipo de información del item es *Numeric (unsigned)*, los valores entrantes con una parte fraccionaria se truncarán (es decir, ’0.9’ se convertirá en ’0’) antes de aplicar el multiplicador personalizado;
- Si utiliza un multiplicador personalizado o almacena el valor como *Change per second* para items cuyo tipo de información está establecido en *Numeric (unsigned)* y el valor calculado resultante es en realidad un número de coma flotante, el valor calculado seguirá aceptándose como correcto al truncar la parte decimal y almacenar el valor como un entero;
- Compatible con: notación científica, por ejemplo, 1e+70; macros de usuario y macros de LLD; cadenas que incluyen macros, por ejemplo, `{#MACRO}e+10`, `{$MACRO1}e+{$MACRO2}`. Las macros deben resolverse en un número entero o de coma flotante.

Cambio simple

Calcula la diferencia entre el valor actual y el anterior.

Comentarios:

- Este paso puede ser útil para medir un valor que aumenta constantemente;
- Se evalúa como **value**-**prev_value**, donde *value* es el valor actual y *prev_value* es el valor recibido anteriormente;
- Solo se permite una operación de cambio por item (”Cambio simple” o ”Cambio por segundo”);
- Si el valor actual es menor que el valor anterior, Zabbix descarta esa diferencia (no almacena nada) y espera otro valor.

Cambio por segundo

Calcula la velocidad de cambio del valor (la diferencia entre el valor actual y el anterior) por segundo.

Comentarios:

- Este paso resulta útil para calcular la velocidad por segundo de un valor que aumenta constantemente;
- Como este cálculo puede producir números de punto flotante, se recomienda establecer el ’Tipo de información’ en *Numérico (flotante)*, incluso si los valores sin procesar recibidos son enteros. Esto es especialmente relevante para números pequeños en los que la parte decimal es importante. Si los valores de punto flotante son grandes y pueden superar la longitud del campo ’float’, en cuyo caso se puede perder el valor completo, se recomienda utilizar *Numérico (sin signo)* y recortar únicamente la parte decimal;
- Se evalúa como (**value**-**prev_value**)/(**time**-**prev_time**), donde *value* es el valor actual; *prev_value* es el valor recibido anteriormente; *time* es la marca de tiempo actual; *prev_time* es la marca de tiempo del valor anterior;
- Solo se permite una operación de cambio por item (”Cambio simple” o ”Cambio por segundo”);
- Si el valor actual es menor que el anterior, Zabbix descarta esa diferencia (no almacena nada) y espera otro valor. Esto permite trabajar correctamente, por ejemplo, con el desbordamiento de contadores SNMP de 32 bits.

Booleano a decimal

Convierte el valor del formato booleano a decimal.

Comentarios:

- La representación textual se convierte en 0 o 1. Por lo tanto, 'TRUE' se almacena como 1 y 'FALSE' se almacena como 0. Todos los valores se comparan sin distinguir entre mayúsculas y minúsculas. Actualmente, los valores reconocidos son, para *TRUE*: true, t, yes, y, on, up, running, enabled, available, ok, master; para *FALSE*: false, f, no, n, off, down, unused, disabled, unavailable, err, slave. Además, cualquier valor numérico distinto de cero se considera TRUE y cero se considera FALSE.

Octal a decimal

Convierte el valor del formato octal a decimal.

Hexadecimal a decimal

Convierte el valor del formato hexadecimal a decimal.

JavaScript

Introduzca el código JavaScript en el editor modal que se abre al hacer clic en el campo del parámetro o en el icono de lápiz junto a él.

> **Attention:**
> No utilice asignaciones no declaradas en el JavaScript de preprocesamiento. Utilice `var` para declarar variables locales.

Comentarios:

- La longitud de JavaScript disponible depende de la base de datos utilizada;
- Para más información, consulte: Preprocesamiento Javascript.

En el rango

Defina un rango en el que debe encontrarse un valor especificando los valores mínimo y máximo (inclusive).

Comentarios:

- Se aceptan valores numéricos (incluido cualquier número de dígitos, una parte decimal opcional y una parte exponencial opcional, así como valores negativos);
- El valor mínimo debe ser menor que el máximo;
- Debe existir al menos un valor;
- Se pueden utilizar macros de usuario y macros de descubrimiento de bajo nivel.

Coincide con la expresión regular

Especifique una expresión regular con la que debe coincidir un valor.

No coincide con la expresión regular

Especifique una expresión regular con la que el valor no debe coincidir.

Comprobar si hay un error en JSON

Comprueba si hay un mensaje de error a nivel de aplicación ubicado en JSONPath.
Detén el procesamiento si la comprobación se realiza correctamente y el mensaje no está vacío; de lo contrario, continúa el procesamiento con el valor que tenía antes de este paso de preprocesamiento.

Comentarios:

- Estos errores del servicio externo se muestran al usuario tal cual, sin añadir información sobre el paso de preprocesamiento;
- No se notificará ningún error si no se puede analizar un JSON no válido.

Comprobar si hay un error en XML

Comprueba si hay un mensaje de error a nivel de aplicación ubicado en XPath.
Detén el procesamiento si la comprobación se realiza correctamente y el mensaje no está vacío; de lo contrario, continúa el procesamiento con el valor que tenía antes de este paso de preprocesamiento.

Comentarios:

- Estos errores del servicio externo se muestran al usuario tal cual, sin añadir información sobre el paso de preprocesamiento;
- No se notificará ningún error si no se puede analizar XML no válido.

Comprobar si hay un error mediante una expresión regular

Comprueba si hay un mensaje de error a nivel de aplicación mediante una expresión regular.
Detiene el procesamiento si tiene éxito y el mensaje no está vacío; de lo contrario, continúa el procesamiento con el valor que había antes de este paso de preprocesamiento.

Parámetros:

- **pattern**: la expresión regular;
- **output**: la plantilla de formato de salida.
  Una secuencia de escape \N (donde N=1…9) se sustituye por el grupo coincidente N.
  Una secuencia de escape \0 se sustituye por el texto coincidente.

Comentarios:

- Estos errores del servicio externo se muestran al usuario tal cual, sin añadir información sobre el paso de preprocesamiento.

Comprobar si el valor no es compatible

Comprueba si no se pudo recuperar ningún valor del item. Especifique cómo debe procesarse el error, según el análisis del mensaje de error devuelto.

Parámetros:

- **scope**: seleccione el ámbito del procesamiento de errores:
any error: cualquier error;
error matches: solo el error que coincida con la expresión regular especificada en *pattern*;
error does not match: solo el error que no coincida con la expresión regular especificada en *pattern*
- **pattern**: la expresión regular con la que se comparará el error. Si se selecciona *any error* en el parámetro de ámbito, este campo no se muestra. Si se muestra, este campo es obligatorio.

Comentarios:

- Normalmente, la ausencia de un valor o la imposibilidad de recuperarlo haría que el item dejara de ser compatible. Este paso de preprocesamiento permite modificar este comportamiento.
- Este paso de preprocesamiento solo comprueba si no se pudo recuperar ningún valor del item. No comprueba, por ejemplo, si el tipo del valor recuperado (p. ej., una cadena) coincide con el tipo de información del item (p. ej., numérico); para obtener más información, consulte Ejemplos de preprocesamiento. Si hay una discrepancia de tipos, el item aún puede dejar de ser compatible después de ejecutar todos los pasos de preprocesamiento. Para comprobar si hay una discrepancia de tipos, puede utilizar, por ejemplo, el paso de preprocesamiento *Custom multiplier*; consulte Ejemplos de preprocesamiento.
- Se admiten grupos de expresiones regulares de captura en los campos *Set value to* o *Set error to*. Use \N (donde N=1…9) para recuperar el grupo coincidente N; use \0 para recuperar el texto coincidente;
- Estos pasos siempre se ejecutan como los primeros pasos de preprocesamiento y se colocan por encima de todos los demás después de guardar los cambios en el item;
- Se admiten varios pasos de *Check for not supported value*, en el orden especificado. Un paso para *any error* se colocará automáticamente como el último paso de este grupo.

Descartar sin cambios

Descartar un valor si no ha cambiado.

Comentarios:

- Si se descarta un valor, no se guarda en la base de datos y el servidor Zabbix no tiene conocimiento de que se recibió ese valor. No se evaluarán expresiones de disparadores, por lo tanto, no se crearán/solucionarán problemas para los disparadores relacionados. Las funciones solo funcionarán en base a los datos que realmente se guarden en la base de datos. Como las tendencias se construyen en base a los datos de la base de datos, si no se guarda ningún valor durante una hora, tampoco habrá datos de tendencias para esa hora.
- Solo se puede especificar una opción de limitación por elemento.

Descartar valores sin cambios con latido

Descarta un valor si no ha cambiado durante el período definido (en segundos).

Comentarios:

- Se admiten valores enteros positivos para especificar los segundos (mínimo: 1 segundo);
- Se pueden utilizar sufijos de tiempo (por ejemplo, `30s`, `1m`, `2h`, `1d`);
- Se pueden utilizar macros de usuario y macros de descubrimiento de bajo nivel;
- Si se descarta un valor, no se guarda en la base de datos y el server de Zabbix no tiene conocimiento de que se haya recibido este valor. No se evaluarán expresiones de trigger y, como resultado, no se crearán ni resolverán problemas para los triggers relacionados. Las funciones solo funcionarán en función de los datos que realmente se hayan guardado en la base de datos. Dado que las tendencias se generan a partir de los datos de la base de datos, si no se guarda ningún valor durante una hora, tampoco habrá datos de tendencias para esa hora.
- Solo se puede especificar una opción de limitación por item.

Patrón de Prometheus

Use la siguiente consulta para extraer los datos requeridos de las métricas de Prometheus.

Consulte Prometheus checks para obtener más detalles.

Prometheus a JSON

Convierte las métricas de Prometheus requeridas a JSON.

Consulta Prometheus checks para más detalles.

Soporte de macros

Se admiten macros de usuario y macros de usuario con contexto en:

- parámetros de pasos de preprocesamiento, incluido el código JavaScript;
- parámetros personalizados de manejo de errores (campos *Establecer valor en* y *Establecer error en*).

Note:
El contexto de la macro se ignora cuando una macro se reemplaza por su valor. El valor de la macro se inserta en el código tal cual, no es posible agregar un escape adicional antes de colocar el valor en el código JavaScript. Tenga en cuenta que esto puede causar errores de JavaScript en algunos casos.

Pruebas

Consulte pruebas de preprocesamiento.

Pruebas de preprocesamiento

Pruebas

Probar los pasos de preprocesamiento es útil para asegurarse de que las canalizaciones de preprocesamiento complejas produzcan los resultados esperados, sin esperar a que se reciba y se procese el valor del item.

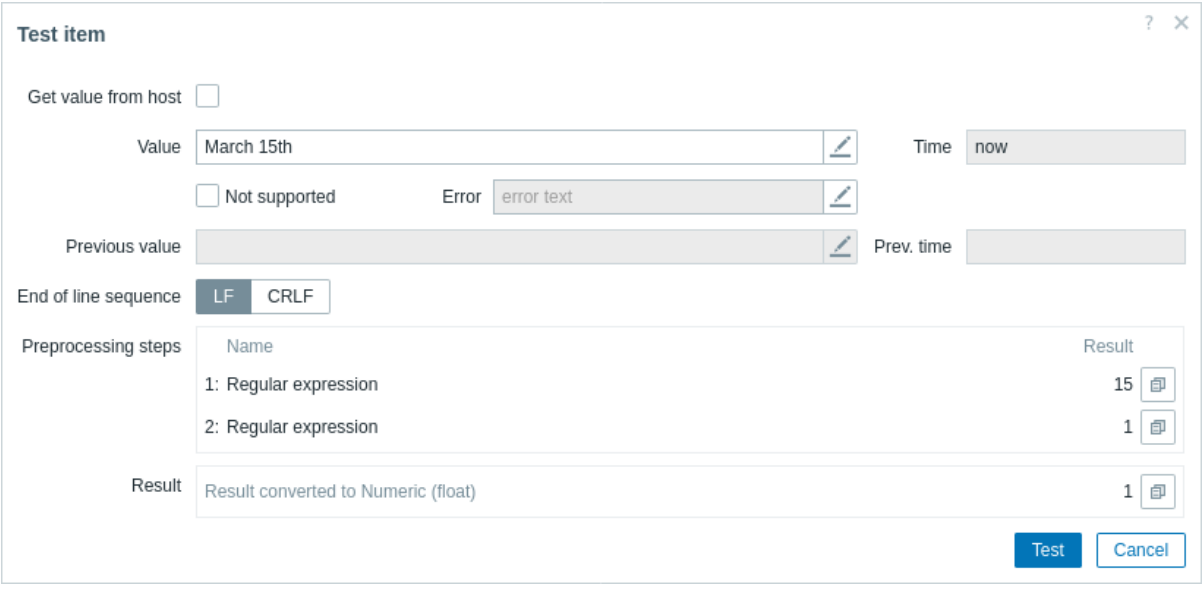


Es posible probar:

- con un valor hipotético
- con un valor real de un host

Cada paso de preprocesamiento puede probarse de forma individual, así como todos los pasos pueden probarse juntos. Cuando hace clic en el botón *Test* o *Test all steps*, respectivamente, en el bloque Actions, se abre una ventana de prueba.

Prueba de valor hipotético



Parameter	Description
Get value from host	Si desea probar un valor hipotético, deje esta casilla sin marcar. Véase también: Prueba de valor real.
Value	Introduzca el valor de entrada que desea probar. Al hacer clic en el campo del parámetro o en el botón de vista/edición  se abrirá una ventana de área de texto para introducir el valor o un bloque de código.
Not supported	Marque esta casilla para probar un valor no compatible. Esta opción es útil para probar el paso de preprocesamiento *Check for not supported value*.
Error	Introduzca el texto del error. Este campo se habilita cuando *Get value from host* está desmarcado, pero *Not supported* está marcado. Si *Get value from host* está marcado, este campo se rellena con el mensaje de error real (solo lectura) procedente del host.
Time	Se muestra la hora del valor de entrada: `now` (solo lectura).
Previous value	Introduzca un valor de entrada anterior para comparar. Solo para los pasos de preprocesamiento *Change* y *Throttling*.
Previous time	Introduzca la hora del valor de entrada anterior para comparar. Solo para los pasos de preprocesamiento *Change* y *Throttling*. El valor predeterminado se basa en el valor del campo 'Update interval' del item (si es '1m', entonces este campo se rellena con `now-1m`). Si no se especifica nada o el usuario no tiene acceso al host, el valor predeterminado es `now-30s`.
Macros	Si se utilizan macros, se enumeran junto con sus valores. Los valores son editables con fines de prueba, pero los cambios solo se guardarán dentro del contexto de prueba.
End of line sequence	Seleccione la secuencia de fin de línea para valores de entrada multilínea: **LF** - secuencia LF (line feed) **CRLF** - secuencia CRLF (carriage-return line-feed).
Preprocessing steps	Se enumeran los pasos de preprocesamiento; el resultado de la prueba se muestra para cada paso después de hacer clic en el botón *Test*. Los resultados de la prueba se truncan a un tamaño máximo de 512 KB cuando se envían al frontend. Los resultados de la prueba se pueden copiar (no más de los 512 KB truncados). Si un resultado se trunca, se muestra un icono de advertencia. La descripción de la advertencia se muestra al pasar el ratón por encima. Tenga en cuenta que los datos de más de 512 KB siguen siendo procesados completamente por Zabbix server. Si el paso falló durante la prueba, se muestra un icono de error. La descripción del error se muestra al pasar el ratón por encima. En caso de que se especifique "Custom on fail" para el paso y se ejecute esa acción, aparece una nueva línea justo después de la fila del paso de prueba de preprocesamiento, mostrando qué acción se realizó y qué resultado produjo (error o valor).
Result	El resultado final de la prueba de los pasos de preprocesamiento se muestra en todos los casos cuando todos los pasos se prueban juntos (al hacer clic en el botón *Test all steps*). También se muestra el tipo de conversión al tipo de valor del item, por ejemplo `Result converted to Numeric (unsigned)`. Los resultados de la prueba se truncan a un tamaño máximo de 512 KB cuando se envían al frontend. El resultado final se puede copiar (no más de los 512 KB truncados). Si un resultado se trunca, se muestra un icono de advertencia. La descripción de la advertencia se muestra al pasar el ratón por encima. Tenga en cuenta que los datos de más de 512 KB siguen siendo procesados completamente por Zabbix server.

Haga clic en *Test* para ver el resultado después de cada paso de preprocesamiento.

Los valores de prueba se almacenan entre sesiones de prueba, ya sea para pasos individuales o para todos los pasos, lo que permite al usuario cambiar los pasos de preprocesamiento o la configuración del item y luego volver a la ventana de prueba sin tener que volver a introducir la información. Sin embargo, los valores se pierden al actualizar la página.

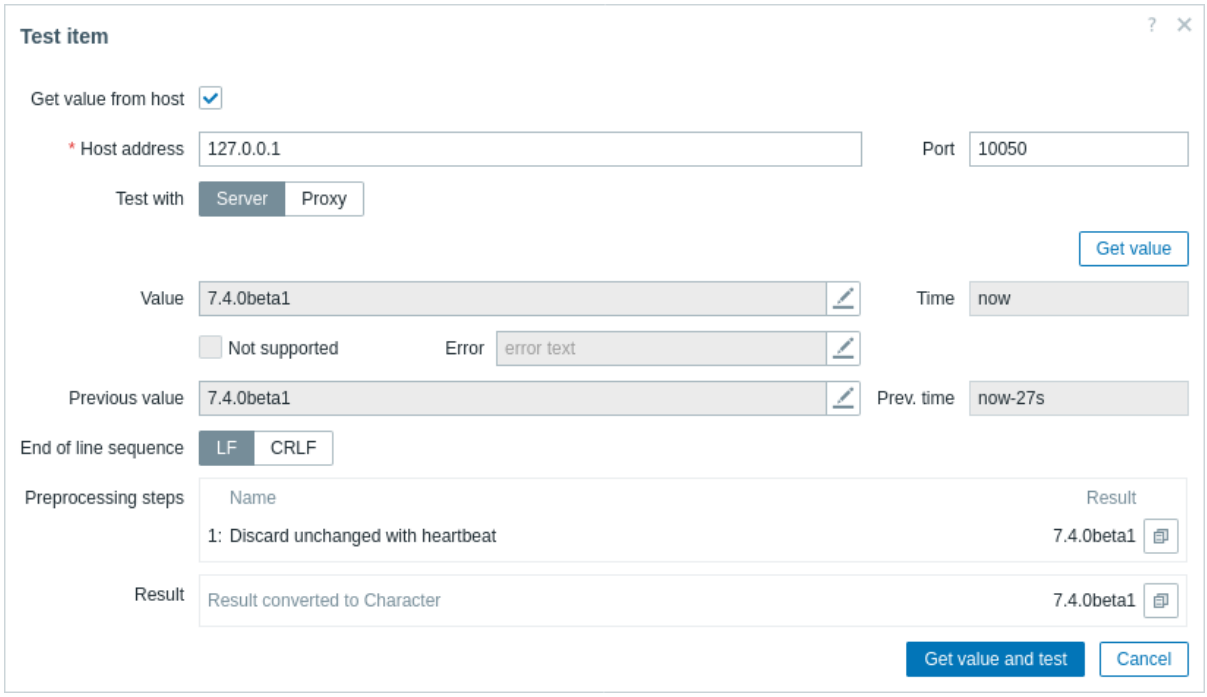
La prueba la realiza Zabbix server. El frontend envía una solicitud correspondiente al server y espera el resultado. La solicitud contiene el valor de entrada y los pasos de preprocesamiento (con las macros de usuario expandidas). Para los pasos *Change* y *Throttling*, se puede especificar opcionalmente un valor y una hora anteriores. El server responde con los resultados de cada paso de preprocesamiento.

Todos los errores técnicos o de validación de entrada se muestran en el cuadro de error situado en la parte superior de la ventana de prueba.

Probar un valor real

Para probar el preprocesamiento con un valor real:

- Marque la casilla *Obtener valor del host*
- Introduzca o verifique los parámetros del host (dirección del host, puerto, nombre del proxy o ausencia de proxy) y los detalles específicos del item (como la comunidad SNMPv2 o las credenciales de seguridad SNMPv3). Estos campos tienen en cuenta el contexto:
  - Los valores se rellenan previamente cuando es posible; por ejemplo, para los items que requieren un agent, se toma la información de la interfaz de agent seleccionada del host
  - Los valores deben rellenarse manualmente para los items de template
  - Los valores de las macros de texto sin formato se resuelven
  - Cuando el valor del campo (o parte de este) es un secreto o una macro de Vault, el campo estará vacío y deberá rellenarse manualmente. Si algún parámetro del item contiene un valor de macro secreta, se muestra el siguiente mensaje de advertencia: "El item contiene macros definidas por el usuario con valores secretos. Los valores de estas macros deben introducirse manualmente."
  - Los campos se deshabilitan cuando no son necesarios en el contexto del tipo de item (por ejemplo, los campos de dirección del host y proxy se deshabilitan para los items calculados)
- Haga clic en *Obtener valor y probar* para probar el preprocesamiento



Si ha especificado una asignación de valores en el formulario de configuración del item (el campo *Mostrar valor*), el cuadro de diálogo de prueba del item mostrará otra línea después del resultado final, denominada *Resultado con la asignación de valores aplicada*.

Parámetros específicos para obtener un valor real de un host:

Parámetro	Descripción
Obtener valor del host	Marque esta casilla para obtener un valor real del host.
Dirección del host	Introduzca la dirección del host. Este campo se rellena automáticamente con la dirección de la interfaz del host del item.
Puerto	Introduzca el puerto del host. Este campo se rellena automáticamente con el puerto de la interfaz del host del item.
Campos adicionales para interfaces SNMP (versión SNMP, comunidad SNMP, nombre de contexto, etc.)	Consulte Configuración de la monitorización SNMP para obtener más información sobre la configuración de una interfaz SNMP (v1, v2 y v3). Estos campos se rellenan automáticamente a partir de la interfaz del host del item.
Proxy	Especifique el proxy si el host está monitorizado por un proxy. Este campo se rellena automáticamente con el proxy del host (si existe).

Parámetro	Descripción
Valor	Valor recuperado del host. Al hacer clic en el campo del parámetro o en el botón de visualización/edición , se abrirá una ventana de área de texto con el valor o el bloque de código. Los valores se truncan a un tamaño máximo de 512 KB únicamente en el frontend. Si un resultado se trunca, se muestra un icono de advertencia. La descripción de la advertencia se muestra al pasar el ratón por encima. Tenga en cuenta que los datos de más de 512 KB siguen siendo procesados completamente por Zabbix server.

Para el resto de los parámetros, consulte Probar un valor hipotético más arriba.

Detalles de preprocesamiento

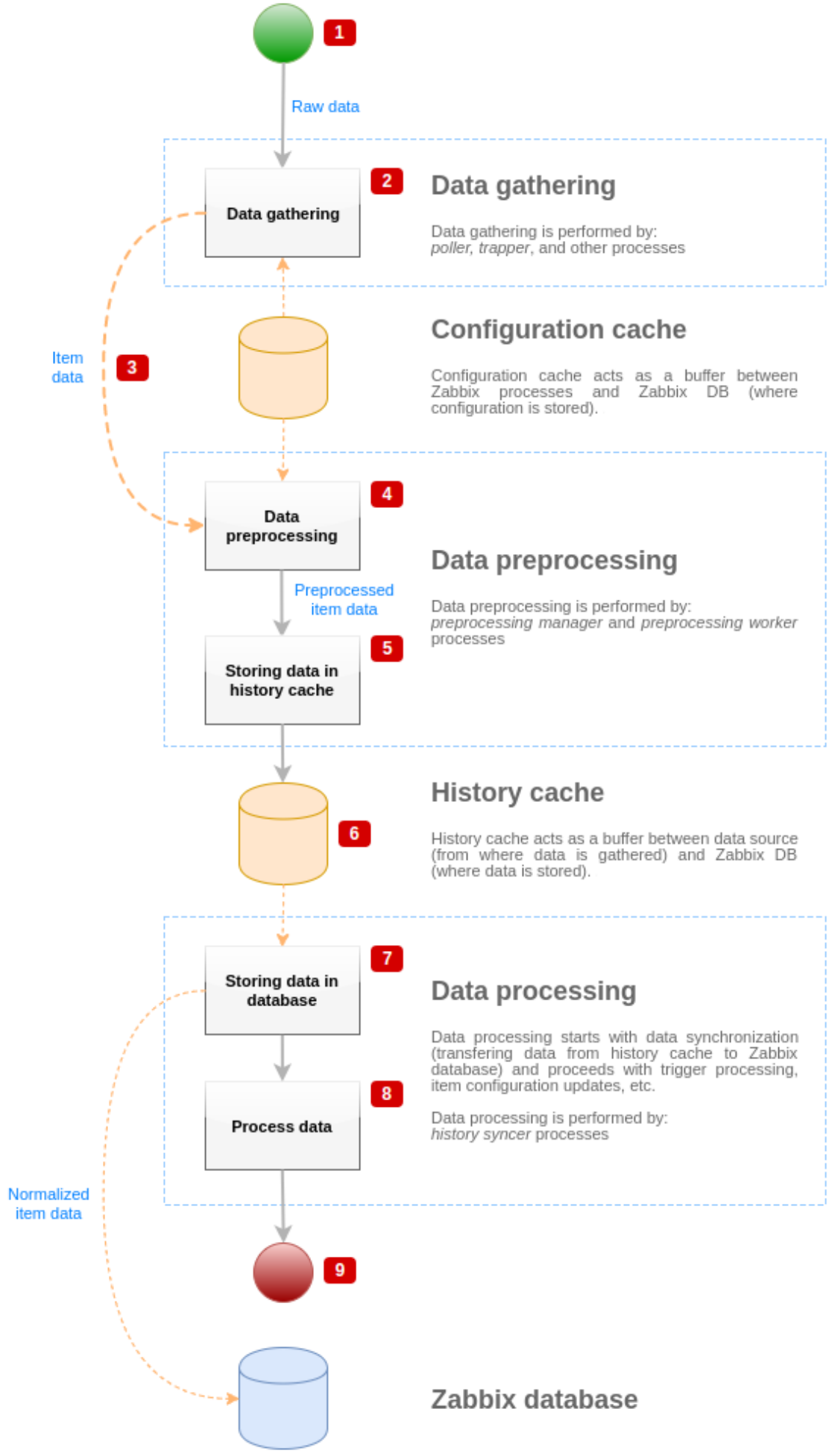
Descripción general

Esta sección proporciona detalles sobre el preprocesamiento de los valores de los item. El preprocesamiento de los item permite definir y ejecutar reglas de transformación para los valores recibidos de los item.

El preprocesamiento es gestionado por el proceso de administración del preprocesamiento, junto con los workers de preprocesamiento que realizan los pasos de preprocesamiento. Todos los valores con preprocesamiento, recibidos de diferentes recopiladores de datos, pasan por el administrador de preprocesamiento antes de añadirse a la caché de historial. Se utiliza la comunicación IPC basada en sockets entre los recopiladores de datos (pollers, trappers, etc.) y el proceso de preprocesamiento. El Zabbix server o el Zabbix proxy (para los item supervisados por el proxy) realiza los pasos de preprocesamiento.

Procesamiento del valor del item

Para visualizar el flujo de datos desde la fuente de datos hasta la base de datos de Zabbix, podemos utilizar el siguiente diagrama simplificado:


1
Raw data
Data gathering
2
Data gathering
Data gathering is performed by:
poller, trapper, and other processes
Item data
3
Configuration cache
Configuration cache acts as a buffer between Zabbix processes and Zabbix DB (where configuration is stored).
Data preprocessing
4
Data preprocessing
Data preprocessing is performed by:
preprocessing manager and preprocessing worker processes
Preprocessed item data
Storing data in history cache
5
History cache
6
History cache acts as a buffer between data source (from where data is gathered) and Zabbix DB (where data is stored).
Storing data in database
7
Data processing
Data processing starts with data synchronization (transfering data from history cache to Zabbix database) and proceeds with trigger processing, item configuration updates, etc.
Process data
8
Data processing is performed by:
history syncer processes
Normalized item data
9
Zabbix database

El diagrama anterior muestra solo los procesos, objetos y acciones relacionados con el procesamiento del valor del item en una forma **simplificada**. El diagrama no muestra cambios condicionales de dirección, manejo de errores o bucles. La caché de datos local del gestor de preprocesamiento tampoco se muestra porque no afecta directamente al flujo de datos. El objetivo de este diagrama es mostrar los procesos involucrados en el procesamiento del valor del item y la forma en que interactúan.

- La recopilación de datos comienza con datos sin procesar de una fuente de datos. En este punto, los datos contienen solo ID, marca de tiempo y valor (también pueden ser múltiples valores).
- No importa qué tipo de recopilador de datos se utilice, la idea es la misma para comprobaciones activas o pasivas, para items trapper, etc., ya que solo cambia el formato de los datos y el iniciador de la comunicación (ya sea que el recopilador de datos esté esperando una conexión y datos, o el recopilador de datos inicia la comunicación y solicita los datos). Los datos sin procesar se validan, la configuración del item se recupera de la caché de configuración (los datos se enriquecen con los datos de configuración).
- Se utiliza un mecanismo IPC basado en sockets para pasar los datos de los recopiladores de datos al gestor de preprocesamiento. En este punto, el recopilador de datos continúa recopilando datos sin esperar la respuesta del gestor de preprocesamiento.
- Se realiza el preprocesamiento de los datos. Esto incluye la ejecución de los pasos de preprocesamiento y el procesamiento de items dependientes.

Note:
Un item puede cambiar su estado a NO SOPORTADO mientras se realiza el preprocesamiento si alguno de los pasos de preprocesamiento falla.

- Los datos históricos de la caché de datos local del gestor de preprocesamiento se vacían en la caché de historial.
- En este punto, el flujo de datos se detiene hasta la siguiente sincronización de la caché de historial (cuando el proceso de sincronización de historial realiza la sincronización de datos).
- El proceso de sincronización comienza con la normalización de los datos antes de almacenar los datos en la base de datos de Zabbix. La normalización de datos realiza conversiones al tipo de item deseado (tipo definido en la configuración del item), incluida la truncación de datos textuales según los tamaños predefinidos permitidos para esos tipos (HISTORY_STR_VALUE_LEN para string, HISTORY_TEXT_VALUE_LEN para text y HISTORY_LOG_VALUE_LEN para valores de log). Los datos se envían a la base de datos de Zabbix después de que se completa la normalización.

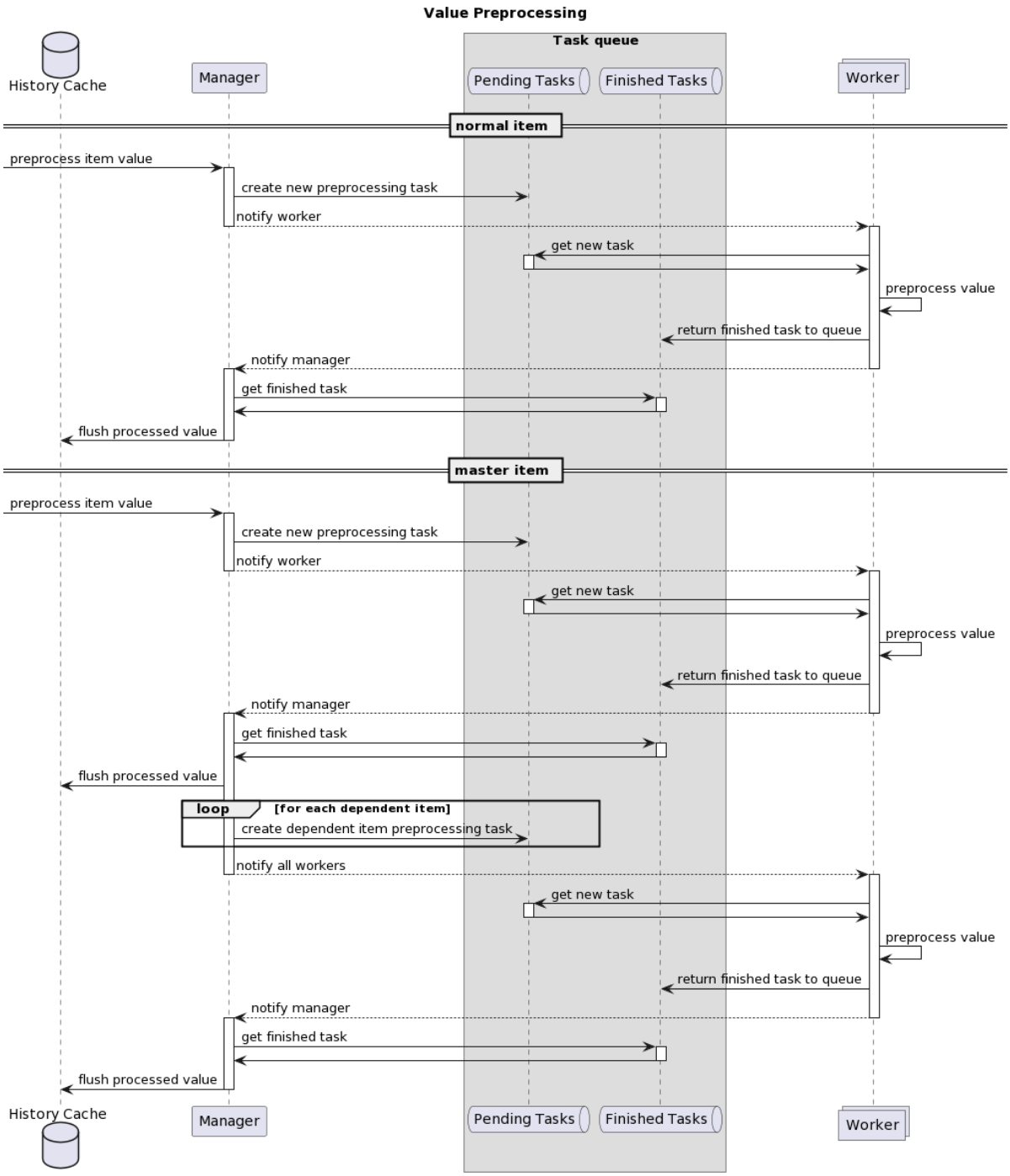
Note:
Un item puede cambiar su estado a NO SOPORTADO si la normalización de datos falla (por ejemplo, cuando un valor textual no se puede convertir a número).

- Los datos recopilados se procesan: se comprueban los triggers, se actualiza la configuración del item si el item se vuelve NO SOPORTADO, etc.
- Esto se considera el final del flujo de datos desde el punto de vista del procesamiento del valor del item.

Preprocesamiento del valor del ítem

El preprocesamiento de datos se realiza en los siguientes pasos:

- Si el ítem no tiene preprocesamiento ni ítems dependientes, su valor se agrega a la caché de historial o se envía al gestor de LLD. De lo contrario, el valor del ítem se pasa al gestor de preprocesamiento utilizando un mecanismo IPC basado en sockets UNIX.
- Se crea una tarea de preprocesamiento y se añade a la cola, y se notifica a los trabajadores de preprocesamiento sobre la nueva tarea.
- En este punto, el flujo de datos se detiene hasta que haya al menos un trabajador de preprocesamiento desocupado (es decir, que no esté ejecutando ninguna tarea).
- Cuando un trabajador de preprocesamiento está disponible, toma la siguiente tarea de la cola.
- Una vez realizado el preprocesamiento (tanto si la ejecución de los pasos de preprocesamiento falla como si tiene éxito), el valor preprocesado se añade a la cola de tareas finalizadas y se notifica al gestor sobre una nueva tarea finalizada.
- El gestor de preprocesamiento convierte el resultado al formato deseado (definido por el tipo de valor del ítem) y lo agrega a la caché de historial o lo envía al gestor de LLD.
- Si hay ítems dependientes para el ítem procesado, entonces los ítems dependientes se añaden a la cola de preprocesamiento con el valor preprocesado del ítem maestro. Los ítems dependientes se ponen en cola omitiendo las solicitudes normales de preprocesamiento de valores, pero solo para ítems maestros con el valor establecido y que no estén en estado NO SOPORTADO.



Tenga en cuenta que en el diagrama el preprocesamiento del ítem maestro está ligeramente simplificado al omitir el almacenamiento en caché del preprocesamiento.

Cola de preprocesamiento

La cola de preprocesamiento se organiza como:

- la lista de tareas pendientes:
  - tareas creadas directamente a partir de solicitudes de preprocesamiento de valores en el orden en que se recibieron
- la lista de tareas inmediatas (procesadas antes que las tareas pendientes):
  - tareas de prueba (creadas en respuesta a solicitudes de prueba de métrica/preprocesamiento por la interfaz web)
  - tareas de métricas dependientes
  - tareas de secuencia (tareas que deben ejecutarse en un orden estricto):
    - que tienen pasos de preprocesamiento usando el último valor:
      - cambio
      - limitación de frecuencia
      - JavaScript (caché de bytecode)
    - caché de preprocesamiento de métricas dependientes
- la lista de tareas finalizadas

Caché de preprocesamiento

Se introdujo la caché de preprocesamiento para mejorar el rendimiento del preprocesamiento en múltiples métricas dependientes que tienen pasos de preprocesamiento similares (lo cual es un resultado común de LLD).

La caché se realiza preprocesando una métrica dependiente y reutilizando algunos de los datos internos de preprocesamiento para el resto de las métricas dependientes. La caché de preprocesamiento solo es compatible con el primer paso de preprocesamiento de los siguientes tipos:

- Patrón Prometheus (indexa la entrada por métricas)
- JSONPath (analiza los datos en un árbol de objetos e indexa la primera expresión `[?(@.path == "value")]`)

Workers de preprocesamiento

El archivo de configuración de Zabbix server permite a los usuarios establecer la cantidad de hilos de workers de preprocesamiento. El parámetro de configuración StartPreprocessors debe usarse para definir el número de instancias preiniciadas de workers de preprocesamiento, que como mínimo debe coincidir con el número de núcleos de CPU disponibles.

Si las tareas de preprocesamiento no están limitadas por la CPU e implican solicitudes de red frecuentes, se recomienda configurar workers adicionales. El número óptimo de workers de preprocesamiento puede determinarse por muchos factores, incluido el número de items "preprocesables" (items que requieren ejecutar cualquier paso de preprocesamiento), el número de procesos de recopilación de datos, el número promedio de pasos para el preprocesamiento de items, etc. Un número insuficiente de workers puede provocar un alto uso de memoria. Para solucionar problemas de uso excesivo de memoria en su instalación de Zabbix, consulte Profiling excessive memory usage with tcmalloc.

Pero suponiendo que no haya operaciones de preprocesamiento intensivas, como el análisis de grandes fragmentos XML/JSON, el número de workers de preprocesamiento puede coincidir con el número total de recolectores de datos. De este modo, normalmente (excepto en los casos en que los datos del recolector llegan en bloque) habrá al menos un worker de preprocesamiento libre para los datos recopilados.

Warning:
Demasiados procesos de recopilación de datos (pollers, unreachable pollers, ODBC pollers, HTTP pollers, Java pollers, pingers, trappers, proxypollers) junto con IPMI manager, SNMP trapper y workers de preprocesamiento pueden agotar el límite de descriptores de archivo por proceso para el preprocessing manager.

 Agotar el límite de descriptores de archivo por proceso hará que Zabbix server se detenga, normalmente poco después del inicio, aunque a veces puede tardar más. Para evitar este tipo de problemas, revise el archivo de configuración de Zabbix server para optimizar el número de comprobaciones y procesos simultáneos. Además, si es necesario, asegúrese de que el límite de descriptores de archivo esté configurado suficientemente alto comprobando y ajustando los límites del sistema.

Canalización de procesamiento de valores

El procesamiento de valores de métricas se ejecuta en múltiples pasos (o fases) por múltiples procesos. Esto puede causar:

- Una métrica dependiente puede recibir valores, mientras que la métrica principal no puede. Esto se puede lograr utilizando el siguiente caso de uso:
  - La métrica principal tiene tipo de valor `UINT` (se puede usar una métrica trapper), la métrica dependiente tiene tipo de valor `TEXT`.
  - No se requieren pasos de preprocesamiento para ambas métricas, principal y dependiente.
  - Se debe pasar un valor textual (por ejemplo, "abc") a la métrica principal.
  - Como no hay pasos de preprocesamiento que ejecutar, el gestor de preprocesamiento verifica si la métrica principal no está en estado NO SOPORTADO y si el valor está establecido (ambos son verdaderos) y pone en cola la métrica dependiente con el mismo valor que la métrica principal (ya que no hay pasos de preprocesamiento).
  - Cuando tanto la métrica principal como la dependiente alcanzan la fase de sincronización del historial, la métrica principal pasa a estar en estado NO SOPORTADO debido al error de conversión de valor (los datos textuales no se pueden convertir a un entero sin signo).

Como resultado, la métrica dependiente recibe un valor, mientras que la métrica principal cambia su estado a NO SOPORTADO.

- Una métrica dependiente recibe un valor que no está presente en el historial de la métrica principal. El caso de uso es muy similar al anterior, excepto por el tipo de la métrica principal. Por ejemplo, si se utiliza el tipo `CHAR` para la métrica principal, entonces el valor de la métrica principal se truncará en la fase de sincronización del historial, mientras que las métricas dependientes recibirán sus valores a partir del valor inicial (no truncado) de la métrica principal.

Ejemplos de preprocesamiento

Descripción general

Esta sección presenta ejemplos de uso de pasos de preprocesamiento para realizar algunas tareas prácticas.

Filtrado de registros del registro de eventos de VMware

Este ejemplo utiliza el paso de preprocesamiento Coincide con expresión regular para filtrar eventos innecesarios del registro de eventos de VMware.

1. En un equipo VMware Hypervisor en funcionamiento, verifique que la métrica vmware.eventlog esté presente y funcionando correctamente. Tenga en cuenta que la métrica del registro de eventos podría estar ya presente en el hipervisor si se ha vinculado una plantilla VMware durante la creación del equipo.

2. En el equipo VMware Hypervisor, cree una métrica dependiente de tipo *Log* y establezca la métrica del registro de eventos como su métrica principal.

3. En la pestaña *Preprocesamiento* de la métrica dependiente, haga clic en *Añadir* para crear un paso de preprocesamiento y seleccione *Coincide con expresión regular* en el menú desplegable. Luego, especifique uno de los siguientes patrones:

- Para filtrar todos los eventos del registro:

```
.* logged in .*
```

- Para filtrar líneas que contienen nombres de usuario después de "User":

```
\bUser\s+\K\S+
```

Attention:
Si la expresión regular no coincide, la métrica dependiente se vuelve no soportada con un mensaje de error correspondiente. Para evitar esto, marque la casilla *Personalizado en caso de error* y seleccione una opción como descartar el valor o establecer uno personalizado. Tenga en cuenta que los valores descartados no se almacenan en la base de datos; como resultado, los disparadores no se evalúan y no se generan datos de tendencias.

Alternativamente, puede utilizar el paso de preprocesamiento Expresión regular para extraer grupos coincidentes y controlar la salida:

- Para extraer y mostrar todo el evento del registro que contiene "logged in", especifique los siguientes parámetros:

```
Patrón: .*logged in.*
Salida: \0
```

- Para extraer y mostrar nombres de usuario que siguen a "User":

```
Patrón: User (.*?)(?=\ )
Salida: \1
```

Comprobación del tipo de valor recuperado

Este ejemplo utiliza el paso de preprocesamiento Multiplicador personalizado para comprobar si el tipo de valor de la métrica recuperada es numérico.

En la pestaña *Preprocesamiento* de una métrica, seleccione el paso de preprocesamiento *Multiplicador personalizado* y especifique el siguiente parámetro (multiplica el valor recuperado por 1):

```
1
```

Attention:
Si el preprocesamiento falla (por ejemplo, la entrada no es numérica), la métrica se vuelve no soportada con un mensaje de error correspondiente. Para evitar esto, marque la casilla *Personalizado en caso de error* y seleccione una opción como descartar el valor o establecer uno personalizado. Tenga en cuenta que los valores descartados no se almacenan en la base de datos; como resultado, los disparadores no se evalúan y no se generan datos de tendencias.

Comprobación de valor no soportado

Este ejemplo utiliza el paso de preprocesamiento Comprobar valor no soportado para verificar si no se pudo recuperar el valor de la métrica.

Cuando un proceso poller de Zabbix server/proxy intenta recopilar el valor de una métrica, puede:

- Devolver un resultado válido.
- Devolver un resultado que inicialmente parece válido pero que puede volverse no soportado más adelante (por ejemplo, debido a una incompatibilidad de tipo de valor después del preprocesamiento).
- Devolver un error al recopilar el valor, haciendo que la métrica se vuelva no soportada. Las causas comunes incluyen:
  - Clave de métrica desconocida (para Zabbix agent, Comprobación simple o métricas internas de Zabbix)
  - OID desconocido (agente SNMP), sensor desconocido (agente IPMI) o métrica JMX no encontrada (agente JMX)

- No se puede leer el archivo de traps (trap SNMP)
- Script no encontrado (Comprobación externa)
- URL inexistente (agente HTTP, Navegador)
- Error de inicio de sesión (agente SSH, agente TELNET)
- Sintaxis de fórmula no válida (Calculada), error de sintaxis de JavaScript (Script) o SQL no válido (Monitor de base de datos)

Para detectar y manejar errores al recopilar valores de métricas, puede utilizar el paso de preprocesamiento *Comprobar valor no soportado*. Tenga en cuenta que este paso siempre se ejecuta primero y solo detecta errores que ocurren antes de que comience el preprocesamiento.

En la pestaña *Preprocesamiento* de una métrica, seleccione el paso de preprocesamiento *Comprobar valor no soportado* y especifique uno de los siguientes parámetros:

- Para cualquier error:

```
Parámetro: any error
```

- Para errores que contengan "cannot connect":

```
Parámetro: error matches
Patrón: (?i)cannot connect
```

Luego, utilice la opción *Personalizado en caso de fallo* para descartar el valor (en este caso, el error), establecer un valor personalizado o devolver un mensaje de error personalizado. Tenga en cuenta que los valores descartados no se almacenan en la base de datos; como resultado, no se evalúan disparadores ni se generan datos de tendencias.

Funcionalidad de JSONPath

Descripción general

Esta sección describe la funcionalidad JSONPath compatible dentro de los pasos de preprocesamiento de valores de métricas.

JSONPath se compone de segmentos separados por puntos. Un segmento puede tomar la forma de una palabra simple, que representa un nombre de valor JSON, el carácter comodín (*), o una construcción más compleja encerrada entre corchetes. El punto antes de un segmento entre corchetes es opcional y puede omitirse.

Ejemplo de JSONPath	Descripción
`$.object.name`	Devuelve el contenido de `object.name`.
`$.object['name']`	Devuelve el contenido de `object.name`.
`$.object.['name']`	Devuelve el contenido de `object.name`.
`$["object"]['name']`	Devuelve el contenido de `object.name`.
`$.['object'].["name"]`	Devuelve el contenido de `object.name`.
`$.object.history.length()`	Devuelve el número de elementos del array `object.history`.
`$[?(@.name == 'Object')].price.first()`	Devuelve el valor de la propiedad `price` del primer objeto llamado "Object".
`$[?(@.name == 'Object')].history.first().length()`	Devuelve el número de elementos del array history del primer objeto llamado "Object".
`$[?(@.price > 10)].length()`	Devuelve el número de objetos con un precio mayor que 10.

Véase también: Escapando caracteres especiales de valores de macros LLD en JSONPath.

Segmentos soportados

Segmento	Descripción
`<name>`	Coincide con la propiedad del objeto por nombre.
`*`	Coincide con todas las propiedades del objeto.
`['<name>']`	Coincide con la propiedad del objeto por nombre.
`['<name>', '<name>', ...]`	Coincide con la propiedad del objeto por cualquiera de los nombres listados.
`[<index>]`	Coincide con el elemento del array por índice.
`[<number>, <number>, ...]`	Coincide con el elemento del array por cualquiera de los índices listados.
`[*]`	Coincide con todas las propiedades del objeto o elementos del array.

Segmento	Descripción
`[<start>:<end>]`	Coincide con los elementos del array por el rango definido: **<start>** - el primer índice a coincidir (incluido); si no se especifica, coincide con todos los elementos del array desde el principio; si es negativo, especifica el desplazamiento inicial desde el final del array; **<end>** - el último índice a coincidir (excluido); si no se especifica, coincide con todos los elementos del array hasta el final; si es negativo, especifica el desplazamiento inicial desde el final del array.
`[?(<expression>)]`	Coincide con objetos/elementos del array aplicando una expresión de filtro.

Para encontrar un segmento coincidente ignorando su ascendencia (segmento separado), debe ir precedido de dos puntos (`..`). Por ejemplo, `$..name` o `$..['name']` devuelven los valores de todas las propiedades `name`.

Los nombres de los elementos coincidentes pueden extraerse añadiendo un sufijo de tilde (`~`) a la JSONPath. Devuelve el nombre del objeto coincidente o un índice en formato de cadena del elemento del array coincidente. El formato de salida sigue las mismas reglas que otras consultas JSONPath: los resultados de rutas definidas se devuelven 'tal cual', y los resultados de rutas indefinidas se devuelven en un array. Sin embargo, tiene un valor mínimo extraer el nombre de un elemento que coincide con una ruta definitiva, ya que ya se conoce.

Expresión de filtro

La expresión de filtro es una expresión aritmética en notación infija.

Operandos admitidos:

Operand	Descripción

`"<text>"`
`'<text>'`
	Constante de texto.
	
	Ejemplo:
	`'value: \'1\''`
	`"value: '1'"`

`<number>`	Constante numérica con compatibilidad con notación científica. Ejemplo: 123
`<jsonpath starting with $>`	Valor al que hace referencia el JSONPath desde el nodo raíz del documento de entrada; solo se admiten rutas definitivas. Ejemplo: `$.object.name`
`<jsonpath starting with @>`	Valor al que hace referencia el JSONPath desde el objeto/elemento actual; solo se admiten rutas definitivas. Ejemplo: `@.name`

Operadores admitidos: |Operator|Type|Descripción|Resultado| |-|---|---|---| |−|Binary|Resta|Number| |+|Binary|Suma|Number| |/|Binary|División|Number| |*|Binary|Multiplicación|Number| |==|Binary|Igualdad|Boolean (1/0)| |!=|Binary|Desigualdad|Boolean (1/0)| |<|Binary|Menor que|Boolean (1/0)| |<=|Binary|Menor o igual que|Boolean (1/0)| |>|Binary|Mayor que|Boolean (1/0)| |>=|Binary|Mayor o igual que|Boolean (1/0)| |=~|Binary|Coincide con una expresión regular|Boolean (1/0)| |!|Unary|NOT booleano|Boolean (1/0)| ||||Binary|OR booleano|Boolean (1/0)| |&&|Binary|AND booleano|Boolean (1/0)|

Funciones

Las funciones pueden usarse al final de JSONPath. Se pueden encadenar múltiples funciones si la función anterior devuelve un valor que es aceptado por la siguiente función.

Funciones soportadas:

Función	Descripción	Entrada	Salida
`avg`	Valor promedio de los números en una matriz de entrada	Matriz de números	Número
`min`	Valor mínimo de los números en una matriz de entrada	Matriz de números	Número
`max`	Valor máximo de los números en una matriz de entrada	Matriz de números	Número

Función	Descripción	Entrada	Salida
`sum`	Suma de los números en una matriz de entrada	Matriz de números	Número
`length`	Número de elementos en una matriz de entrada	Matriz	Número
`first`	El primer elemento de una matriz	Matriz	Una construcción JSON (objeto, matriz, valor) dependiendo del contenido de la matriz de entrada

Las funciones de agregación de JSONPath aceptan valores numéricos entre comillas. Estos valores se convierten automáticamente de cadenas a tipos numéricos cuando se necesita la agregación. Una entrada incompatible hará que la función genere un error.

Valor de salida

Los JSONPaths pueden dividirse en rutas definidas e indefinidas. Una ruta definida solo puede devolver null o una coincidencia única. Una ruta indefinida puede devolver múltiples coincidencias: JSONPaths con listas de nombres/índices separadas, cortes de arrays o segmentos de expresión. Sin embargo, cuando se utiliza una función, el JSONPath se convierte en definido, ya que las funciones siempre devuelven un único valor.

Una ruta definida devuelve el objeto/array/valor al que hace referencia. En cambio, una ruta indefinida devuelve un array de los objetos/arrays/valores coincidentes.

> **Attention:**
> El orden de las propiedades en los resultados de la consulta JSONPath puede no coincidir con el orden original de las propiedades JSON debido a métodos internos de optimización. Por ejemplo, el JSONPath `$.books[1]["author", "title"]` puede devolver `["title", "author"]`. Si es esencial preservar el orden original de las propiedades, se deben considerar métodos alternativos de procesamiento posterior a la consulta.

Reglas de formato de ruta

Los espacios en blanco (espacio, carácter de tabulación) pueden usarse en los segmentos de notación de corchetes y expresiones, por ejemplo: `$[ 'a' ][ 0 ][ ?( $.b == 'c' ) ][ : -1 ].first( )`.

Las cadenas deben ir entre comillas simples (') o dobles ("). Dentro de las cadenas, las comillas simples o dobles (dependiendo de cuáles se utilicen para encerrarlas) y las barras invertidas (\) se escapan con el carácter de barra invertida (\).

Ejemplo

```
{
  "books": [
    {
      "category": "reference",
      "author": "Nigel Rees",
      "title": "Sayings of the Century",
      "price": 8.95,
      "id": 1
    },
    {
      "category": "fiction",
      "author": "Evelyn Waugh",
      "title": "Sword of Honour",
      "price": 12.99,
      "id": 2
    },
    {
      "category": "fiction",
      "author": "Herman Melville",
      "title": "Moby Dick",
      "isbn": "0-553-21311-3",
      "price": 8.99,
      "id": 3
    },
    {
      "category": "fiction",
      "author": "J. R. R. Tolkien",
```

```
      "title": "The Lord of the Rings",
      "isbn": "0-395-19395-8",
      "price": 22.99,
      "id": 4
    }
  ],
  "services": {
    "delivery": {
      "servicegroup": 1000,
      "description": "Entrega al día siguiente en la localidad",
      "active": true,
      "price": 5
    },
    "bookbinding": {
      "servicegroup": 1001,
      "description": "Impresión y encuadernación del libro en formato A5",
      "active": true,
      "price": 154.99
    },
    "restoration": {
      "servicegroup": 1002,
      "description": "Varios métodos de restauración",
      "active": false,
      "methods": [
        {
          "description": "Limpieza química",
          "price": 46
        },
        {
          "description": "Prensado de páginas dañadas por la humedad",
          "price": 24.5
        },
        {
          "description": "Reencuadernación de libro roto",
          "price": 99.49
        }
      ]
    }
  },
  "filters": {
    "price": 10,
    "category": "fiction",
    "no filters": "no \"filters\""
  },
  "closed message": "La tienda está cerrada",
  "tags": [
    "a",
    "b",
    "c",
    "d",
    "e"
  ]
}
```


JSONPath	Type	Result
`$.filters.price`	definite	10
`$.filters.category`	definite	fiction
`$.filters['no filters']`	definite	no "filters"

JSONPath	Type	Result
`$.filters`	definite	{ ”price”: 10, ”category”: ”fiction”, ”no filters”: ”no \”filters\”” }
`$.books[1].title`	definite	Sword of Honour
`$.books[-1].author`	definite	J. R. R. Tolkien
`$.books.length()`	definite	4
`$.tags[:]`	indefinite	[”a”, ”b”, ”c”, ”d”, ”e”]
`$.tags[2:]`	indefinite	[”c”, ”d”, ”e”]
`$.tags[:3]`	indefinite	[”a”, ”b”, ”c”]
`$.tags[1:4]`	indefinite	[”b”, ”c”, ”d”]
`$.tags[-2:]`	indefinite	[”d”, ”e”]
`$.tags[:-3]`	indefinite	[”a”, ”b”]
`$.tags[:-3].length()`	definite	2
`$.books[0, 2].title`	indefinite	[”Moby Dick”, ”Sayings of the Century”]
`$.books[1]['author', "title"]`	indefinite	[”Sword of Honour”, ”Evelyn Waugh”]
`$..id`	indefinite	[1, 2, 3, 4]
`$.services..price`	indefinite	[154.99, 5, 46, 24.5, 99.49]
`$.books[?(@.id == 4 - 0.4 * 5)].title`	indefinite	[”Sword of Honour”] Nota: Esta consulta muestra que se pueden usar operaciones aritméticas en las consultas; se puede simplificar a `$.books[?(@.id == 2)].title`
`$.books[?(@.id == 2 || @.id == 4)].title`	indefinite	[”Sword of Honour”, ”The Lord of the Rings”]
`$.books[?(!(@.id == 2))].title`	indefinite	[”Sayings of the Century”, ”Moby Dick”, ”The Lord of the Rings”]
`$.books[?(@.id != 2)].title`	indefinite	[”Sayings of the Century”, ”Moby Dick”, ”The Lord of the Rings”]
`$.books[?(@.title =~ " of ")].title`	indefinite	[”Sayings of the Century”, ”Sword of Honour”, ”The Lord of the Rings”]
`$.books[?(@.price > 12.99)].title`	indefinite	[”The Lord of the Rings”]
`$.books[?(@.author > "Herman Melville")].title`	indefinite	[”Sayings of the Century”, ”The Lord of the Rings”]
`$.books[?(@.price > $.filters.price)].title`	indefinite	[”Sword of Honour”, ”The Lord of the Rings”]
`$.books[?(@.category == $.filters.category)].title`	indefinite	[”Sword of Honour”,”Moby Dick”,”The Lord of the Rings”]
`$.books[?(@.category == "fiction" && @.price < 10)].title`	indefinite	[”Moby Dick”]

JSONPath	Type	Result

`$..[?(@.id)]`	indefinite	[
		{
		”price”: 8.95,
		”id”: 1,
		”category”: ”reference”,
		”author”: ”Nigel Rees”,
		”title”: ”Sayings of the Century”
		},
		{
		”price”: 12.99,
		”id”: 2,
		”category”: ”fiction”,
		”author”: ”Evelyn Waugh”,
		”title”: ”Sword of Honour”
		},
		{
		”price”: 8.99,
		”id”: 3,
		”category”: ”fiction”,
		”author”: ”Herman Melville”,
		”title”: ”Moby Dick”,
		”isbn”: ”0-553-21311-3”
		},
		{
		”price”: 22.99,
		”id”: 4,
		”category”: ”fiction”,
		”author”: ”J. R. R. Tolkien”,
		”title”: ”The Lord of the Rings”,
		”isbn”: ”0-395-19395-8”
		}
		]

`$.services..[?(@.price > 50)].description`	indefinite	[”Impresión y encuadernación del libro en formato A5”, ”Reencuadernación de libro roto”]
`$..id.length()`	definite	4
`$.books[?(@.id == 2)].title.first()`	definite	Sword of Honour
`$..tags.first().length()`	definite	5 Nota: `$..tags` es una ruta indefinida, por lo que devuelve un array de elementos coincidentes, es decir, `[["a", "b", "c", "d", "e"]]`; `first()` devuelve el primer elemento, es decir, `["a", "b", "c", "d", "e"]`; `length()` calcula la longitud del elemento, es decir, 5.
`$.books[*].price.min()`	definite	8.95
`$..price.max()`	definite	154.99
`$.books[?(@.category == "fiction")].price.avg()`	definite	14.99
`$.books[?(@.category == $.filters.xyz)].title`	indefinite	Nota: Una consulta sin coincidencias devuelve NULL para rutas definidas e indefinidas.
`$.services[?(@.active=="true")].servicegroup`	indefinite	[1001,1000] Nota: Las constantes de texto deben usarse en comparaciones de valores booleanos.
`$.services[?(@.active=="false")].servicegroup`	indefinite	[1002] Nota: Las constantes de texto deben usarse en comparaciones de valores booleanos.
`$.services[?(@.servicegroup==1002)]~.first()`	definition	[1002]

Escape de caracteres especiales de los valores de macros LLD en JSONPath

Cuando se usan macros de descubrimiento de bajo nivel en el preprocesamiento de JSONPath y se resuelven sus valores, se aplican las siguientes reglas de escape de caracteres especiales:

- solo se consideran para el escape la barra invertida (\) y las comillas dobles (");
- si el valor de la macro resuelta contiene estos caracteres, cada uno de ellos se escapa con una barra invertida;
- si ya están escapados con una barra invertida, no se considera un escape y tanto la barra invertida como los caracteres especiales siguientes se vuelven a escapar una vez más.

Por ejemplo:

JSONPath	Valor de la macro LLD	Después de la sustitución
$.[?(@.value == "{#MACRO}")]	special "value"	$.[?(@.value == "special \"value\"")]
	c:\temp	$.[?(@.value == "c:\\temp")]
	a\\b	$.[?(@.value == "a\\\\b")]

Cuando se usa en la expresión, la macro que pueda contener caracteres especiales debe ir entre comillas dobles:

JSONPath	Valor de la macro LLD	Después de la sustitución	Resultado
$.[?(@.value == "{#MACRO}")]	special "value"	$.[?(@.value == "special \"value\"")]	OK
$.[?(@.value == {#MACRO})]		$.[?(@.value == special \"value\")]	**Expresión JSONPath incorrecta**

Cuando se usa en la ruta, la macro que pueda contener caracteres especiales debe ir entre corchetes **y** comillas dobles:

JSONPath	Valor de la macro LLD	Después de la sustitución	Resultado
$.["{#MACRO}"].value	c:\temp	$.["c:\\temp"].value	OK
$.{#MACRO}.value		$.c:\\temp.value	**Expresión JSONPath incorrecta**

Preprocesamiento de JavaScript

Descripción general

Esta sección proporciona detalles del preprocesamiento mediante JavaScript.

> **Attention:**
> No utilice asignaciones no declaradas en el JavaScript de preprocesamiento. Utilice `var` para declarar variables locales.

Preprocesamiento de JavaScript

El preprocesamiento de JavaScript se realiza invocando una función de JavaScript con un único parámetro 'value' y un cuerpo de función proporcionado por el usuario. El resultado del paso de preprocesamiento es el valor devuelto por esta función; por ejemplo, para realizar una conversión de Fahrenheit a Celsius, introduzca:

```
return (value - 32)  * 5 / 9
```

en los parámetros de preprocesamiento de JavaScript, que serán envueltos en una función de JavaScript por el server:

```
function (value)
{
   return (value - 32) * 5 / 9
}
```

El parámetro de entrada 'value' siempre se pasa como una cadena. El valor devuelto se convierte automáticamente a cadena mediante el método toString() (si falla, entonces el error se devuelve como un valor de cadena), con algunas excepciones:

- devolver un valor undefined provocará un error;
- devolver un valor null hará que el valor de entrada se descarte, de forma similar al preprocesamiento 'Discard value' en la acción 'Custom on fail'.

Los errores pueden devolverse lanzando valores/objetos (normalmente cadenas u objetos Error).

Por ejemplo:

```
if (value == 0)
    throw "Zero input value"
return 1/value
```

Cada script tiene un tiempo de espera de ejecución de 10 segundos (según el script, puede tardar más en activarse el tiempo de espera); si se supera, se devolverá un error. Se aplica un límite de heap de 512 megabytes.

El bytecode del paso de preprocesamiento de JavaScript se almacena en caché y se reutiliza cuando el paso se aplica la siguiente vez. Cualquier cambio en los pasos de preprocesamiento del item hará que el script almacenado en caché se restablezca y se vuelva a compilar más adelante.

Los fallos consecutivos en tiempo de ejecución (3 seguidos) harán que el motor se reinicialice para mitigar la posibilidad de que un script rompa el entorno de ejecución para los siguientes scripts (esta acción se registra con DebugLevel 4 y superior).

El preprocesamiento de JavaScript está implementado con el motor de JavaScript Duktape.

> **Note:**
> Las definiciones de métodos integrados de objetos de JavaScript no pueden redefinirse en los scripts de preprocesamiento de item.

Véase también: Additional JavaScript objects and global functions

Uso de macros en scripts

Es posible utilizar macros de usuario (así como macros LLD en el contexto del descubrimiento de bajo nivel) en el código JavaScript. Si un script contiene macros de usuario, estas macros son resueltas por el server/proxy antes de ejecutar pasos de preprocesamiento específicos. Tenga en cuenta que al probar los pasos de preprocesamiento en el frontend, los valores de las macros no se obtendrán y deberán introducirse manualmente.

> **Note:**
> El contexto se ignora cuando una macro se reemplaza por su valor. El valor de la macro se inserta en el código tal cual, no es posible añadir un escape adicional antes de colocar el valor en el código JavaScript. Tenga en cuenta que esto puede causar errores de JavaScript en algunos casos.

En el siguiente ejemplo, si el valor recibido supera el valor de la macro {$THRESHOLD}, se devolverá el valor del umbral (si está presente):

```
var threshold = '{$THRESHOLD}';
return (!isNaN(threshold) && value > threshold) ? threshold : value;
```

Ejemplos

Los siguientes ejemplos ilustran cómo puede utilizar el preprocesamiento de JavaScript.

Cada ejemplo contiene una breve descripción, un cuerpo de función para los parámetros de preprocesamiento de JavaScript y el resultado del paso de preprocesamiento: el valor devuelto por la función.

Ejemplo 1: Convertir un número (notación científica a entero)

Convierte el número "2.62128e+07" de notación científica a un entero.

```
return (Number(value))
```

Valor devuelto por la función: 26212800.

Ejemplo 2: Convertir un número (binario a decimal)

Convertir el número binario "11010010" a un número decimal.

```
return(parseInt(value,2))
```

Valor devuelto por la función: 210.

Ejemplo 3: Redondear un número

Redondear el número "18.2345" a 2 dígitos.

```
return(Math.round(value* 100) / 100)
```

Valor devuelto por la función: 18.23.

Ejemplo 4: Contar letras en una cadena

Cuenta el número de letras en la cadena "Zabbix".

```
return (value.length)
```

Valor devuelto por la función: 6.

Ejemplo 5: Obtener el tiempo restante

Obtener el tiempo restante (en segundos) hasta la fecha de expiración de un certificado (12 de febrero 12:33:56 2022 GMT).

```
var split = value.split(' '),
    MONTHS_LIST = ['Jan', 'Feb', 'Mar', 'Apr', 'May', 'Jun', 'Jul', 'Aug', 'Sep', 'Oct', 'Nov', 'Dec'],
    month_index = ('0' + (MONTHS_LIST.indexOf(split[0]) + 1)).slice(-2),
    ISOdate = split[3] + '-' + month_index + '-' + split[1] + 'T' + split[2],
    now = Date.now();

return parseInt((Date.parse(ISOdate) - now) / 1000);
```

Valor devuelto por la función: 44380233.

Ejemplo 6: Eliminar propiedades JSON

Modifica la estructura de datos JSON eliminando cualquier propiedad con la clave "data_size" o "index_size".

```
var obj=JSON.parse(value);

for (i = 0; i < Object.keys(obj).length; i++) {
    delete obj[i]["data_size"];
    delete obj[i]["index_size"];
}

return JSON.stringify(obj)
```

Valor aceptado por la función:

```
[
    {
        "table_name":"history",
        "data_size":"326.05",
        "index_size":"174.34"
    },
    {
        "table_name":"history_log",
        "data_size":"6.02",
        "index_size":"3.45"
    }
]
```

Valor devuelto por la función:

```
[
    {
        "table_name":"history"
    },
    {
        "table_name":"history_log"
    }
]
```

Ejemplo 7: Convertir el estado de Apache a JSON

Convierta el valor recibido de un item de Zabbix agent web.page.get (por ejemplo, web.page.get[http://127.0.0.1:80/server-status?auto]) en un objeto JSON.

```
// Convertir el estado de Apache a JSON

// Dividir el valor en subcadenas y colocar estas subcadenas en un array
```

```
var lines = value.split('\n');

// Crear un objeto vacío "output"
var output = {};

// Crear un objeto "workers" con propiedades predefinidas
var workers = {
    '_': 0, 'S': 0, 'R': 0, 'W': 0,
    'K': 0, 'D': 0, 'C': 0, 'L': 0,
    'G': 0, 'I': 0, '.': 0
};

// Añadir las subcadenas del array "lines" al objeto "output" como propiedades (pares clave-valor)
for (var i = 0; i < lines.length; i++) {
    var line = lines[i].match(/([A-z0-9 ]+): (.*)/);

    if (line !== null) {
        output[line[1]] = isNaN(line[2]) ? line[2] : Number(line[2]);
    }
}

// Métricas multiversión
output.ServerUptimeSeconds = output.ServerUptimeSeconds || output.Uptime;
output.ServerVersion = output.ServerVersion || output.Server;

// Analizar la propiedad "Scoreboard" para obtener el recuento de workers
if (typeof output.Scoreboard === 'string') {
    for (var i = 0; i < output.Scoreboard.length; i++) {
        var char = output.Scoreboard[i];

        workers[char]++;
    }
}

// Añadir los datos de workers al objeto "output"
output.Workers = {
    waiting: workers['_'], starting: workers['S'], reading: workers['R'],
    sending: workers['W'], keepalive: workers['K'], dnslookup: workers['D'],
    closing: workers['C'], logging: workers['L'], finishing: workers['G'],
    cleanup: workers['I'], slot: workers['.']
};

// Devolver la cadena JSON
return JSON.stringify(output);
```

Valor aceptado por la función:

```
HTTP/1.1 200 OK
Date: Mon, 27 Mar 2023 11:08:39 GMT
Server: Apache/2.4.52 (Ubuntu)
Vary: Accept-Encoding
Content-Encoding: gzip
Content-Length: 405
Content-Type: text/plain; charset=ISO-8859-1

127.0.0.1
ServerVersion: Apache/2.4.52 (Ubuntu)
ServerMPM: prefork
Server Built: 2023-03-08T17:32:01
CurrentTime: Monday, 27-Mar-2023 14:08:39 EEST
RestartTime: Monday, 27-Mar-2023 12:19:59 EEST
ParentServerConfigGeneration: 1
ParentServerMPMGeneration: 0
```

```
ServerUptimeSeconds: 6520
ServerUptime: 1 hour 48 minutes 40 seconds
Load1: 0.56
Load5: 0.33
Load15: 0.28
Total Accesses: 2476
Total kBytes: 8370
Total Duration: 52718
CPUUser: 8.16
CPUSystem: 3.44
CPUChildrenUser: 0
CPUChildrenSystem: 0
CPULoad: .177914
Uptime: 6520
ReqPerSec: .379755
BytesPerSec: 3461.58
BytesPerReq: 3461.58
DurationPerReq: 21.2916
BusyWorkers: 2
IdleWorkers: 6
Scoreboard: ____KW__.....................................................................................
```

Valor devuelto por la función:


```
{
    "Date": "Mon, 27 Mar 2023 11:08:39 GMT",
    "Server": "Apache/2.4.52 (Ubuntu)",
    "Vary": "Accept-Encoding",
    "Encoding": "gzip",
    "Length": 405,
    "Type": "text/plain; charset=ISO-8859-1",
    "ServerVersion": "Apache/2.4.52 (Ubuntu)",
    "ServerMPM": "prefork",
    "Server Built": "2023-03-08T17:32:01",
    "CurrentTime": "Monday, 27-Mar-2023 14:08:39 EEST",
    "RestartTime": "Monday, 27-Mar-2023 12:19:59 EEST",
    "ParentServerConfigGeneration": 1,
    "ParentServerMPMGeneration": 0,
    "ServerUptimeSeconds": 6520,
    "ServerUptime": "1 hour 48 minutes 40 seconds",
    "Load1": 0.56,
    "Load5": 0.33,
    "Load15": 0.28,
    "Total Accesses": 2476,
    "Total kBytes": 8370,
    "Total Duration": 52718,
    "CPUUser": 8.16,
    "CPUSystem": 3.44,
    "CPUChildrenUser": 0,
    "CPUChildrenSystem": 0,
    "CPULoad": 0.177914,
    "Uptime": 6520,
    "ReqPerSec": 0.379755,
    "BytesPerSec": 1314.55,
    "BytesPerReq": 3461.58,
    "DurationPerReq": 21.2916,
    "BusyWorkers": 2,
    "IdleWorkers": 6,
    "Scoreboard": "____KW__.....................................................................................
    "Workers": {
        "waiting": 6,
        "starting": 0,
        "reading": 0,
```

```
        "sending": 1,
        "keepalive": 1,
        "dnslookup": 0,
        "closing": 0,
        "logging": 0,
        "finishing": 0,
        "cleanup": 0,
        "slot": 142
    }
}
```

Objetos JavaScript adicionales

Descripción general

Esta sección describe las adiciones de Zabbix al lenguaje JavaScript implementadas con Duktape, y las funciones globales de JavaScript soportadas.

Attention:
No utilice asignaciones no declaradas en el JavaScript de preprocesamiento. Utilice `var` para declarar variables locales.

Objetos integrados

Zabbix

El objeto Zabbix proporciona interacción con la funcionalidad interna de Zabbix.

Método	Descripción
`log(loglevel, message)`	Escribe <message> en el registro de Zabbix usando el nivel de registro <loglevel> (ver el parámetro DebugLevel en el archivo de configuración).

Ejemplo:

```
Zabbix.log(3, "esta es una entrada de registro escrita con el nivel de registro 'Warning'")
```

Puede utilizar los siguientes alias:

Alias	Alias de
console.log(object)	Zabbix.log(4, JSON.stringify(object))
console.warn(object)	Zabbix.log(3, JSON.stringify(object))
console.error(object)	Zabbix.log(2, JSON.stringify(object))

Attention:
El tamaño total de todos los mensajes registrados está limitado a 8 MB por ejecución de script.

Método	Descripción
`sleep(delay)`	Retrasa la ejecución de JavaScript por `delay` milisegundos.

Ejemplo (retrasar la ejecución 15 segundos):

```
Zabbix.sleep(15000)
```

HttpRequest

Este objeto encapsula un controlador cURL y permite realizar solicitudes HTTP sencillas. Los errores se generan como excepciones.

Attention:
La inicialización de varios objetos `HttpRequest` está limitada a 10 por ejecución del script.

Método	Descripción
`addHeader(value)`	Añade un campo de encabezado HTTP. Este campo se utiliza para todas las solicitudes posteriores hasta que se borra con el método `clearHeader()`. La longitud total de los campos de encabezado que se pueden añadir a un único objeto `HttpRequest` está limitada a 128 Kbytes (incluidos los caracteres especiales y los nombres de los encabezados).
`clearHeader()`	Borra el encabezado HTTP. Si no se han establecido campos de encabezado, `HttpRequest` establecerá Content-Type en application/json si los datos enviados tienen formato JSON; de lo contrario, en text/plain.
`connect(url)`	Envía una solicitud HTTP CONNECT a la URL y devuelve la respuesta.
`customRequest(method, url, data)`	Permite especificar cualquier método HTTP en el primer parámetro. Envía la solicitud del método a la URL con una carga útil *data* opcional y devuelve la respuesta.
`delete(url, data)`	Envía una solicitud HTTP DELETE a la URL con una carga útil *data* opcional y devuelve la respuesta.
`getHeaders(<asArray>)`	Devuelve el objeto con los campos de encabezado HTTP recibidos. El parámetro `asArray` puede establecerse en "true" (por ejemplo, `getHeaders(true)`), "false" o dejarse sin definir. Si se establece en "true", los valores de los campos de encabezado HTTP recibidos se devolverán como matrices; esto debe utilizarse para recuperar los valores de campos de varios encabezados con el mismo nombre. Si no se establece o se establece en "false", los valores de los campos de encabezado HTTP recibidos se devolverán como cadenas.
`get(url, data)`	Envía una solicitud HTTP GET a la URL con una carga útil *data* opcional y devuelve la respuesta.
`head(url)`	Envía una solicitud HTTP HEAD a la URL y devuelve la respuesta.
`options(url)`	Envía una solicitud HTTP OPTIONS a la URL y devuelve la respuesta.
`patch(url, data)`	Envía una solicitud HTTP PATCH a la URL con una carga útil *data* opcional y devuelve la respuesta.
`put(url, data)`	Envía una solicitud HTTP PUT a la URL con una carga útil *data* opcional y devuelve la respuesta.
`post(url, data)`	Envía una solicitud HTTP POST a la URL con una carga útil *data* opcional y devuelve la respuesta.
`getStatus()`	Devuelve el código de estado de la última solicitud HTTP.
`setProxy(proxy)`	Establece el proxy HTTP en el valor "proxy". Si este parámetro está vacío, no se utiliza ningún proxy.

`setHttpAuth(bitmask, username, password)`	Establece los métodos de autenticación HTTP habilitados (HTTPAUTH_BASIC, HTTPAUTH_DIGEST, HTTPAUTH_NEGOTIATE, HTTPAUTH_NTLM, HTTPAUTH_NONE) en el parámetro 'bitmask'.
	La marca HTTPAUTH_NONE permite deshabilitar la autenticación HTTP.
	Ejemplos:
	`request.setHttpAuth(HTTPAUTH_NTLM | HTTPAUTH_BASIC, username, password)`
	`request.setHttpAuth(HTTPAUTH_NONE)`

`trace(url, data)`	Envía una solicitud HTTP TRACE a la URL con una carga útil *data* opcional y devuelve la respuesta.

Ejemplo:

```
try {
    Zabbix.log(4, 'jira webhook script value='+value);

    var result = {
        'tags': {
            'endpoint': 'jira'
        }
    },
    params = JSON.parse(value),
    req = new HttpRequest(),
    fields = {},
    resp;

    req.addHeader('Content-Type: application/json');
    req.addHeader('Authorization: Basic '+params.authentication);

    fields.summary = params.summary;
    fields.description = params.description;
    fields.project = {"key": params.project_key};
    fields.issuetype = {"id": params.issue_id};
    resp = req.post('https://jira.example.com/rest/api/2/issue/',
        JSON.stringify({"fields": fields})
```

```
    );

    if (req.getStatus() != 201) {
        throw 'Response code: '+req.getStatus();
    }

    resp = JSON.parse(resp);
    result.tags.issue_id = resp.id;
    result.tags.issue_key = resp.key;
} catch (error) {
    Zabbix.log(4, 'jira issue creation failed json : '+JSON.stringify({"fields": fields}));
    Zabbix.log(4, 'jira issue creation failed : '+error);

    result = {};
}

return JSON.stringify(result);
```

XML

El objeto XML permite el procesamiento de datos XML en el preprocesamiento de elementos y descubrimiento de bajo nivel y webhooks.

Attention:
Para utilizar el objeto XML, el servidor/proxy debe estar compilado con soporte para libxml2.

Método	Descripción
`XML.query(data, expression)`	Recupera el contenido del nodo usando XPath. Devuelve null si no se encuentra el nodo. **expression**: una expresión XPath; **data**: datos XML como cadena.
`XML.toJson(data)`	Convierte datos en formato XML a JSON.
`XML.fromJson(object)`	Convierte datos en formato JSON a XML.

Ejemplo:

Entrada:

```
<menu>
    <food type = "breakfast">
        <name>Chocolate</name>
        <price>$5.95</price>
        <description></description>
        <calories>650</calories>
    </food>
</menu>
```

Salida:

```
{
    "menu": {
        "food": {
            "@type": "breakfast",
            "name": "Chocolate",
            "price": "$5.95",
            "description": null,
            "calories": "650"
        }
    }
}
```

Reglas de serialización

La conversión de XML a JSON se procesará de acuerdo con las siguientes reglas (para las conversiones de JSON a XML se aplican las reglas inversas):

1. Los atributos XML se convertirán en claves cuyos nombres estarán precedidos por '@'.

Ejemplo:

Entrada:

```
<xml foo="FOO">
  <bar>
    <baz>BAZ</baz>
  </bar>
</xml>
```

Salida:

```
{
  "xml": {
    "@foo": "FOO",
    "bar": {
      "baz": "BAZ"
    }
  }
}
```

2. Los elementos autocontenidos (<foo/>) se convertirán como si tuvieran el valor 'null'.

Ejemplo:

Entrada:

```
<xml>
  <foo/>
</xml>
```

Salida:

```
{
  "xml": {
    "foo": null
  }
}
```

3. Los atributos vacíos (con valor "") se convertirán como si tuvieran el valor de cadena vacía ('').

Ejemplo:

Entrada:

```
<xml>
  <foo bar="" />
</xml>
```

Salida:

```
{
  "xml": {
    "foo": {
      "@bar": ""
    }
  }
}
```

4. Los nodos hijos múltiples con el mismo nombre de elemento se convertirán en una sola clave que tiene como valor un array de valores.

Ejemplo:

Entrada:

```
<xml>
  <foo>BAR</foo>
  <foo>BAZ</foo>
  <foo>QUX</foo>
</xml>
```

Salida:

```
{
  "xml": {
    "foo": ["BAR", "BAZ", "QUX"]
  }
}
```

5. Si un elemento de texto no tiene atributos ni hijos, se convertirá como una cadena.

Ejemplo:

Entrada:

```
<xml>
    <foo>BAZ</foo>
</xml>
```

Salida:

```
{
  "xml": {
    "foo": "BAZ"
   }
}
```

6. Si un elemento de texto no tiene hijos pero sí atributos, el contenido de texto se convertirá en un elemento con la clave '#text' y el contenido como valor; los atributos se convertirán como se describe en la regla de serialización 1.

Ejemplo:

Entrada:

```
<xml>
  <foo bar="BAR">
    BAZ
  </foo>
</xml>
```

Salida:

```
{
  "xml": {
    "foo": {
      "@bar": "BAR",
      "#text": "BAZ"
    }
  }
}
```

Funciones globales de JavaScript

Se han implementado funciones globales adicionales de JavaScript con Duktape:

- `btoa(data)` - codifica los datos en una cadena base64.
- `atob(base64_string)` - decodifica la cadena base64 como un buffer Uint8Array.

```
try {
    b64 = btoa("test string");
    buffer = atob(b64);

    // Tenga en cuenta que la lógica de decodificación depende del formato de datos del buffer.
    decoded = String.fromCharCode.apply(this, [].slice.call(buffer));
}
catch (error) {
    return {'error.name' : error.name, 'error.message' : error.message};
}
```

- `md5(data)` - calcula el hash MD5 de los datos.
- `sha256(data)` - calcula el hash SHA256 de los datos.

- `hmac('<hash type>',key,data)` - devuelve un hash HMAC como una cadena en formato hexadecimal; se admiten `md5` y `sha256` como `hash type`; los parámetros `key` y `data` admiten datos binarios.

  Ejemplos:

  - `hmac('md5',key,data)`
  - `hmac('sha256',key,data)`

- `sign(hash,key,data)` - devuelve la firma calculada (firma RSA con SHA-256) como una cadena, donde:
 hash - solo se permite `sha256`, de lo contrario se lanza un error.
 key - la clave privada. Debe corresponder al estándar PKCS#1 o PKCS#8. La clave puede proporcionarse en diferentes formas:

  - con espacios en lugar de saltos de línea
  - con `\n` escapados o no escapados en lugar de saltos de línea
  - sin ningún salto de línea como una cadena de una sola línea
  - como una cadena en formato JSON

  La clave también puede cargarse desde una macro de usuario/macro secreta/vault.

  data - los datos que se firmarán. Puede ser una cadena (también se admiten datos binarios) o un buffer (Uint8Array/ArrayBuffer).

  Ejemplo:

  - `sign('sha256',key,data)`

  Se utiliza OpenSSL o GnuTLS para calcular las firmas. Si Zabbix se compiló sin ninguna de estas bibliotecas de cifrado, se lanzará un error ('missing OpenSSL or GnuTLS library').

Objetos JavaScript de item de navegador

Descripción general

Esta sección describe las adiciones de Zabbix al lenguaje JavaScript implementadas con Duktape para su uso en el script de la métrica de navegador. Estas adiciones complementan los objetos JavaScript descritos en la página de Objetos JavaScript adicionales.

Browser

El objeto `Browser` gestiona las sesiones de WebDriver, inicializando una sesión al crearse y terminándola al destruirse. Un solo script puede soportar hasta cuatro objetos `Browser`.

Para construir un objeto `Browser`, utilice la sintaxis `new Browser(options)`. El parámetro `options` (*objeto JSON*) especifica las opciones del navegador, normalmente el resultado del método de opciones de WebDriver (por ejemplo, `Browser.chromeOptions()`).

Se admiten los siguientes métodos con el objeto `Browser`.

Método	Descripción
`navigate(url)`	Navega a la URL especificada. Parámetros: `url` - (string) URL a la que navegar.
`getUrl()`	Devuelve una cadena con la URL de la página abierta.
`getPageSource()`	Devuelve una cadena con el código fuente de la página abierta.
`findElement(strategy, selector)`	Devuelve un objeto `Element` con un elemento en la página abierta (o devuelve `null` si ningún elemento coincide con `strategy` y `selector`). Parámetros: `strategy` - (string, css selector/link text/partial link text/tag name/xpath) Estrategia de localización; `selector` - (string) Selector del elemento utilizando la estrategia de localización especificada.

Método	Descripción
`findElements(strategy, target)`	Devuelve un array de objetos `Element` con múltiples elementos en la página abierta (o devuelve un array vacío si ningún elemento coincide con la estrategia de localización y el objetivo). Parámetros: `strategy` - (string, css selector/link text/partial link text/tag name/xpath) Estrategia de localización; `target` - (string) Selector del elemento utilizando la estrategia de localización especificada.
`getCookies()`	Devuelve un array de objetos `Cookie`.
`addCookie(cookie)`	Establece cookies. Parámetros: `cookie` - (objeto `Cookie`) Cookie a establecer.
`getScreenshot()`	Devuelve una cadena codificada en base64 que representa una imagen del área visible del navegador.
`setScreenSize(x,y)`	Establece el tamaño del área visible del navegador. Parámetros: `x` - (string) Ancho en píxeles; `y` - (string) Alto en píxeles.
`setScriptTimeout(timeout)`	Establece el tiempo de espera para la carga de scripts. Parámetros: `timeout` - (integer) Valor de tiempo de espera en milisegundos.
`setSessionTimeout(timeout)`	Establece el tiempo de espera de la sesión (carga de página). Parámetros: `timeout` - (integer) Valor de tiempo de espera en milisegundos.
`setElementWaitTimeout(timeout)`	Establece el tiempo de espera de la estrategia de localización de elementos (implícito). Parámetros: `timeout` - (integer) Valor de tiempo de espera en milisegundos.
`collectPerfEntries(mark)`	Recopila entradas de rendimiento para recuperarlas con el método `getResult()`. Parámetros: `mark` - (string, opcional) marca de la instantánea de rendimiento.
`getRawPerfEntries()`	Devuelve un array de objetos de entrada de rendimiento.
`getResult()`	Devuelve un objeto `Result` con estadísticas de la sesión del navegador (información de error, instantáneas de rendimiento, etc.).
`getError()`	Devuelve un objeto `BrowserError` con errores del navegador (o devuelve `null` si no hay errores del navegador).
`setError(message)`	Establece un mensaje de error personalizado que se incluirá en el objeto `Result`. Parámetros: `message` - (string) Mensaje de error.
`discardError()`	Descarta el error que se devolverá en el objeto `Result`.
`getAlert()`	Devuelve un objeto `Alert` con alertas del navegador (o devuelve `null` si no hay alertas del navegador).
`chromeOptions()`	Devuelve un objeto `chromeOptions` con opciones predefinidas del navegador Chrome.
`firefoxOptions()`	Devuelve un objeto `firefoxOptions` con opciones predefinidas del navegador Firefox.
`safariOptions()`	Devuelve un objeto `safariOptions` con opciones predefinidas del navegador Safari.
`edgeOptions()`	Devuelve un objeto `edgeOptions` con opciones predefinidas del navegador Edge.

Método	Descripción
`switchFrame(target)`	Cambia al frame especificado. Parámetros: `target` - (elemento del navegador o integer, opcional) Frame de destino. Para seleccionar un frame por elemento, pase el elemento. Para seleccionar un frame por índice, pase el número. Si se deja vacío, cambiará al contexto de navegación de nivel superior.

Todos los métodos de Browser pueden lanzar los siguientes errores:

- `BrowserError` - derivado del objeto `Error` que se lanza si el constructor de `Browser` falla; contiene una propiedad adicional `browser` con un objeto `Browser` que lanzó este `BrowserError`.
- `WebdriverError` - derivado de `BrowserError`; contiene las mismas propiedades que el objeto `BrowserError`, que indican si el error se generó en respuesta a un error en la respuesta de WebDriver.

Element

El objeto `Element` es devuelto por los métodos `findElement()`/`findElements()` del objeto `Browser` y no puede construirse directamente.

El objeto `Element` representa un elemento en la página web y proporciona métodos para interactuar con él.

Los siguientes métodos son compatibles con el objeto `Element`.

Method	Description
`getAttribute(name)`	Devuelve una cadena con el valor de un atributo del elemento (o devuelve `null` si no se encontró el atributo especificado). Parameters: `name` - (string) Nombre del atributo.
`getProperty(name)`	Devuelve una cadena con el valor de una propiedad del elemento (o devuelve `null` si no se encontró la propiedad especificada). Parameters: `name` - (string) Nombre de la propiedad.
`getText()`	Devuelve una cadena con el valor del texto del elemento.
`click()`	Hace clic en un elemento.
`clear()`	Borra el contenido de un elemento editable.
`sendKeys(keys)`	Envía teclas. Parameters: `keys` - (string) Teclas a enviar.

Cookie

El objeto `Cookie` es devuelto por el método `getCookies()` del objeto `Browser` y pasado al método `addCookie()`.

Aunque el objeto `Cookie` no tiene ningún método, puede contener las siguientes propiedades:

Propiedad	Tipo	Descripción
name	string	Nombre de la cookie.
value	string	Valor de la cookie.
path	string	Ruta para la que la cookie es válida. Por defecto es `"/"` si se omite al añadir una cookie.
domain	string	Dominio al que la cookie es visible. Por defecto es el dominio de la URL del documento activo del contexto de navegación actual de la sesión si se omite al añadir una cookie.
secure	boolean	Indica si la cookie es segura. Por defecto es `false` si se omite al añadir una cookie.
httpOnly	boolean	Indica si la cookie es sólo HTTP. Por defecto es `false` si se omite al añadir una cookie.

Propiedad	Tipo	Descripción
expiry	integer	Tiempo de expiración de la cookie (en segundos desde la Época Unix). No debe establecerse si se omite al añadir una cookie.
sameSite	string	El atributo `sameSite` de la cookie, que controla si la cookie debe restringirse a un contexto de primera parte o de mismo sitio. Puede establecerse en `"Lax"` o `"Strict"`. Por defecto es `"None"` si se omite al añadir una cookie.

Alerta

El objeto `Alert` representa una alerta de página web, es devuelto por el método `getAlert()` del objeto `Browser` y no puede ser construido directamente.

El objeto `Alert` contiene la propiedad `text` con el texto de la alerta (o `null` si no hay alertas).

Los siguientes métodos son compatibles con el objeto `Alert`.

Método	Descripción
`accept()`	Aceptar la alerta.
`dismiss()`	Descartar la alerta.

Resultado

El objeto `Result` contiene estadísticas de la sesión y es devuelto por el método `getResult()` del objeto `Browser`.

Normalmente, el objeto `Result` se convierte en cadena y se devuelve desde el script, y luego se analiza en valores de métricas dependientes mediante preprocesamiento.

Aunque el objeto `Result` no tiene ningún método, puede contener las siguientes propiedades.

Propiedad		Tipo	Descripción
duration		string	Duración de la sesión desde la creación de la sesión hasta la obtención del resultado.
error		object	Información de error.
	http_status	integer	Estado HTTP devuelto por el WebDriver (o 0 si no hay errores de WebDriver).
	error_code	string	Error devuelto por el WebDriver (o cadena vacía si no hay errores de WebDriver).
	message	string	Mensaje de error de WebDriver (o cadena vacía si no hay errores de WebDriver).
performance_data		object	Estadísticas de rendimiento.
	summary	object	Resumen de rendimiento.
		navigation object	Resumen de navegación.
		resource object	Resumen de recursos.
	details	array of objects	Estadísticas de rendimiento después de cada operación que podría haber resultado en una navegación.
		mark string	*(opcional)* Marca de instantánea de rendimiento especificada con el método `collectPerfEntries()`.
		navigation object	Estadísticas de navegación.
		resource object	Resumen de recursos para este paso.
		user array of objects	Matriz de estadísticas de tipo marca/medida.
	marks	array of objects	Índices de instantáneas de rendimiento marcadas.
		name string	Nombre de la marca de instantánea de rendimiento.
		index integer	Índice de la instantánea de rendimiento en la matriz details.

Preprocesamiento de CSV a JSON

Descripción general

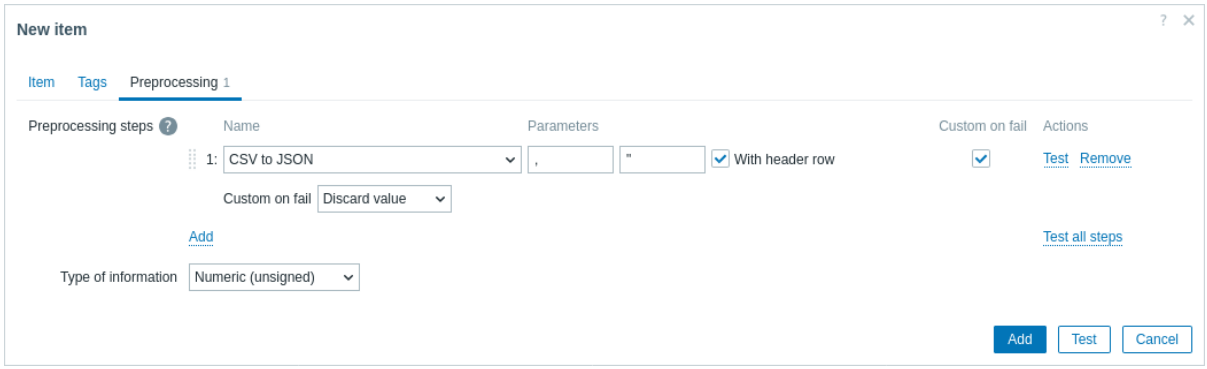
En este paso de preprocesamiento es posible convertir los datos de un archivo CSV al formato JSON. Es compatible con:

- items (prototipos de item)
- reglas de descubrimiento de bajo nivel

Configuración

Para configurar un paso de preprocesamiento de CSV a JSON:

1. Vaya a la pestaña *Preprocesamiento* en la configuración de item/regla de descubrimiento.
2. Haga clic en *Añadir*.
3. Seleccione la opción *CSV a JSON*.



El primer parámetro permite establecer un delimitador personalizado. Tenga en cuenta que, si la primera línea de la entrada CSV comienza con "Sep=" y va seguida de un único carácter UTF-8, dicho carácter se utilizará como delimitador en caso de que no se haya establecido el primer parámetro. Si no se ha establecido el primer parámetro y no se recupera ningún delimitador de la línea "Sep=", se utilizará una coma como separador.

El segundo parámetro opcional permite establecer un símbolo de comillas.

Si se marca la casilla *Con fila de encabezado*, los valores de la línea de encabezado se interpretarán como nombres de columnas (consulte Procesamiento del encabezado para obtener más información).

Si se marca la casilla *Personalizado en caso de error*, el item no pasará a ser no compatible en caso de que falle un paso de preprocesamiento. Además, se pueden establecer opciones personalizadas para gestionar errores: descartar el valor, establecer un valor especificado o establecer un mensaje de error especificado.

Procesamiento de la cabecera

La línea de cabecera del archivo CSV puede procesarse de dos maneras diferentes:

- Si la casilla *Con fila de cabecera* está marcada, los valores de la línea de cabecera se interpretan como nombres de columna. En este caso, los nombres de columna deben ser únicos y la fila de datos no debe contener más columnas que la fila de cabecera.
- Si la casilla *Con fila de cabecera* no está marcada, la línea de cabecera se interpreta como datos. Los nombres de columna se generan automáticamente (1,2,3,4...).

Ejemplo de archivo CSV:

```
Nr,Item name,Key,Qty
1,active agent item,agent.hostname,33
"2","passive agent item","agent.version","44"
3,"active,passive agent items",agent.ping,55
```

Note:
Un carácter de comillas dentro de un campo entre comillas en la entrada debe escaparse anteponiéndole otro carácter de comillas.

Procesando la línea de cabecera

Salida JSON cuando se espera una línea de cabecera:

```
[
   {
      "Nr":"1",
      "Item name":"active agent item",
      "Key":"agent.hostname",
      "Qty":"33"
   },
   {
      "Nr":"2",
      "Item name":"passive agent item",
      "Key":"agent.version",
      "Qty":"44"
   },
```

```
   {
      "Nr":"3",
      "Item name":"active,passive agent items",
      "Key":"agent.ping",
      "Qty":"55"
   }
]
```

Sin procesamiento de línea de cabecera

Salida JSON cuando no se espera una línea de cabecera:

```
[
   {
      "1":"Nr",
      "2":"Item name",
      "3":"Key",
      "4":"Qty"
   },
   {
      "1":"1",
      "2":"active agent item",
      "3":"agent.hostname",
      "4":"33"
   },
   {
      "1":"2",
      "2":"passive agent item",
      "3":"agent.version",
      "4":"44"
   },
   {
      "1":"3",
      "2":"active,passive agent items",
      "3":"agent.ping",
      "4":"55"
   }
]
```

Tipos de item

Descripción general

Los tipos de item abarcan diversos métodos para obtener datos de su sistema. Cada tipo de item tiene su propio conjunto de claves de item compatibles y parámetros obligatorios.

Actualmente, Zabbix ofrece los siguientes tipos de item:

- Zabbix agent
  - Supervisión de archivos de registro
- Comprobación simple
  - Supervisión de VMware
- SNMP agent
- Trampa SNMP
- Zabbix interno
- Zabbix trapper
- Comprobación externa
- Supervisión de bases de datos
- HTTP agent
  - Comprobación de Prometheus
- IPMI agent
- SSH agent
- Telnet agent
- JMX agent
- Item calculado

- Item dependiente
- Item de script
- Item de navegador
- Consulta de telemetría

Los detalles de todos los tipos de item se incluyen en las subpáginas de esta sección. Aunque los tipos de item ofrecen muchas opciones para recopilar datos, también existen opciones adicionales mediante parámetros de usuario o módulos cargables.

Algunas comprobaciones las realiza únicamente el Zabbix server (como supervisión sin agent), mientras que otras requieren Zabbix agent o incluso Zabbix Java gateway (con supervisión JMX).

> **Attention:**
> Si un tipo de item concreto requiere una interfaz específica (por ejemplo, una comprobación IPMI necesita una interfaz IPMI en el host), dicha interfaz debe existir en la definición del host.

En la definición del host se pueden configurar varias interfaces: Zabbix agent, SNMP agent, JMX e IPMI. Si un item puede utilizar más de una interfaz, buscará entre las interfaces disponibles del host (en el siguiente orden: Agent→SNMP→JMX→IPMI) la primera adecuada con la que vincularse.

Todos los items que devuelven texto (tipos de información carácter, registro y texto) también pueden devolver únicamente espacios en blanco, cuando corresponda, estableciendo el valor devuelto como una cadena vacía (compatible desde la versión 2.0).

Zabbix agent

Descripción general

Esta sección proporciona detalles sobre las claves de item que utilizan la comunicación con el agent de Zabbix para la recopilación de datos.

Hay comprobaciones de agent pasivas y activas. Al configurar un item, puede seleccionar el tipo requerido:

- *Zabbix agent* - para comprobaciones pasivas
- *Zabbix agent (active)* - para comprobaciones activas

Tenga en cuenta que todas las claves de item compatibles con Zabbix agent también son compatibles con la nueva generación de Zabbix agent 2. Consulte las claves de item adicionales que puede usar solo con agent 2.

Claves de item compatibles

A continuación se indican las claves de item que puede utilizar con Zabbix agent.

Las claves de item se muestran sin parámetros ni información adicional.
Haga clic en la clave de item para consultar todos los detalles.

Clave de item	Descripción	Grupo de items
kernel.maxfiles	El número máximo de archivos abiertos compatible con el sistema operativo.	Kernel
kernel.maxproc	El número máximo de procesos compatible con el sistema operativo.	
kernel.openfiles	El número de descriptores de archivo abiertos actualmente.	
log	La supervisión de un archivo de registro.	Supervisión de registros
log.count	El número de líneas coincidentes en un archivo de registro supervisado.	
logrt	La supervisión de un archivo de registro rotado.	
logrt.count	El número de líneas coincidentes en un archivo de registro supervisado que se ha rotado.	
modbus.get	Lee datos de Modbus.	Modbus
net.dns	Comprueba el estado de un servicio DNS.	Red
net.dns.perf	Comprueba el rendimiento de un servicio DNS.	
net.dns.record	Realiza una consulta DNS.	
net.if.collisions	El número de colisiones fuera de ventana.	
net.if.discovery	La lista de interfaces de red.	
net.if.get	La lista de interfaces de red con información detallada.	
net.if.in	Las estadísticas del tráfico entrante en una interfaz de red.	
net.if.out	Las estadísticas del tráfico saliente en una interfaz de red.	
net.if.total	La suma de las estadísticas del tráfico entrante y saliente en una interfaz de red.	
net.tcp.listen	Comprueba si este puerto TCP está en estado LISTEN.	

Clave de item	Descripción	Grupo de items
net.tcp.port	Comprueba si es posible establecer una conexión TCP con el puerto especificado.	
net.tcp.service	Comprueba si un servicio está en ejecución y acepta conexiones TCP.	
net.tcp.service.perf	Comprueba el rendimiento de un servicio TCP.	
net.tcp.socket.count	Devuelve el número de sockets TCP que coinciden con los parámetros.	
net.udp.listen	Comprueba si este puerto UDP está en estado LISTEN.	
net.udp.service	Comprueba si un servicio está en ejecución y responde a solicitudes UDP.	
net.udp.service.perf	Comprueba el rendimiento de un servicio UDP.	
net.udp.socket.count	Devuelve el número de sockets UDP que coinciden con los parámetros.	
proc.cpu.util	El porcentaje de utilización de CPU del proceso.	Procesos
proc.get	La lista de procesos del sistema operativo y sus parámetros.	
proc.mem	La memoria utilizada por el proceso, en bytes.	
proc.num	El número de procesos.	
sensor	Lectura del sensor de hardware.	Sensores
system.boottime	La hora de arranque del sistema.	Sistema
system.cpu.discovery	La lista de CPU/núcleos de CPU detectados.	
system.cpu.intr	Las interrupciones del dispositivo.	
system.cpu.load	La carga de CPU.	
system.cpu.num	El número de CPU.	
system.cpu.switches	El número de cambios de contexto.	
system.cpu.util	El porcentaje de utilización de CPU.	
system.hostname	El nombre de host del sistema.	
system.hw.chassis	La información del chasis.	
system.hw.cpu	La información de la CPU.	
system.hw.devices	La lista de dispositivos PCI o USB.	
system.hw.macaddr	La lista de direcciones MAC.	
system.localtime	La hora del sistema.	
system.run	Ejecuta el comando especificado en el host.	
system.stat	Las estadísticas del sistema.	
system.sw.arch	La información sobre la arquitectura del software.	
system.sw.os	La información del sistema operativo.	
system.sw.os.get	Información detallada sobre el sistema operativo (versión, tipo, nombre de la distribución, versión menor y mayor, etc.).	
system.sw.packages	La lista de paquetes instalados.	
system.sw.packages.get	Una lista detallada de los paquetes instalados.	
system.swap.in	Las estadísticas de swap-in (del dispositivo a la memoria).	
system.swap.out	Las estadísticas de swap-out (de la memoria al dispositivo).	
system.swap.size	El tamaño del espacio de swap en bytes o como porcentaje del total.	
system.uname	La identificación del sistema.	
system.uptime	El tiempo de actividad del sistema en segundos.	
system.users.num	El número de usuarios conectados.	
vfs.dev.discovery	La lista de dispositivos de bloques y su tipo.	Sistemas de archivos virtuales
vfs.dev.get	La lista de dispositivos de bloques con información detallada.	
vfs.dev.read	Las estadísticas de lectura del disco.	
vfs.dev.write	Las estadísticas de escritura del disco.	
vfs.dir.count	El número de entradas del directorio.	
vfs.dir.get	La lista de entradas del directorio.	
vfs.dir.size	El tamaño del directorio.	
vfs.file.cksum	La suma de comprobación del archivo, calculada mediante el algoritmo UNIX cksum.	
vfs.file.contents	Recupera el contenido de un archivo.	
vfs.file.exists	Comprueba si el archivo existe.	
vfs.file.get	Devuelve información sobre un archivo.	
vfs.file.md5sum	La suma de comprobación MD5 del archivo.	
vfs.file.owner	Recupera el propietario de un archivo.	
vfs.file.permissions	Devuelve una cadena de 4 dígitos que contiene el número octal con los permisos UNIX.	
vfs.file.regexp	Recupera una cadena del archivo.	
vfs.file.regmatch	Busca una cadena en el archivo.	
vfs.file.size	El tamaño del archivo.	

Clave de item	Descripción	Grupo de items
vfs.file.time	La información de fecha y hora del archivo.	
vfs.fs.discovery	La lista de sistemas de archivos montados con su tipo y opciones de montaje.	
vfs.fs.get	La lista de sistemas de archivos montados con su tipo y opciones de montaje y, en el modo *full*, el espacio de disco disponible y las estadísticas de inodos.	
vfs.fs.inode	El número o porcentaje de inodos.	
vfs.fs.size	El espacio de disco en bytes o como porcentaje del total.	
vm.memory.size	El tamaño de la memoria en bytes o como porcentaje del total.	Memoria virtual
web.page.get	Obtiene el contenido de una página web.	Supervisión web
web.page.perf	El tiempo de carga de una página web completa.	
web.page.regexp	Busca una cadena en la página web.	
agent.hostmetadata	Los metadatos del host del agent.	Zabbix
agent.hostname	El nombre de host del agent.	
agent.ping	La comprobación de disponibilidad del agent.	
agent.variant	La variante de Zabbix agent (Zabbix agent o Zabbix agent 2).	
agent.version	La versión de Zabbix agent.	
zabbix.stats	Devuelve un conjunto de métricas internas de Zabbix server o proxy. Si se especifican `ip` y `port`, las métricas se recuperan de la instancia remota; de lo contrario, de la instancia local.	
zabbix.stats (queue)	Devuelve el número de items supervisados en la cola que presentan retraso en Zabbix server o proxy. Si se especifican `ip` y `port`, las métricas se recuperan de la instancia remota; de lo contrario, de la instancia local.	

Plataformas compatibles

Salvo que se especifique lo contrario en los detalles del elemento, los elementos del agente (y todos los parámetros) son compatibles con:

- **Linux**
- **FreeBSD**
- **Solaris**
- **HP-UX**
- **AIX**
- **MacOS X**
- **OpenBSD**
- **NetBSD**

Muchos elementos del agente también son compatibles con **Windows**. Consulte la página de elementos del agente de Windows para más detalles.

Detalles de la clave del elemento

Los parámetros sin corchetes angulares son obligatorios. Los parámetros marcados con corchetes angulares **< >** son opcionales.

kernel.maxfiles

 El número máximo de archivos abiertos soportados por el sistema operativo.
 Valor de retorno: *Entero*.
 Plataformas soportadas: Linux, FreeBSD, MacOS X, OpenBSD, NetBSD.

kernel.maxproc

 El número máximo de procesos soportados por el sistema operativo.
 Valor de retorno: *Entero*.
 Plataformas soportadas: Linux 2.6 y posteriores, FreeBSD, Solaris, MacOS X, OpenBSD, NetBSD.

kernel.openfiles

 El número de descriptores de archivos abiertos actualmente.
 Valor de retorno: *Entero*.
 Plataformas soportadas: Linux (la métrica puede funcionar en otras plataformas tipo UNIX).

log[file,<regexp>,<encoding>,<maxlines>,<mode>,<output>,<maxdelay>,<options>,<persistent dir>]

 La supervisión de un archivo de registro.
 Valor devuelto: *Log*.
 Consulte las plataformas compatibles.

Parámetros:

- **file** - la ruta completa y el nombre de un archivo de registro;

- **regexp** - una expresión regular que describe el patrón requerido;
- **encoding** - el identificador de la página de códigos;
- **maxlines** - el número máximo de nuevas líneas por segundo que el agent enviará a Zabbix server o proxy. Este parámetro anula el valor de `MaxLinesPerSecond` en `zabbix_agentd.conf`;
- **mode** - valores posibles:
  - *all* (predeterminado);
  - *skip* - omite el procesamiento de datos antiguos (afecta solo a los items creados recientemente);
- **output** - una plantilla opcional de formato de salida. La secuencia de escape **\0** se reemplaza por la parte coincidente del texto (desde el primer carácter donde comienza la coincidencia hasta el carácter donde termina la coincidencia), mientras que una secuencia de escape **\N** (donde N=1...9) se reemplaza por el N-ésimo grupo coincidente (o una cadena vacía si N supera el número de grupos capturados);
- **maxdelay** - el retraso máximo en segundos. Tipo: float. Valores:
  - 0 - (predeterminado) nunca ignorar líneas del archivo de registro;
  - > 0.0 - ignorar líneas más antiguas para obtener las líneas más recientes analizadas dentro de los segundos de ”maxdelay”. ¡Lea las notas sobre maxdelay antes de usarlo!
- **options** - opciones adicionales:
mtime-noreread - registros no únicos, volver a leer solo si cambia el tamaño del archivo (ignorar el cambio de la hora de modificación). (Este parámetro está obsoleto desde 5.0.2, porque ahora se ignora mtime.)
- **persistent dir** (solo en zabbix_agentd en sistemas Unix; no compatible con Zabbix agent 2) - la ruta absoluta del directorio donde almacenar los archivos persistentes. Consulte también las notas adicionales sobre archivos persistentes.

Comentarios:

- El item debe configurarse como una comprobación activa.
- Si el archivo no existe o los permisos no permiten el acceso, el item pasa a no compatible.
- Si `output` se deja vacío, se devuelve toda la línea que contiene el texto coincidente. Tenga en cuenta que todos los tipos de expresiones regulares globales, excepto *Matches regular expression*, siempre devuelven la línea completa coincidente y el parámetro `output` se ignora.
- La extracción de contenido mediante el parámetro `output` se realiza en el agent.

Ejemplos:

```
log[/var/log/syslog]
log[/var/log/syslog,error]
log[/home/zabbix/logs/logfile,,,100]
```

Ejemplo de uso del parámetro `output` para extraer un número de un registro de log:

```
log[/app1/app.log,"task run [0-9.]+ sec, processed ([0-9]+) records, [0-9]+ errors",,,,\1] #this item will
```

Ejemplo de uso del parámetro `output` para reescribir un registro de log antes de enviarlo al server:

```
log[/app1/app.log,"([0-9 :-]+) task run ([0-9.]+) sec, processed ([0-9]+) records, ([0-9]+) errors",,,,"\1
```

log.count[file,<regexp>,<encoding>,<maxproclines>,<mode>,<maxdelay>,<options>,<persistent dir>]

 El número de líneas coincidentes en un archivo de registro supervisado.
 Valor de retorno: *Integer*.
 Consulte las plataformas compatibles.

Parámetros:

- **file** - la ruta completa y el nombre del archivo de registro;
- **regexp** - una expresión regular que describe el patrón requerido;
- **encoding** - el identificador de la página de códigos;
- **maxproclines** - el número máximo de nuevas líneas por segundo que el agent analizará (no puede superar 10000). El valor predeterminado es 10*`MaxLinesPerSecond` en `zabbix_agentd.conf`.
- **mode** - valores posibles:
  - *all* (predeterminado);
  - *skip* - omite el procesamiento de datos más antiguos (afecta solo a los items creados recientemente).
- **maxdelay** - el retraso máximo en segundos. Tipo: float. Valores:
  - 0 - (predeterminado) nunca ignora líneas del archivo de registro;
  - > 0.0 - ignora las líneas más antiguas para que las líneas más recientes se analicen dentro de los segundos de ”maxdelay”. ¡Lea las notas de maxdelay antes de usarlo!
- **options** - opciones adicionales:
mtime-noreread - registros no únicos, vuelve a leer solo si cambia el tamaño del archivo (ignora el cambio de la hora de modificación). (Este parámetro está obsoleto desde la versión 5.0.2, porque ahora se ignora mtime.)
- **persistent dir** (solo en `zabbix_agentd` en sistemas Unix; no compatible con Zabbix agent 2) - la ruta absoluta del directorio donde almacenar los archivos persistentes. Consulte también las notas adicionales sobre archivos persistentes.

Comentarios:

- El item debe configurarse como una comprobación activa.
- Las líneas coincidentes se cuentan en las nuevas líneas desde la última comprobación del registro por parte del agent, y por lo tanto dependen del intervalo de actualización del item.
- Si el archivo no existe o los permisos no permiten el acceso, el item pasa a no compatible.

logrt[file regexp,<regexp>,<encoding>,<maxlines>,<mode>,<output>,<maxdelay>,<options>,<persistent dir>]

 La supervisión de un archivo de log que se rota.
 Valor devuelto: *Log*.
 Consulte las plataformas compatibles.

Parámetros:

- **file regexp** - la ruta absoluta al archivo, con el nombre de archivo especificado mediante una expresión regular. Tenga en cuenta que la expresión regular se aplica solo al nombre del archivo y no necesita coincidir con el nombre completo (por ejemplo, `/path/to/agent` coincidirá con `zabbix_agentd.log`).
- **regexp** - una expresión regular que describe el patrón de contenido requerido.
- **encoding** - el identificador de la página de códigos identifier.
- **maxlines** - el número máximo de nuevas líneas por segundo que el agent enviará a Zabbix server o proxy. Este parámetro anula el valor de `MaxLinesPerSecond` en `zabbix_agentd.conf`.
- **mode** - valores posibles:
  - *all* (predeterminado);
  - *skip* - omite el procesamiento de datos antiguos (afecta solo a los items creados recientemente).
- **output** - una plantilla opcional de formato de salida. La secuencia de escape **\0** se reemplaza por la parte del texto coincidente (desde el primer carácter donde comienza la coincidencia hasta el carácter donde termina la coincidencia), mientras que una secuencia de escape **\N** (donde N=1...9) se reemplaza por el N-ésimo grupo coincidente (o una cadena vacía si N supera el número de grupos capturados).
- **maxdelay** - el retraso máximo en segundos. Tipo: float. Valores:
  - 0 - (predeterminado) nunca ignora las líneas del archivo de log;
  - > 0.0 - ignora las líneas más antiguas para obtener las líneas más recientes analizadas dentro de "maxdelay" segundos. Lea las notas de maxdelay antes de usarlo!
- **options** - el tipo de rotación del archivo de log y otras opciones. Valores posibles:
  - *rotate* (predeterminado).
  - *copytruncate* - tenga en cuenta que *copytruncate* no se puede usar junto con *maxdelay*. En este caso *maxdelay* debe ser 0 o no especificarse; consulte las notas de copytruncate.
  - *mtime-reread* - registros no únicos, volver a leer si cambia la hora de modificación o el tamaño (se usa de forma predeterminada si no se establece explícitamente ninguna opción `mtime-*`).
  - *mtime-noreread* - registros no únicos, volver a leer solo si cambia el tamaño (ignorar el cambio de la hora de modificación).
- **persistent dir** (solo en `zabbix_agentd` en sistemas Unix; no compatible con Zabbix agent 2) - la ruta absoluta del directorio donde almacenar los archivos persistentes. Consulte también las notas adicionales sobre archivos persistentes.

Comentarios:

- El item debe configurarse como una active check.
- La rotación del log se basa en la última hora de modificación de los archivos.
- Tenga en cuenta que `logrt` está diseñado para funcionar con un archivo de log actualmente activo, con varios otros archivos inactivos coincidentes rotados. Si, por ejemplo, un directorio tiene muchos archivos de log activos, debe crearse un item `logrt` independiente para cada uno. De lo contrario, si un item `logrt` recoge demasiados archivos, puede provocar agotamiento de memoria y un fallo de la supervisión.
- Si `output` se deja vacío, se devuelve toda la línea que contiene el texto coincidente. Tenga en cuenta que todos los tipos globales de expresiones regulares, excepto *Matches regular expression*, siempre devuelven la línea completa coincidente y el parámetro `output` se ignora.
- La extracción de contenido mediante el parámetro `output` se realiza en el agent.
- En el parámetro `file regexp`, la ruta del directorio de log y la expresión regular del archivo de log deben estar separadas por el separador de directorio correcto:
  - En Windows, el separador debe ser una barra invertida (\); las barras diagonales pueden tolerarse en otras posiciones, excepto en la que separa la ruta del directorio de log y la expresión regular del archivo de log (consulte los ejemplos a continuación).
  - En otros sistemas, debe ser una barra diagonal (/).

Ejemplos para Windows:

```
logrt["c:/dir1/dir2/dir3\filename.*\.log","pattern_to_match"] #this item will collect data from log files
logrt["//example.com/share/dir1/dir2/dir3\filename.*\.log","pattern_to_match"] #this item will collect dat
```

Ejemplos para otros sistemas:

```
logrt["/home/zabbix/logs/^logfile[0-9]{1,3}$",,,100] #this item will match a file like "logfile1" (will no
logrt["/home/user/^logfile_.*_[0-9]{1,3}$","pattern_to_match","UTF-8",100] #this item will collect data fr
```

Ejemplo de uso del parámetro output para extraer un número de un registro de log:

```
logrt[/app1/^test.*log$,"task run [0-9.]+ sec, processed ([0-9]+) records, [0-9]+ errors",,,,\1] #this ite
```

Ejemplo de uso del parámetro output para reescribir un registro de log antes de enviarlo a server:

```
logrt[/app1/^test.*log$,"([0-9 :-]+) task run ([0-9.]+) sec, processed ([0-9]+) records, ([0-9]+) errors",
```

logrt.count[file regexp,<regexp>,<encoding>,<maxproclines>,<mode>,<maxdelay>,<options>,<persistent dir>]

 El conteo de líneas coincidentes en un archivo de log monitoreado que está rotado.
 Valor devuelto: *Integer*.
 Consulte las plataformas compatibles.

Parámetros:

- **file regexp** - la ruta absoluta al archivo, con el nombre de archivo especificado mediante una expresión regular. Tenga en cuenta que la expresión regular se aplica solo al nombre del archivo y no necesita coincidir con el nombre completo (por ejemplo, `/path/to/agent` coincidirá con `zabbix_agentd.log`).
- **regexp** - una expresión regular que describe el patrón requerido.
- **encoding** - el identificador de la página de códigos.
- **maxproclines** - el número máximo de nuevas líneas por segundo que el agent analizará (no puede superar 10000). El valor predeterminado es 10*`MaxLinesPerSecond` en `zabbix_agentd.conf`.
- **mode** - valores posibles:
  - *all* (predeterminado);
  - *skip* - omite el procesamiento de datos más antiguos (afecta solo a los items creados recientemente).
- **maxdelay** - el retraso máximo en segundos. Tipo: float. Valores:
  - 0 - (predeterminado) nunca ignora líneas del archivo de log;
  - > 0.0 - ignora líneas más antiguas para que las líneas más recientes se analicen dentro de los segundos de ”maxdelay”. Lea las notas de `maxdelay` antes de usarlo!
- **options** - el tipo de rotación del archivo de log y otras opciones. Valores posibles:
  - *rotate* (predeterminado).
  - *copytruncate* - tenga en cuenta que `copytruncate` no puede usarse junto con `maxdelay`. En este caso `maxdelay` debe ser 0 o no especificarse; consulte las notas de `copytruncate`.
  - *mtime-reread* - registros no únicos, volver a leer si cambia la hora de modificación o el tamaño (predeterminado).
  - *mtime-noreread* - registros no únicos, volver a leer solo si cambia el tamaño (ignorar el cambio de la hora de modificación).
- **persistent dir** (solo en `zabbix_agentd` en sistemas Unix; no compatible con Zabbix agent 2) - la ruta absoluta del directorio donde almacenar los archivos persistentes. Consulte también las notas adicionales sobre archivos persistentes.

Comentarios:

- El item debe configurarse como una comprobación activa.
- Las líneas coincidentes se cuentan en las nuevas líneas desde la última comprobación del log por parte del agent, y por lo tanto dependen del intervalo de actualización del item.
- La rotación del log se basa en la última hora de modificación de los archivos.
- En el parámetro `file regexp`, la ruta del directorio del log y la expresión regular del archivo de log deben separarse con el separador de directorio correcto:
  - En Windows, el separador debe ser una barra invertida (\); las barras diagonales (/) pueden tolerarse en otras posiciones, excepto en la que separa la ruta del directorio del log y la expresión regular del archivo de log (consulte los ejemplos a continuación).
  - En otros sistemas, debe ser una barra diagonal (/).

Ejemplos para Windows:

```
logrt.count["c:/dir1/dir2/dir3\filename.*\.log","pattern_to_match"] #este item contará el número de coinci
logrt.count["//example.com/share/dir1/dir2/dir3\filename.*\.log","pattern_to_match"] #este item contará el
```

Ejemplos para otros sistemas:

```
logrt.count["/home/zabbix/logs/.*","^logfile[0-9]{1,3}$",,100] #este item contará el número de coincidenci
logrt.count["/home/user/^logfile_.*_[0-9]{1,3}$","pattern_to_match","UTF-8",100] #este item contará el núm
```

modbus.get[endpoint,<slave id>,<function>,<address>,<count>,<type>,<endianness>,<offset>]

 Lee datos Modbus.
 Valor devuelto: *cadena JSON*.
 Plataformas compatibles: Linux, Windows.

Parámetros:

- **endpoint** - el endpoint definido como `protocol://connection_string`;
- **slave id** - el ID del esclavo;
- **function** - la función Modbus;
- **address** - la dirección del primer registro, bobina o entrada;
- **count** - el número de registros a leer;
- **type** - el tipo de datos;
- **endianness** - la configuración de endianness;
- **offset** - el número de registros, comenzando desde `address`, cuyos resultados se descartarán.

En Windows, este item sigue rutas UNC en el parámetro `endpoint`, lo que puede suponer un riesgo de seguridad.

Consulte una descripción detallada de los parámetros.

net.dns[<ip>,name,<type>,<timeout>,<count>,<protocol>]

 Comprueba el estado de un servicio DNS.
 Valores devueltos: - 0 - la resolución DNS falló (el servidor DNS no respondió o devolvió un error); - 1 - la resolución DNS se realizó correctamente.
 Consulte las plataformas compatibles.

Parámetros:

- **ip** (se ignora en Windows salvo que se use Zabbix agent 2) - la dirección IP del servidor DNS (deje el campo vacío para usar el servidor DNS predeterminado);
- **name** - el nombre DNS que se va a consultar;
- **type** - el tipo de registro que se va a consultar (el valor predeterminado es `SOA`);
- **timeout** (se ignora en Windows salvo que se use Zabbix agent 2) - el tiempo de espera de la solicitud en segundos (el valor predeterminado es 1 segundo);
- **count** (se ignora en Windows salvo que se use Zabbix agent 2) - el número de intentos de la solicitud (el valor predeterminado es 2);
- **protocol** - el protocolo usado para realizar consultas DNS: *udp* (predeterminado) o *tcp*.

Comentarios:

- Los valores posibles de `type` son: *ANY*, *A*, *NS*, *CNAME*, *MB*, *MG*, *MR*, *PTR*, *MD*, *MF*, *MX*, *SOA*, *NULL*, *WKS* (no compatible con Zabbix agent en Windows, Zabbix agent 2 en todos los SO), *HINFO*, *MINFO*, *TXT*, *SRV*.
- Para búsquedas DNS inversas (cuando `type` está establecido en *PTR*), puede proporcionar el nombre DNS tanto en formato invertido como no invertido (consulte los ejemplos a continuación). Tenga en cuenta que, cuando se solicita un registro PTR, el nombre DNS en realidad es una dirección IP.
- Los nombres de dominio internacionalizados no son compatibles; utilice en su lugar nombres codificados en IDNA.

Ejemplos:

```
net.dns[198.51.100.1,example.com,MX,2,1]

net.dns[,198.51.100.1,PTR]
net.dns[,1.100.51.198.in-addr.arpa,PTR]

net.dns[,2a00:1450:400f:800::200e,PTR]
net.dns[,e.0.0.2.0.0.0.0.0.0.0.0.0.0.0.0.8.0.f.0.0.4.0.5.4.1.0.0.a.2.ip6.arpa,PTR]
```

net.dns.perf[<ip>,name,<type>,<timeout>,<count>,<protocol>]

 Comprueba el rendimiento de un servicio DNS.
 Valor devuelto: *Float* (0 - el servicio está caído; segundos - la cantidad de segundos empleados esperando una respuesta del servicio).
 Consulte plataformas compatibles.

Parámetros:

- **ip** (se ignora en Windows salvo que se use Zabbix agent 2) - la dirección IP del servidor DNS (deje vacío para usar el servidor DNS predeterminado);
- **name** - el nombre DNS que se va a consultar;
- **type** - el tipo de registro que se va a consultar (el valor predeterminado es `SOA`);
- **timeout** (se ignora en Windows salvo que se use Zabbix agent 2) - el tiempo de espera de la solicitud en segundos (el valor predeterminado es 1 segundo);
- **count** (se ignora en Windows salvo que se use Zabbix agent 2) - el número de intentos para la solicitud (el valor predeterminado es 2);
- **protocol** - el protocolo utilizado para realizar consultas DNS: *udp* (predeterminado) o *tcp*.

Comentarios:

- Los valores posibles para `type` son:
ANY, *A*, *NS*, *CNAME*, *MB*, *MG*, *MR*, *PTR*, *MD*, *MF*, *MX*, *SOA*, *NULL*, *WKS* (no compatible con Zabbix agent en Windows, Zabbix agent 2 en todos los SO), *HINFO*, *MINFO*, *TXT*, *SRV*.

- Para búsquedas DNS inversas (cuando `type` se establece en *PTR*), puede proporcionar el nombre DNS tanto en formato invertido como no invertido (consulte los ejemplos a continuación). Tenga en cuenta que, cuando se solicita un registro PTR, el nombre DNS en realidad es una dirección IP.
- Los nombres de dominio internacionalizados no son compatibles; utilice en su lugar nombres codificados en IDNA.
- El item devuelve un tiempo de respuesta en lugar de 0 cuando el servidor DNS responde con un código de error (por ejemplo, `NXDOMAIN` o `SERVFAIL`).

Ejemplos:

```
net.dns.perf[198.51.100.1,example.com,MX,2,1]

net.dns.perf[,198.51.100.1,PTR]
net.dns.perf[,1.100.51.198.in-addr.arpa,PTR]

net.dns.perf[,2a00:1450:400f:800::200e,PTR]
net.dns.perf[,e.0.0.2.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.8.0.f.0.0.4.0.5.4.1.0.0.a.2.ip6.arpa,PTR]
```

net.dns.record[<ip>,name,<type>,<timeout>,<count>,<protocol>]

 Realiza una consulta DNS.

Zabbix agent 2 también proporciona net.dns.get, que ofrece funciones adicionales, como más tipos de registros y mayor control sobre la supervisión DNS. Valor devuelto: una cadena de caracteres con el tipo de información requerido.
 Consulte las plataformas compatibles.

Parámetros:

- **ip** (se ignora en Windows salvo que se use Zabbix agent 2) - la dirección IP del servidor DNS (deje el campo vacío para usar el servidor DNS predeterminado);
- **name** - el nombre DNS que se va a consultar;
- **type** - el tipo de registro que se va a consultar (el valor predeterminado es `SOA`);
- **timeout** (se ignora en Windows salvo que se use Zabbix agent 2) - el tiempo de espera de la solicitud en segundos (el valor predeterminado es 1 segundo);
- **count** (se ignora en Windows salvo que se use Zabbix agent 2) - el número de intentos de la solicitud (el valor predeterminado es 2);
- **protocol** - el protocolo utilizado para realizar consultas DNS: *udp* (predeterminado) o *tcp*.

Comentarios:

- Los valores posibles para `type` son:
ANY, *A*, *NS*, *CNAME*, *MB*, *MG*, *MR*, *PTR*, *MD*, *MF*, *MX*, *SOA*, *NULL*, *WKS* (no compatible con Zabbix agent en Windows, Zabbix agent 2 en todos los SO), *HINFO*, *MINFO*, *TXT*, *SRV*.
- Para búsquedas DNS inversas (cuando `type` se establece en *PTR*), puede proporcionar el nombre DNS en formato invertido o no invertido (consulte los ejemplos a continuación). Tenga en cuenta que, cuando se solicita un registro PTR, el nombre DNS en realidad es una dirección IP.
- Los nombres de dominio internacionalizados no son compatibles; utilice en su lugar nombres codificados en IDNA.

Ejemplos:

```
net.dns.record[198.51.100.1,example.com,MX,2,1]

net.dns.record[,198.51.100.1,PTR]
net.dns.record[,1.100.51.198.in-addr.arpa,PTR]

net.dns.record[,2a00:1450:400f:800::200e,PTR]
net.dns.record[,e.0.0.2.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.8.0.f.0.0.4.0.5.4.1.0.0.a.2.ip6.arpa,PTR]
```

net.if.collisions[if]

 El número de colisiones fuera de ventana.
 Valor de retorno: *Entero*.
 Plataformas soportadas: Linux, FreeBSD, Solaris, AIX, MacOS X, OpenBSD, NetBSD. Se requieren privilegios de root en NetBSD.

Parámetro:

- **if** - nombre de la interfaz de red

net.if.discovery

 La lista de interfaces de red. Se utiliza para el descubrimiento de bajo nivel.
 Valor devuelto: *cadena JSON*.
 Plataformas compatibles: Linux, FreeBSD, Solaris, HP-UX, AIX, OpenBSD, NetBSD, Windows.

net.if.get[<regexp>]

 La lista de interfaces de red con información detallada.
 Se usa para el descubrimiento de bajo nivel.
 Este item es una versión ampliada del item de Zabbix agent `net.if.discovery`.
 Valor devuelto: *objeto JSON*.
 Plataformas compatibles: Linux.

Parámetros:

- **regexp** - una expresión regular que coincide con los nombres de las interfaces de red.

Comentarios:

- Devuelve un objeto JSON con las interfaces de red cuyos nombres coinciden con la expresión regular dada (ver más detalles).

net.if.in[if,<mode>]

 Las estadísticas de tráfico entrante en una interfaz de red.
 Valor devuelto: *Integer*.
 Plataformas compatibles: Linux, FreeBSD, Solaris[5], HP-UX, AIX, MacOS X, OpenBSD, NetBSD, Windows. Se requieren privilegios de root en NetBSD.

Parámetros:

- **if** - nombre de la interfaz de red (Unix); descripción completa de la interfaz o dirección IPv4; o, si está entre llaves, GUID de la interfaz de red (Windows);
- **mode** - valores posibles:
  - *bytes* - número de bytes (predeterminado);
  - *packets* - número de paquetes;
  - *errors* - número de errores;
  - *dropped* - número de paquetes descartados;
  - *overruns (fifo)* - número de errores del búfer FIFO;
  - *frame* - número de errores de encapsulación de paquetes;
  - *compressed* - número de paquetes comprimidos recibidos por el controlador del dispositivo;
  - *multicast* - número de tramas multicast recibidas por el controlador del dispositivo.

Comentarios:

- Puede usar esta clave con el paso de preprocesamiento *Change per second* para obtener las estadísticas de bytes por segundo.
- El modo *dropped* solo es compatible en Linux, FreeBSD, HP-UX, MacOS X, OpenBSD, NetBSD.
- Los modos *overruns*, *frame*, *compressed*, *multicast* solo son compatibles en Linux.
- En HP-UX este item no proporciona detalles sobre interfaces de loopback (por ejemplo, lo0).

Ejemplos:

```
net.if.in[eth0]
net.if.in[eth0,errors]
```

net.if.out[if,<mode>]

 Las estadísticas de tráfico saliente en una interfaz de red.
 Valor devuelto: *Integer*.
 Plataformas compatibles: Linux, FreeBSD, Solaris[5], HP-UX, AIX, MacOS X, OpenBSD, NetBSD, Windows. Se requieren privilegios de root en NetBSD.

Parámetros:

- **if** - nombre de la interfaz de red (Unix); descripción completa de la interfaz de red o dirección IPv4; o, si está entre llaves, GUID de la interfaz de red (Windows);
- **mode** - valores posibles:
  - *bytes* - número de bytes (predeterminado);
  - *packets* - número de paquetes;
  - *errors* - número de errores;
  - *dropped* - número de paquetes descartados;
  - *overruns (fifo)* - número de errores del búfer FIFO;
  - *collisions (colls)* - número de colisiones detectadas en la interfaz;
  - *carrier* - número de pérdidas de portadora detectadas por el controlador del dispositivo;
  - *compressed* - número de paquetes comprimidos transmitidos por el controlador del dispositivo.

Comentarios:

- Puede usar esta clave con el paso de preprocesamiento *Change per second* para obtener las estadísticas de bytes por segundo.
- El modo *dropped* solo es compatible en Linux, HP-UX.
- Los modos *overruns*, *collision*, *carrier*, *compressed* solo son compatibles en Linux.
- En HP-UX este item no proporciona detalles sobre interfaces de loopback (por ejemplo, lo0).

Ejemplos:

```
net.if.out[eth0]
net.if.out[eth0,errors]
```

net.if.total[if,<mode>]

 La suma de las estadísticas de tráfico entrante y saliente en una interfaz de red.
 Valor devuelto: *Integer*.
 Plataformas compatibles: Linux, FreeBSD, Solaris[5], HP-UX, AIX, MacOS X, OpenBSD, NetBSD, Windows. Se requieren privilegios de root en NetBSD.

Parámetros:

- **if** - nombre de la interfaz de red (Unix); descripción completa de la interfaz de red o dirección IPv4; o, si está entre llaves, GUID de la interfaz de red (Windows);
- **mode** - valores posibles:
  - *bytes* - número de bytes (predeterminado);
  - *packets* - número de paquetes;
  - *errors* - número de errores;
  - *dropped* - número de paquetes descartados;
  - *overruns (fifo)* - número de errores del búfer FIFO;
  - *collisions (colls)* - número de colisiones detectadas en la interfaz;
  - *compressed* - número de paquetes comprimidos transmitidos o recibidos por el controlador del dispositivo.

Comentarios:

- Puede usar esta clave con el paso de preprocesamiento *Change per second* para obtener las estadísticas de bytes por segundo.
- El modo *dropped* solo es compatible en Linux, HP-UX. Los paquetes descartados solo son compatibles si tanto `net.if.in` como `net.if.out` funcionan para paquetes descartados en su plataforma.
- Los modos *overruns*, *collision*, *compressed* solo son compatibles en Linux.
- En HP-UX este item no proporciona detalles sobre las interfaces de loopback (por ejemplo, lo0).

Ejemplos:

```
net.if.total[eth0]
net.if.total[eth0,errors]
```

net.tcp.listen[port]

 Comprueba si este puerto TCP está en estado LISTEN.
 Valores de retorno: 0 - no está en estado LISTEN; 1 - está en estado LISTEN.
 Plataformas compatibles: Linux, FreeBSD, Solaris, MacOS X, Windows.

Parámetro:

- **port** - número de puerto TCP

En kernels Linux 2.6.14 y posteriores, la información sobre sockets TCP en escucha se obtiene de la interfaz NETLINK del kernel, si es posible. De lo contrario, la información se recupera de los archivos `/proc/net/tcp` y `/roc/net/tcp6`.

Ejemplo:

```
net.tcp.listen[80]
```

net.tcp.port[<ip>,port]

 Comprueba si es posible establecer una conexión TCP con el puerto especificado.
 Valores devueltos: 0 - no se puede conectar; 1 - se puede conectar.
 Consulte plataformas compatibles.

Parámetros:

- **ip** - la dirección IP o el nombre DNS (el valor predeterminado es `127.0.0.1`);
- **port** - el número de puerto.

Comentarios:

- Para pruebas simples de rendimiento TCP, use `net.tcp.service.perf[tcp,<ip>,<port>]`.
- Estas comprobaciones pueden generar mensajes adicionales en los archivos de registro del daemon del sistema (normalmente se registran las sesiones SMTP y SSH).

Ejemplo:

```
net.tcp.port[,80] #this item can be used to test the web server availability running on port 80
```

net.tcp.service[service,<ip>,<port>]

 Comprueba si un servicio está en ejecución y acepta conexiones TCP.
 Valores devueltos: 0 - el servicio no está disponible; 1 - el servicio está en ejecución.
 Consulte las plataformas compatibles.

Parámetros:

- **service** - *ssh*, *ldap*, *smtp*, *ftp*, *http*, *pop*, *nntp*, *imap*, *tcp*, *https* o *telnet* (consulte los detalles);
- **ip** - la dirección IP o el nombre DNS (el valor predeterminado es `127.0.0.1`);
- **port** - el número de puerto (de forma predeterminada, se usa el número de puerto estándar del servicio).

Comentarios:

- Estas comprobaciones pueden generar mensajes adicionales en los archivos de registro del daemon del sistema (normalmente se registran sesiones SMTP y SSH).
- Actualmente no se admite la comprobación de protocolos cifrados (como IMAP en el puerto 993 o POP en el puerto 995). Como solución alternativa, utilice `net.tcp.port[]` para comprobaciones como estas.
- La comprobación de LDAP y HTTPS en Windows solo es compatible con Zabbix agent 2.
- La comprobación telnet busca un mensaje de inicio de sesión (':' al final).

Ejemplo:

```
net.tcp.service[ftp,,45] #this item can be used to test the availability of FTP server on TCP port 45
```

net.tcp.service.perf[service,<ip>,<port>]

 Comprueba el rendimiento de un servicio TCP.
 Valores devueltos: *Float* (0 - el servicio no está disponible; segundos - el número de segundos empleados esperando una respuesta del servicio).
 Consulte las plataformas compatibles.

Parámetros:

- **service** - *ssh*, *ldap*, *smtp*, *ftp*, *http*, *pop*, *nntp*, *imap*, *tcp*, *https* o *telnet* (consulte los detalles);
- **ip** - la dirección IP o el nombre DNS (el valor predeterminado es `127.0.0.1`);
- **port** - el número de puerto (de forma predeterminada, se usa el número de puerto estándar del servicio).

Comentarios:

- Actualmente no se admite la comprobación de protocolos cifrados (como IMAP en el puerto 993 o POP en el puerto 995). Como solución alternativa, use `net.tcp.service.perf[tcp,<ip>,<port>]` para comprobaciones como estas.
- La comprobación telnet busca un prompt de inicio de sesión (':' al final).

Ejemplo:

```
net.tcp.service.perf[ssh] #this item can be used to test the speed of initial response from the SSH server
```

net.tcp.socket.count[<laddr>,<lport>,<raddr>,<rport>,<state>]

 Devuelve el número de sockets TCP que coinciden con los parámetros.
 Valor devuelto: *Integer*.
 Plataformas compatibles: Linux, Windows.

Parámetros:

- **laddr** - la dirección IPv4/6 local o subred CIDR;
- **lport** - el número de puerto local o nombre de servicio;
- **raddr** - la dirección IPv4/6 remota o subred CIDR;
- **rport** - el número de puerto remoto o nombre de servicio;
- **state** - el estado de la conexión (*established*, *syn_sent*, *syn_recv*, *fin_wait1*, *fin_wait2*, *time_wait*, *close*, *close_wait*, *last_ack*, *listen*, *closing*).

Ejemplo:

```
net.tcp.socket.count[,80,,,established] #número de conexiones al puerto TCP local 80 en estado established
```

net.udp.listen[puerto]

 Comprueba si este puerto UDP está en estado LISTEN.
 Valores de retorno: 0 - no está en estado LISTEN; 1 - está en estado LISTEN.
 Plataformas compatibles: Linux, FreeBSD, Solaris, MacOS X.

Parámetro:

- **puerto** - número de puerto UDP

Ejemplo:

```
net.udp.listen[68]
```

net.udp.service[service,<ip>,<port>]

 Comprueba si un servicio está en ejecución y responde a solicitudes UDP.
 Valores devueltos: 0 - el servicio no está disponible; 1 - el servicio está en ejecución.
 Consulte las plataformas compatibles.

Parámetros:

- **service** - *ntp* (consulte los detalles);
- **ip** - la dirección IP o el nombre DNS (el valor predeterminado es 127.0.0.1);
- **port** - el número de puerto (de forma predeterminada, se usa el número de puerto estándar del servicio).

Ejemplo:

```
net.udp.service[ntp,,45] #this item can be used to test the availability of NTP service on UDP port 45
```

net.udp.service.perf[service,<ip>,<port>]

 Comprueba el rendimiento de un servicio UDP.
 Valores devueltos: *Float* (0 - el servicio no está disponible; segundos - la cantidad de segundos empleados esperando una respuesta del servicio).
 Consulte las plataformas compatibles.

Parámetros:

- **service** - *ntp* (consulte los detalles);
- **ip** - la dirección IP o el nombre DNS (el valor predeterminado es 127.0.0.1);
- **port** - el número de puerto (de forma predeterminada, se usa el número de puerto estándar del servicio).

Ejemplo:

```
net.udp.service.perf[ntp] #this item can be used to test response time from NTP service
```

net.udp.socket.count[<laddr>,<lport>,<raddr>,<rport>,<state>]

 Devuelve el número de sockets UDP que coinciden con los parámetros.
 Valor devuelto: *Integer*.
 Plataformas soportadas: Linux, Windows.

Parámetros:

- **laddr** - la dirección IPv4/6 local o subred CIDR;
- **lport** - el número de puerto local o nombre de servicio;
- **raddr** - la dirección IPv4/6 remota o subred CIDR;
- **rport** - el número de puerto remoto o nombre de servicio;
- **state** - el estado de la conexión (*established*, *unconn*).

Ejemplo:

```
net.udp.socket.count[,,,,established] #devuelve el número de sockets UDP en estado conectado
```

proc.cpu.util[<name>,<user>,<type>,<cmdline>,<mode>,<zone>]

 El porcentaje de utilización de CPU del proceso.
 Valor devuelto: *Float*.
 Plataformas compatibles: Linux, Solaris[6].

Parámetros:

- **name** - el nombre del proceso (el valor predeterminado es *all processes*);
- **user** - el nombre de usuario (el valor predeterminado es *all users*);
- **type** - el tipo de utilización de CPU: *total* (predeterminado), *user* o *system*;
- **cmdline** - filtra por línea de comandos (es una expresión regular);
- **mode** - el modo de recopilación de datos: *avg1* (predeterminado), *avg5* o *avg15*;
- **zone** - la zona de destino: *current* (predeterminado) o *all*. Este parámetro solo es compatible en Solaris.

Comentarios:

- El valor devuelto se basa en el porcentaje de utilización de un solo núcleo de CPU. Por ejemplo, la utilización de CPU de un proceso que usa completamente dos núcleos es 200%.
- Los datos de utilización de CPU del proceso son recopilados por un collector que admite un máximo de 1024 consultas únicas (por nombre, usuario y línea de comandos). Las consultas a las que no se ha accedido durante las últimas 24 horas se eliminan del collector.
- Al establecer el parámetro zone en *current* (o dejar el valor predeterminado), si el agent se ha compilado en un Solaris sin compatibilidad con zonas, pero se ejecuta en un Solaris más reciente donde sí se admiten zonas, entonces el agent devolverá NOTSUPPORTED (el agent no puede limitar los resultados solo a la zona actual). Sin embargo, *all* sí es compatible en este caso.

Ejemplos:

```
proc.cpu.util[,root] #utilización de CPU de todos los procesos que se ejecutan bajo el usuario "root"
proc.cpu.util[zabbix_server,zabbix] #utilización de CPU de todos los procesos zabbix_server que se ejecuta
```

proc.get[<name>,<user>,<cmdline>,<mode>]

 La lista de procesos del sistema operativo y sus parámetros. Puede usarse para el descubrimiento de bajo nivel.
 Valor devuelto: *cadena JSON*.
 Plataformas compatibles: Linux, FreeBSD, Windows, OpenBSD, NetBSD.

Parámetros:

- **name** - el nombre del proceso (valor predeterminado: *todos los procesos*).
- **user** - el nombre de usuario (valor predeterminado: *todos los usuarios*).
- **cmdline** - filtra por la línea de comandos (es una expresión regular). Este parámetro no es compatible con Windows; en otras plataformas no es compatible si mode está establecido en 'summary'.
- **mode** - valores posibles:
  - *process* (predeterminado);
  - *thread* (no compatible con NetBSD);
  - *summary*. Consulte una lista de parámetros de proceso devueltos para cada modo y sistema operativo.

Comentarios:

- Si no se puede recuperar un valor, por ejemplo, debido a un error (el proceso ya terminó, falta de permisos, fallo de la llamada al sistema), se devolverá −1.
- Consulte las notas sobre la selección de procesos con los parámetros `name` y `cmdline` (específico de Linux).

Ejemplos:

```
proc.get[zabbix_server,zabbix,,process] #lista de todos los procesos zabbix_server que se ejecutan bajo el
proc.get[java,,,thread] #lista de todos los procesos Java, devuelve una entrada por hilo
proc.get[,zabbix,,summary] #datos combinados para los procesos de cada tipo que se ejecutan bajo el usuari
```

proc.mem[<name>,<user>,<mode>,<cmdline>,<memtype>]

 La memoria utilizada por el proceso en bytes.
 Valor devuelto: - *Integer* - con `mode` como *max*, *min*, *sum*; - *Float* - con `mode` como *avg*
 Plataformas compatibles: Linux, FreeBSD, Solaris, AIX, OpenBSD, NetBSD.

Parámetros:

- **name** - el nombre del proceso (el valor predeterminado es *all processes*);
- **user** - el nombre de usuario (el valor predeterminado es *all users*);
- **mode** - valores posibles: *avg*, *max*, *min* o *sum* (predeterminado);
- **cmdline** - filtra por la línea de comandos (es una expresión regular);
- **memtype** - el tipo de memoria utilizada por el proceso

Comentarios:

- El parámetro `memtype` solo es compatible en Linux, FreeBSD, Solaris[6], AIX.
- Cuando varios procesos usan memoria compartida, la suma de la memoria utilizada por los procesos puede dar como resultado valores grandes y poco realistas.

Consulte las notas sobre la selección de procesos con los parámetros `name` y `cmdline` (específicas de Linux).

Cuando este item se invoca desde la línea de comandos y contiene un parámetro de línea de comandos (por ejemplo, usando el modo de prueba de agent: `zabbix_agentd -t proc.mem[,,,apache2]`), se contará un proceso adicional, ya que el agent se contará a sí mismo.

Ejemplos:

```
proc.mem[,root] #la memoria utilizada por todos los procesos que se ejecutan bajo el usuario "root"
proc.mem[zabbix_server,zabbix] #la memoria utilizada por todos los procesos zabbix_server que se ejecutan
proc.mem[,oracle,max,oracleZABBIX] #la memoria utilizada por el proceso que consume más memoria y que se e
```

proc.num[<name>,<user>,<state>,<cmdline>,<zone>]

 El número de procesos.
 Valor devuelto: *Integer*.
 Plataformas compatibles: Linux, FreeBSD, Solaris[6], HP-UX, AIX, OpenBSD, NetBSD, Windows.

Parámetros:

- **name** - el nombre del proceso (el valor predeterminado es *all processes*).
- **user** - el nombre de usuario (el valor predeterminado es *all users*).
- **state** - valores posibles:
  - *all* (predeterminado);
  - *disk* - suspensión ininterrumpible;
  - *run* - en ejecución;

    - *sleep* - suspensión interrumpible;
    - *trace* - detenido;
    - *zomb* - zombi.
- **cmdline** - filtra por la línea de comandos (es una expresión regular).
- **zone** - la zona de destino: *current* (predeterminado) o *all*. Este parámetro solo es compatible en Solaris.

Comentarios:

- Los parámetros de estado *disk* y *trace* solo son compatibles en Linux, FreeBSD, OpenBSD y NetBSD.
- En Windows, solo se admiten los parámetros `name` y `user`.
- Cuando este item se invoca desde la línea de comandos y contiene un parámetro de línea de comandos (por ejemplo, usando el modo de prueba de agent: `zabbix_agentd -t proc.num[,,,apache2]`), se contará un proceso adicional, ya que el agent se contará a sí mismo.
- Al establecer el parámetro `zone` en `current` (o el valor predeterminado), si el agent se compiló en un Solaris sin soporte para zonas, pero se ejecuta en un Solaris más reciente donde sí se admiten zonas, entonces el agent devolverá NOTSUPPORTED (el agent no puede limitar los resultados solo a la zona actual). Sin embargo, *all* sí es compatible en este caso.
- Consulte las notas sobre la selección de procesos con los parámetros `name` y `cmdline` (específico de Linux).

Ejemplos:

```
proc.num[,mysql] #el número de procesos que se ejecutan bajo el usuario mysql
proc.num[apache2,www-data] #el número de procesos apache2 que se ejecutan bajo el usuario www-data
proc.num[,oracle,sleep,oracleZABBIX] #el número de procesos en estado sleep que se ejecutan bajo Oracle y
```

sensor[device,sensor,<mode>]

 Lectura de un sensor de hardware.
 Valor devuelto: *Float*.
 Plataformas compatibles: Linux, OpenBSD.

Parámetros:

- **device** - el nombre del dispositivo;
- **sensor** - el nombre del sensor;
- **mode** - valores posibles: *avg*, *max* o *min*.

Comentarios:

- Lee `/proc/sys/dev/sensors` en Linux 2.4.
- Lee `/sys/class/hwmon` en Linux 2.6+. Consulte una descripción más detallada del item sensor en Linux.
- Lee el MIB `hw.sensors` en OpenBSD.
- En Linux 2.4, si especifica mode, entonces device y sensor se tratan como expresiones regulares.
- En Linux 2.6+ y OpenBSD, o si se omite mode, device y sensor se tratan literalmente.

Ejemplos:

```
sensor[w83781d-i2c-0-2d,temp1]
sensor[cpu0,temp0] #la temperatura de una CPU
sensor["cpu[0-2]$",temp,avg] #la temperatura media de las tres primeras CPU
```

system.boottime

 El tiempo de arranque del sistema.
 Valor devuelto: *Entero (timestamp Unix)*.
 Plataformas soportadas: Linux, FreeBSD, Solaris, MacOS X, OpenBSD, NetBSD.

system.cpu.discovery

 La lista de CPUs/núcleos de CPU detectados. Se utiliza para el descubrimiento de bajo nivel.
 Valor de retorno: *cadena JSON*.
 Consulte plataformas compatibles.

system.cpu.intr

 Las interrupciones del dispositivo.
 Valor de retorno: *Entero*.
 Plataformas soportadas: Linux, FreeBSD, Solaris, AIX, OpenBSD, NetBSD.

system.cpu.load[<cpu>,<mode>]

 La carga de la CPU.
 Valor de retorno: *Float*.
 Consulte las plataformas compatibles.

Parámetros:

- **cpu**: valores posibles: *all* (por defecto) o *percpu* (la carga total dividida por el número de CPUs en línea);
- **mode**: valores posibles: *avg1* (promedio de un minuto, por defecto), *avg5* o *avg15*.

Ejemplo:

```
system.cpu.load[,avg5]
```

system.cpu.num[<type>]

 El número de CPUs.
 Valor devuelto: *Integer*.
 Plataformas soportadas: Linux, FreeBSD, Solaris, HP-UX, AIX, MacOS X, OpenBSD, NetBSD, Windows.

Parámetro:

- **type** - valores posibles: *online* (por defecto) o *max*

El parámetro de tipo *max* solo es soportado en Linux, FreeBSD, Solaris, MacOS X.

Ejemplo:

```
system.cpu.num
```

system.cpu.switches

 El número de cambios de contexto.
 Valor de retorno: *Entero*.
 Plataformas soportadas: Linux, FreeBSD, Solaris, AIX, OpenBSD, NetBSD.

system.cpu.util[<cpu>,<type>,<mode>,<logical or physical>]

 El porcentaje de utilización de la CPU.
 Valor de retorno: *Float*.
 Plataformas soportadas: Linux, FreeBSD, Solaris, HP-UX, AIX, OpenBSD, NetBSD, Windows.

Parámetros:

- **cpu** - *<número de CPU>* o *all* (por defecto);
- **type** - valores posibles: *user* (por defecto), *idle*, *nice*, *system*, *iowait*, *interrupt*, *softirq*, *steal*, *spin* (en OpenBSD), *guest* (en kernels Linux 2.6.24 y superiores), o *guest_nice* (en kernels Linux 2.6.33 y superiores);
- **mode** - valores posibles: *avg1* (media de un minuto, por defecto), *avg5* o *avg15*;
- **logical or physical** - valores posibles: *logical* (por defecto) o *physical*. Este parámetro solo se soporta en AIX.

Comentarios:

- El parámetro de tipo *nice* solo se soporta en Linux, FreeBSD, HP-UX, OpenBSD, NetBSD.
- El parámetro de tipo *iowait* solo se soporta en Linux 2.6 y superiores, Solaris, AIX.
- El parámetro de tipo *interrupt* solo se soporta en Linux 2.6 y superiores, FreeBSD, OpenBSD.
- Los parámetros de tipo *softirq*, *steal*, *guest*, *guest_nice* solo se soportan en Linux 2.6 y superiores.
- Los parámetros de modo *avg5* y *avg15* se soportan en Linux, FreeBSD, Solaris, HP-UX, AIX, OpenBSD, NetBSD.

Ejemplo:

```
system.cpu.util[0,user,avg5]
```

system.hostname[<type>,<transform>]

 El nombre de host del sistema.
 Valor devuelto: *String*.
 Consulte plataformas compatibles.

Parámetros:

- **type** - valores posibles:
  - *netbios* (predeterminado en Windows);
  - *host* (predeterminado en Linux);
  - *shorthost* (devuelve la parte del nombre de host anterior al primer punto, una cadena completa para nombres sin puntos);
  - *fqdn* (devuelve el nombre de dominio completamente calificado).
- **transform** - valores posibles:
  - *none* (predeterminado);
  - *lower* (convierte a minúsculas).

El valor se obtiene tomando `nodename` de la salida de la API del sistema uname().

Ejemplos de valores devueltos:

```
system.hostname → linux-w7x1
system.hostname → example.com
system.hostname[shorthost] → example
system.hostname → WIN-SERV2008-I6
system.hostname[host] → Win-Serv2008-I6LonG
system.hostname[host,lower] → win-serv2008-i6long
system.hostname[fqdn,lower] → blog.zabbix.com
```

system.hw.chassis[<info>]

 La información del chasis.
 Valor de retorno: *Cadena*.
 Plataformas compatibles: Linux.

Parámetro:

- **info**: valores posibles: *full* (por defecto), *model*, *serial*, *type* o *vendor*

Comentarios:

- Esta clave de elemento depende de la disponibilidad de la tabla SMBIOS;
- Intentará leer la tabla DMI desde sysfs, si el acceso a sysfs falla, intentará leer directamente desde la memoria;
- Se requieren **permisos de root** porque el valor se obtiene leyendo desde sysfs o memoria.

Ejemplo:

```
system.hw.chassis[full] → Hewlett-Packard HP Pro 3010 Small Form Factor PC CZXXXXXXXX Desktop
```

system.hw.cpu[<cpu>,<info>]

 La información de la CPU.
 Valor de retorno: *Cadena* o *Entero*.
 Plataformas compatibles: Linux.

Parámetros:

- **cpu** - <*Número de CPU*> o *all* (por defecto);
- **info** - valores posibles: *full* (por defecto), *curfreq*, *maxfreq*, *model* o *vendor*.

Comentarios:

- Recopila información de `/proc/cpuinfo` y `/sys/devices/system/cpu/[cpunum]/cpufreq/cpuinfo_max_freq`;
- Si se especifica un número de CPU y *curfreq* o *maxfreq*, se devuelve un valor numérico (Hz).

Ejemplo:

```
system.hw.cpu[0,vendor] → AuthenticAMD
```

system.hw.devices[<type>]

 La lista de dispositivos PCI o USB.
 Valor de retorno: *Texto*.
 Plataformas compatibles: Linux.

Parámetro:

- **type** - *pci* (por defecto) o *usb*

Devuelve la salida de la utilidad lspci o lsusb (ejecutada sin ningún parámetro).

Ejemplo:

```
system.hw.devices → 00:00.0 Host bridge: Advanced Micro Devices [AMD] RS780 Host Bridge
```

system.hw.macaddr[<interface>,<format>]

 Listado de direcciones MAC.
 Valor devuelto: *String*.
 Plataformas compatibles: Linux.

Parámetros:

- **interface** - *all* (predeterminado) o una expresión regular;
- **format** - *full* (predeterminado) o *short*.

Comentarios:

- Enumera las direcciones MAC de las interfaces cuyo nombre coincide con la expresión regular dada en `interface` (*all* enumera todas las interfaces).
- Si `format` se especifica como *short*, no se enumeran los nombres de las interfaces ni las direcciones MAC idénticas.

Ejemplo:

```
system.hw.macaddr["eth0$",full] → [eth0] 00:11:22:33:44:55
```

system.localtime[<type>]

 La hora del sistema.
 Valor devuelto: *Integer* - con `type` como *utc*; *String* - con `type` como *local*.
 Consulte las plataformas compatibles.

Parámetros:

- **type** - valores posibles:
  - *utc* - (predeterminado) el tiempo transcurrido desde la Época (00:00:00 UTC, 1 de enero de 1970), medido en segundos;
  - *local* - la hora en el formato 'yyyy-mm-dd,hh:mm:ss.nnn,+hh:mm'.

Debe usarse como una comprobación pasiva para Zabbix agent; en Zabbix agent 2 puede configurarse como una comprobación activa.

Ejemplo:

```
system.localtime[local] #cree un item usando esta clave y luego úselo para mostrar la hora del host en el
```

system.run[command,<mode>]

 Ejecuta el comando especificado en el host.
 Valor devuelto: resultado de *Text* del comando o 1 - con `mode` como *nowait* (independientemente del resultado del comando).
 Consulte las plataformas compatibles.

Parámetros:

- **command** - comando para ejecución;
- **mode** - valores posibles:
  - *wait* - esperar al final de la ejecución (predeterminado);
  - *nowait* - no esperar.

Comentarios:

- Este item está deshabilitado de forma predeterminada. Aprenda cómo habilitarlos.
- El valor devuelto del item es la salida estándar junto con el error estándar producido por el comando. No se realiza comprobación del código de salida.
- Para procesarse correctamente, el valor devuelto del comando debe ser de tipo de datos `text`. También se permite un resultado vacío.
- El valor devuelto está limitado a 16 MB (incluido el espacio en blanco final, que se trunca); también se aplican los límites de la base de datos.
- Véase también: Ejecución de comandos.

Ejemplo:

```
system.run[ls -l /] #return a detailed file list of the root directory
```

system.stat[resource,<type>]

 Las estadísticas del sistema.
 Valor devuelto: *Integer* o *float*.
 Plataformas compatibles: AIX.

Parámetros:

- **ent** - el número de unidades de procesador que esta partición tiene derecho a recibir (float).
- **kthr,<type>** - información sobre los estados de los hilos del kernel:
  - *r* - número promedio de hilos del kernel ejecutables (float);
  - *b* - número promedio de hilos del kernel colocados en la cola de espera del Virtual Memory Manager (float).
- **memory,<type>** - información sobre el uso de memoria virtual y real:
  - *avm* - páginas virtuales activas (integer);
  - *fre* - tamaño de la lista libre (integer).
- **page,<type>** - información sobre fallos de página y actividad de paginación:
  - *fi* - entradas de páginas de archivo por segundo (float);
  - *fo* - salidas de páginas de archivo por segundo (float);
  - *pi* - páginas paginadas desde el espacio de paginación (float);
  - *po* - páginas paginadas hacia el espacio de paginación (float);
  - *fr* - páginas liberadas (reemplazo de páginas) (float);
  - *sr* - páginas examinadas por el algoritmo de reemplazo de páginas (float).
- **faults,<type>** - tasa de traps e interrupciones:
  - *in* - interrupciones de dispositivo (float);
  - *sy* - llamadas al sistema (float);
  - *cs* - cambios de contexto de hilos del kernel (float).
- **cpu,<type>** - desglose del porcentaje de uso del tiempo de procesador:
  - *us* - tiempo de usuario (float);
  - *sy* - tiempo de sistema (float);
  - *id* - tiempo de inactividad (float);
  - *wa* - tiempo de inactividad durante el cual el sistema tenía solicitudes pendientes de E/S de disco/NFS (float);
  - *pc* - número de procesadores físicos consumidos (float);
  - *ec* - porcentaje de capacidad asignada consumida (float);
  - *lbusy* - indica el porcentaje de utilización de uno o varios procesadores lógicos que se produjo al ejecutarse en el nivel de usuario y sistema (float);
  - *app* - indica los procesadores físicos disponibles en el grupo compartido (float).
- **disk,<type>** - estadísticas de disco:

  - *bps* - indica la cantidad de datos transferidos (leídos o escritos) a la unidad en bytes por segundo (integer);
  - *tps* - indica el número de transferencias por segundo emitidas al disco/cinta físico (float).

Comentarios:

- Tenga en cuenta las siguientes limitaciones en estos items:
  - `system.stat[cpu,app]` - compatible solo en AIX LPAR de tipo "Shared";
  - `system.stat[cpu,ec]` - compatible en AIX LPAR de tipo "Shared" y "Dedicated" ("Dedicated" siempre devuelve 100 (percent));
  - `system.stat[cpu,lbusy]` - compatible solo en AIX LPAR de tipo "Shared";
  - `system.stat[cpu,pc]` - compatible en AIX LPAR de tipo "Shared" y "Dedicated";
  - `system.stat[ent]` - compatible en AIX LPAR de tipo "Shared" y "Dedicated".

system.sw.arch

 La información de la arquitectura del software.
 Valor de retorno: *Cadena*.
 Vea plataformas soportadas.

La información se obtiene de la función `uname()`.

Ejemplo:

```
system.sw.arch → i686
```

system.sw.os[<info>]

 La información del sistema operativo.
 Valor devuelto: *String*.
 Plataformas compatibles: Linux, Windows.

Parámetro:

- **info** - valores posibles: *full* (predeterminado), *short* o *name*

La información de info se obtiene de (ten en cuenta que no todos los archivos y opciones están presentes en todas las distribuciones):

- `/proc/version` (*full*) en Linux;
- `/proc/version_signature` (*short*) en Linux;
- el parámetro PRETTY_NAME de `/etc/os-release` en sistemas Linux que lo admiten o `/etc/issue.net` (*name*);
- la clave del registro `HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion` en Windows.

Ejemplos:

```
system.sw.os[short] → Ubuntu 2.6.35-28.50-generic 2.6.35.11
system.sw.os[full] → [s|Windows 10 Enterprise 22621.1.asd64fre.ni_release.220506-1250 Build 22621.963]
```

system.sw.os.get

 Información detallada sobre el sistema operativo (versión, tipo, nombre de la distribución, versión menor y mayor, etc).
 Valor de retorno: *cadena JSON*.
 Plataformas soportadas: Linux, Windows.

system.sw.packages[<regexp>,<manager>,<format>]

 El listado de paquetes instalados.
 Valor devuelto: *Text*.
 Plataformas compatibles: Linux.

Parámetros:

- **regexp** - *all* (predeterminado) o una expresión regular;
- **manager** - *all* (predeterminado) o un gestor de paquetes;
- **format** - *full* (predeterminado) o *short*.

Comentarios:

- Enumera (alfabéticamente) los paquetes instalados cuyo nombre coincide con la expresión regular dada (*all* los enumera todos).
- Gestores de paquetes compatibles (comando ejecutado):
  - pkg (`dpkg --get-selections`);
  - pkgtool (`ls /var/log/packages`);
  - rpm (`rpm -qa`);
  - pacman (`pacman -Q`);
  - portage.
- Si `format` se especifica como *full*, los paquetes se agrupan por gestores de paquetes (cada gestor en una línea separada que comienza con su nombre entre corchetes).
- Si `format` se especifica como *short*, los paquetes no se agrupan y se enumeran en una sola línea.

Ejemplo:

```
system.sw.packages[mini,dpkg,short] → python-minimal, python2.6-minimal, ubuntu-minimal
```

system.sw.packages.get[<regexp>,<manager>]

 Un listado detallado de los paquetes instalados.
 Valor devuelto: *cadena JSON*.
 Plataformas compatibles: Linux.

Parámetros:

- **regexp** - *all* (predeterminado) o una expresión regular;
- **manager** - *all* (predeterminado) o un gestor de paquetes (valores posibles: *rpm*, *dpkg*, *pkgtool*, *pacman* o *portage*).

Comentarios:

- Devuelve JSON sin formato con los paquetes instalados cuyo nombre coincide con la expresión regular proporcionada.
- La salida es un array de objetos, cada uno de los cuales contiene las siguientes claves: name, manager, version, size, architecture, buildtime e installtime (consulte más detalles).

system.swap.in[<device>,<type>]

 Las estadísticas de entrada de swap (del dispositivo a la memoria).
 Valor devuelto: *Integer*.
 Plataformas compatibles: Linux, FreeBSD, OpenBSD.

Parámetros:

- **device** - especifica el dispositivo usado para el intercambio (solo Linux) o *all* (predeterminado);
- **type** - valores posibles:
  - *count* (número de entradas de swap, predeterminado en plataformas que no son Linux);
  - *sectors* (sectores intercambiados);
  - *pages* (páginas intercambiadas, predeterminado en Linux).

Comentarios:

- La fuente de esta información es:
  - `/proc/swaps`, `/proc/partitions`, `/proc/stat` (Linux 2.4);
  - `/proc/swaps`, `/proc/diskstats`, `/proc/vmstat` (Linux 2.6).
- Tenga en cuenta que *pages* solo funcionará si no se especificó device.
- El parámetro de tipo *sectors* solo es compatible en Linux.

Ejemplo:

```
system.swap.in[,pages]
```

system.swap.out[<device>,<type>]

 Las estadísticas de swap-out (de la memoria al dispositivo).
 Valor devuelto: *Integer*.
 Plataformas compatibles: Linux, FreeBSD, OpenBSD.

Parámetros:

- **device** - especifica el dispositivo usado para el intercambio (solo Linux) o *all* (predeterminado);
- **type** - valores posibles:
  - *count* (número de swapouts, predeterminado en plataformas que no son Linux);
  - *sectors* (sectores intercambiados);
  - *pages* (páginas intercambiadas, predeterminado en Linux).

Comentarios:

- La fuente de esta información es:
  - `/proc/swaps`, `/proc/partitions`, `/proc/stat` (Linux 2.4);
  - `/proc/swaps`, `/proc/diskstats`, `/proc/vmstat` (Linux 2.6).
- Tenga en cuenta que *pages* solo funcionará si no se especificó device.
- El parámetro de tipo *sectors* solo es compatible en Linux.

Ejemplo:

```
system.swap.out[,pages]
```

system.swap.size[<device>,<type>]

 El tamaño del espacio de intercambio en bytes o en porcentaje del total.
 Valor devuelto: *Integer* - para bytes; *Float* - para porcentaje.
 Plataformas compatibles: Linux, FreeBSD, Solaris, AIX, OpenBSD, Windows.

Parámetros:

- **device** - especifica el dispositivo usado para el intercambio (solo FreeBSD) o *all* (predeterminado);

- **type** - valores posibles:
  - *free* (espacio de intercambio libre, predeterminado);
  - *pfree* (espacio de intercambio libre, en porcentaje);
  - *pused* (espacio de intercambio usado, en porcentaje);
  - *total* (espacio de intercambio total);
  - *used* (espacio de intercambio usado).

Comentarios:

- Tenga en cuenta que *pfree*, *pused* no son compatibles en Windows si el tamaño de swap es 0.
- Si no se especifica device, Zabbix agent solo tendrá en cuenta los dispositivos de swap (archivos); la memoria física se ignorará. Por ejemplo, en sistemas Solaris, el comando `swap -s` incluye una parte de la memoria física y los dispositivos de swap (a diferencia de `swap -l`).

Ejemplo:

```
system.swap.size[,pfree] → porcentaje de espacio de swap libre
```

system.uname

 Identificación del sistema.
 Valor de retorno: *Cadena*.
 Consulte las plataformas compatibles.

Comentarios:

- En UNIX, el valor de este elemento se obtiene con la llamada al sistema uname();
- En Windows, el elemento devuelve la arquitectura del SO, mientras que en UNIX devuelve la arquitectura de la CPU.

Ejemplos:

```
system.uname → FreeBSD localhost 4.2-RELEASE FreeBSD 4.2-RELEASE #0: Mon Nov i386
system.uname → Windows ZABBIX-WIN 6.0.6001 Microsoft® Windows Server® 2008 Standard Service Pack 1 x86
```

system.uptime

 El tiempo de actividad del sistema en segundos.
 Valor devuelto: *Entero*.
 Plataformas soportadas: Linux, FreeBSD, Solaris, AIX, MacOS X, OpenBSD, NetBSD, Windows.

En la configuración del item, utilice las unidades **s** o **uptime** para obtener valores legibles.

system.users.num

 El número de usuarios conectados.
 Valor de retorno: *Entero*.
 Consulte plataformas compatibles.

El comando **who** se utiliza en el lado del agente para obtener el valor.

vfs.dev.discovery

 La lista de dispositivos de bloques y su tipo. Se utiliza para el descubrimiento de bajo nivel.
 Valor de retorno: *cadena JSON*.
 Plataformas compatibles: Linux.

vfs.dev.get[<mode>,<regexp>]

 La lista de dispositivos de bloque con información detallada. Se usa para el descubrimiento de bajo nivel.
 Valor devuelto: *cadena JSON*.
 Plataformas compatibles: Linux, AIX.

Parámetros:

- **mode** - valores posibles:
  - *disks* (predeterminado) - devuelve la lista de discos;
  - *disk_stats* - devuelve estadísticas para cada disco;
  - *devices* (solo Linux) - devuelve la lista de dispositivos de bloque (discos y particiones);
  - *device_stats* (solo Linux) - devuelve estadísticas para cada dispositivo (discos y particiones).
- **regexp** - una expresión regular usada para filtrar dispositivos de bloque por su nombre.

vfs.dev.read[<device>,<type>,<mode>]

 Las estadísticas de lectura de disco.
 Valor devuelto: - *Integer* - con `type` en *sectors*, *operations*, *bytes*; - *Float* - con `type` en *sps*, *ops*, *bps*.
 Plataformas compatibles: Linux, FreeBSD, Solaris, AIX, OpenBSD.

Parámetros:

- **device** - dispositivo de disco (el valor predeterminado es *all* [3]);
- **type** - valores posibles: *sectors*, *operations*, *bytes*, *sps*, *ops* o *bps* (*sps*, *ops*, *bps* significan: sectores, operaciones, bytes por segundo, respectivamente);

- **mode** - valores posibles: *avg1* (promedio de un minuto, predeterminado), *avg5* o *avg15*. Este parámetro solo es compatible con `type` en: sps, ops, bps.

Comentarios:

- Si se usa un intervalo de actualización de tres horas o más[2], este item siempre devolverá 0.
- Los parámetros de tipo *sectors* y *sps* solo son compatibles en Linux.
- El parámetro de tipo *ops* solo es compatible en Linux y FreeBSD.
- El parámetro de tipo *bps* solo es compatible en FreeBSD.
- El parámetro de tipo *bytes* solo es compatible en FreeBSD, Solaris, AIX, OpenBSD.
- El parámetro `mode` solo es compatible en Linux, FreeBSD.
- Puede usar nombres relativos de dispositivos (por ejemplo, `sda`) así como un prefijo opcional `/dev/` (por ejemplo, `/dev/sda`).
- Se admiten volúmenes lógicos LVM.
- Los valores predeterminados del parámetro 'type' para diferentes SO:
  - *AIX* - operations;
  - *FreeBSD* - bps;
  - *Linux* - sps;
  - *OpenBSD* - operations;
  - *Solaris* - bytes.
- *sps*, *ops* y *bps* en las plataformas compatibles están limitados a 1024 dispositivos (1023 individuales y uno para *all*).

Ejemplo:

```
vfs.dev.read[,operations]
```

vfs.dev.write[<device>,<type>,<mode>]

 Las estadísticas de escritura del disco.
 Valor devuelto: *Integer* - con `type` en *sectors*, *operations*, *bytes*; *Float* - con `type` en *sps*, *ops*, *bps*.
 Plataformas compatibles: Linux, FreeBSD, Solaris, AIX, OpenBSD.

Parámetros:

- **device** - dispositivo de disco (el valor predeterminado es *all* [3]);
- **type** - valores posibles: *sectors*, *operations*, *bytes*, *sps*, *ops* o *bps* (*sps*, *ops*, *bps* significan: sectores, operaciones, bytes por segundo, respectivamente);
- **mode** - valores posibles: *avg1* (promedio de un minuto, predeterminado), *avg5* o *avg15*. Este parámetro solo es compatible con `type` en: sps, ops, bps.

Comentarios:

- Si se usa un intervalo de actualización de tres horas o más[2], este item siempre devolverá 0.
- Los parámetros de tipo *sectors* y *sps* solo son compatibles en Linux.
- El parámetro de tipo *ops* solo es compatible en Linux y FreeBSD.
- El parámetro de tipo *bps* solo es compatible en FreeBSD.
- El parámetro de tipo *bytes* solo es compatible en FreeBSD, Solaris, AIX, OpenBSD.
- El parámetro `mode` solo es compatible en Linux, FreeBSD.
- Puede usar nombres relativos de dispositivos (por ejemplo, `sda`) así como un prefijo opcional `/dev/` (por ejemplo, `/dev/sda`).
- Se admiten volúmenes lógicos LVM.
- Los valores predeterminados del parámetro `type` para diferentes sistemas operativos:
  - *AIX* - operations
  - *FreeBSD* - bps
  - *Linux* - sps
  - *OpenBSD* - operations
  - *Solaris* - bytes
- *sps*, *ops* y *bps* en las plataformas compatibles están limitados a 1024 dispositivos (1023 individuales y uno para *all*).

Ejemplo:

```
vfs.dev.write[,operations]
```

vfs.dir.count[dir,<regex incl>,<regex excl>,<types incl>,<types excl>,<max depth>,<min size>,<max size>,<min age>,<max age>,<regex excl dir>]

 El número de entradas del directorio.
 Valor devuelto: *Integer*.
 Consulte las plataformas compatibles.

Parámetros:

- **dir**: la ruta absoluta al directorio;

- **regex incl**: una expresión regular que describe el patrón de nombre de la entidad (archivo, directorio, enlace simbólico) que se incluirá; se incluyen todas si está vacío (valor predeterminado);
- **regex excl**: una expresión regular que describe el patrón de nombre de la entidad (archivo, directorio, enlace simbólico) que se excluirá; no se excluye ninguna si está vacío (valor predeterminado);
- **types incl**: tipos de entradas de directorio que se contarán. Valores posibles:
  - *file*: archivo normal;
  - *dir*: subdirectorio;
  - *sym*: enlace simbólico;
  - *sock*: socket;
  - *bdev*: dispositivo de bloques;
  - *cdev*: dispositivo de caracteres;
  - *fifo*: FIFO;
  - *dev*: sinónimo de "bdev,cdev";
  - *all*: todos los tipos (predeterminado), es decir, "file,dir,sym,sock,bdev,cdev,fifo". Los tipos múltiples deben separarse con comas y escribirse entre comillas.
- **types excl**: tipos de entradas de directorio (consulte `types incl`) que NO se contarán. Si un tipo de entrada aparece tanto en `types incl` como en `types excl`, las entradas de directorio de este tipo NO se cuentan.
- **max depth**: la profundidad máxima de los subdirectorios que se recorrerán:
  - **-1** (predeterminado): ilimitada;
  - **0**: no se desciende a los subdirectorios.
- **min size**: el tamaño mínimo (en bytes) para que se cuente un archivo. Los archivos más pequeños no se contarán. Se pueden utilizar sufijos de memoria.
- **max size**: el tamaño máximo (en bytes) para que se cuente un archivo. Los archivos más grandes no se contarán. Se pueden utilizar sufijos de memoria.
- **min age**: la antigüedad mínima (en segundos) de la entrada de directorio para que se cuente. Las entradas más recientes no se contarán. Se pueden utilizar sufijos de tiempo.
- **max age**: la antigüedad máxima (en segundos) de la entrada de directorio para que se cuente. Las entradas con esa antigüedad o una antigüedad superior no se contarán (hora de modificación). Se pueden utilizar sufijos de tiempo.
- **regex excl dir**: una expresión regular que describe el patrón de nombre del directorio que se excluirá. Se excluirá todo el contenido del directorio (a diferencia de `regex_excl`).

Comentarios:

- En Windows, este item sigue las rutas UNC, lo que puede suponer un riesgo de seguridad.
- No se admiten variables de entorno, como `%APP_HOME%`, `$HOME` y `%TEMP%`.
- Los pseudo-directorios "." y ".." nunca se cuentan.
- Nunca se siguen los enlaces simbólicos para recorrer directorios.
- Tanto `regex incl` como `regex excl` se aplican a archivos y directorios al calcular el número de entradas, pero se ignoran al seleccionar los subdirectorios que se recorrerán (si `regex incl` es `(?i)^.+\.zip$` y `max depth` no está establecido, se recorrerán todos los subdirectorios, pero solo se contarán los archivos de tipo zip).
- El tiempo de ejecución está limitado por el valor de tiempo de espera de la configuración del agent (3 segundos). Dado que recorrer un directorio grande puede tardar más, no se devolverán datos y el item pasará a no compatible. No se devolverá un recuento parcial.
- Al filtrar por tamaño, solo los archivos normales tienen tamaños significativos. En Linux y BSD, los directorios también tienen tamaños distintos de cero (normalmente, unos pocos KB). Los dispositivos tienen tamaño cero; por ejemplo, el tamaño de `/dev/sda1` no refleja el tamaño de la partición correspondiente. Por lo tanto, al utilizar `<min_size>` y `<max_size>`, se recomienda especificar `<types_incl>` como "*file*" para evitar resultados inesperados.

Ejemplos:

```
vfs.dir.count[/dev] #monitors the number of devices in /dev (Linux)
vfs.dir.count["C:\Users\ADMINI~1\AppData\Local\Temp"] #monitors the number of files in a temporary directo
```

vfs.dir.get[dir,<regex incl>,<regex excl>,<types incl>,<types excl>,<max depth>,<min size>,<max size>,<min age>,<max age>,<regex excl dir>]

 La lista de entradas del directorio.
 Valor de retorno: *cadena JSON*.
 Consulte las plataformas compatibles.

Parámetros:

- **dir**: la ruta absoluta al directorio;
- **regex incl**: una expresión regular que describe el patrón del nombre de la entidad (archivo, directorio, enlace simbólico) que se incluirá; se incluyen todas si está vacío (valor predeterminado);
- **regex excl**: una expresión regular que describe el patrón del nombre de la entidad (archivo, directorio, enlace simbólico) que se excluirá; no se excluye ninguna si está vacío (valor predeterminado);
- **types incl**: tipos de entradas de directorio que se mostrarán. Valores posibles:
  - *file*: archivo normal;

  - *dir*: subdirectorio;
  - *sym*: enlace simbólico;
  - *sock*: socket;
  - *bdev*: dispositivo de bloques;
  - *cdev*: dispositivo de caracteres;
  - *fifo*: FIFO;
  - *dev*: sinónimo de "bdev,cdev";
  - *all*: todos los tipos (predeterminado), es decir, "file,dir,sym,sock,bdev,cdev,fifo". Los tipos múltiples deben separarse con comas y escribirse entre comillas.
- **types excl**: tipos de entradas de directorio (consulte `types incl`) que NO se mostrarán. Si un tipo de entrada está presente tanto en `types incl` como en `types excl`, las entradas de directorio de ese tipo NO se mostrarán.
- **max depth**: la profundidad máxima de los subdirectorios que se recorrerán:
  - **-1** (predeterminado): sin límite;
  - **0**: no se desciende a los subdirectorios.
- **min size**: el tamaño mínimo (en bytes) para que se muestre un archivo. Los archivos más pequeños no se mostrarán. Se pueden utilizar sufijos de memoria.
- **max size**: el tamaño máximo (en bytes) para que se muestre un archivo. Los archivos más grandes no se mostrarán. Se pueden utilizar sufijos de memoria.
- **min age**: la antigüedad mínima (en segundos) de la entrada de directorio que se mostrará. Las entradas más recientes no se mostrarán. Se pueden utilizar sufijos de tiempo.
- **max age**: la antigüedad máxima (en segundos) de la entrada de directorio que se mostrará. Las entradas con esa antigüedad o una antigüedad superior no se mostrarán (hora de modificación). Se pueden utilizar sufijos de tiempo.
- **regex excl dir**: una expresión regular que describe el patrón del nombre del directorio que se excluirá. Todo el contenido del directorio se excluirá (a diferencia de `regex excl`).

Comentarios:

- En Windows, este item sigue las rutas UNC, lo que puede suponer un riesgo de seguridad.
- No se admiten variables de entorno como `%APP_HOME%`, `$HOME` y `%TEMP%`.
- Los pseudodirectorios "." y ".." nunca se muestran.
- Los enlaces simbólicos nunca se siguen durante el recorrido de directorios.
- Tanto `regex incl` como `regex excl` se aplican a archivos y directorios al generar la lista de entradas, pero se ignoran al seleccionar los subdirectorios que se recorrerán (si `regex incl` es `(?i)^.+\.zip$` y `max depth` no está establecido, se recorrerán todos los subdirectorios, pero solo se contabilizarán los archivos de tipo zip).
- El tiempo de ejecución está limitado por el valor de tiempo de espera de la configuración del agent. Dado que recorrer directorios grandes puede tardar más, no se devolverán datos y el item pasará a ser no compatible. No se devolverá una lista parcial.
- Al filtrar por tamaño, solo los archivos normales tienen tamaños significativos. En Linux y BSD, los directorios también tienen tamaños distintos de cero (normalmente, unos pocos KB). Los dispositivos tienen tamaño cero; por ejemplo, el tamaño de `/dev/sda1` no refleja el tamaño de la partición correspondiente. Por lo tanto, al utilizar `min size` y `max size`, se recomienda especificar `types incl` como "*file*" para evitar resultados inesperados.

Ejemplos:

```
vfs.dir.get[/dev] #recupera la lista de dispositivos en /dev (Linux)
vfs.dir.get["C:\Users\ADMINI~1\AppData\Local\Temp"] #recupera la lista de archivos de un directorio tempor
```

vfs.dir.size[dir,<regex incl>,<regex excl>,<mode>,<max depth>,<regex excl dir>]

 El tamaño del directorio (en bytes).
 Valor devuelto: *Integer*.
 Plataformas compatibles: Linux, Windows. El item puede funcionar en otras plataformas tipo UNIX.

Parámetros:

- **dir** - la ruta absoluta al directorio;
- **regex incl** - una expresión regular que describe el patrón de nombre de la entidad (archivo, directorio, enlace simbólico) que se incluirá; incluye todo si está vacío (valor predeterminado);
- **regex excl** - una expresión regular que describe el patrón de nombre de la entidad (archivo, directorio, enlace simbólico) que se excluirá; no excluye nada si está vacío (valor predeterminado);
- **mode** - valores posibles:
  - *apparent* (predeterminado) - obtiene tamaños aparentes de los archivos en lugar del uso de disco (actúa como `du -sb dir`);
  - *disk* - obtiene el uso de disco (actúa como `du -s -B1 dir`). A diferencia del comando `du`, el item vfs.dir.size tiene en cuenta los archivos ocultos al calcular el tamaño del directorio (actúa como `du -sb .[^.]* *` dentro de dir).
- **max depth** - la profundidad máxima de subdirectorios a recorrer:
  - **-1** (predeterminado) - ilimitada;
  - **0** - no desciende a subdirectorios.

- **regex excl dir** - una expresión regular que describe el patrón de nombre del directorio a excluir. Todo el contenido del directorio será excluido (a diferencia de `regex excl`)

Comentarios:

- En Windows, este item sigue rutas UNC, lo que puede ser un riesgo de seguridad.
- Solo se calculan los directorios con al menos permiso de lectura para el usuario *zabbix*. Para los directorios con solo permiso de lectura, se calcula el tamaño del propio directorio. Los directorios con permisos de lectura y ejecución se calculan incluyendo el contenido.
- En directorios grandes o unidades lentas, este item puede agotar el tiempo de espera debido al ajuste Timeout en los archivos de configuración de agent y server/proxy. Aumente los valores de timeout según sea necesario.
- El límite de tamaño de archivo depende de large file support.

Ejemplos:

```
vfs.dir.size[/tmp,log] #calcula el tamaño de todos los archivos en /tmp que contienen 'log' en sus nombres
vfs.dir.size[/tmp,log,^.+\.old$] #calcula el tamaño de todos los archivos en /tmp que contienen 'log' en s
```

vfs.file.cksum[file,<mode>]

 La suma de verificación del archivo, calculada mediante el algoritmo UNIX cksum.
 Valor devuelto: *Integer* - con `mode` como *crc32*, *String* - con `mode` como *md5*, *sha256*.
 Consulte las plataformas compatibles.

Parámetros:

- **file** - la ruta completa al archivo;
- **mode** - *crc32* (predeterminado), *md5* o *sha256*.

Comentarios:

- En Windows, este item sigue rutas UNC, lo que puede suponer un riesgo de seguridad.
- El límite de tamaño de archivo depende de large file support.

Ejemplo:

```
vfs.file.cksum[/etc/passwd]
```

Ejemplos de valores devueltos (crc32/md5/sha256 respectivamente):

```
675436101
9845acf68b73991eb7fd7ee0ded23c44
ae67546e4aac995e5c921042d0cf0f1f7147703aa42bfbfb65404b30f238f2dc
```

vfs.file.contents[file,<encoding>]

 Recupera el contenido de un archivo[7].
 Valor devuelto: *Text*.
 Consulte las plataformas compatibles.

Parámetros:

- **file** - la ruta completa al archivo;
- **encoding** - el identificador de la página de códigos identifier.

Comentarios:

- En Windows, este item sigue rutas UNC, lo que puede suponer un riesgo de seguridad.
- El valor devuelto está limitado a 16 MB (incluido el espacio en blanco final, que se recorta); también se aplican los límites de la base de datos.
- Se devuelve una cadena vacía si el archivo está vacío o contiene solo caracteres LF/CR.
- La marca de orden de bytes (BOM) se excluye de la salida.

Ejemplo:

```
vfs.file.contents[/etc/passwd]
```

vfs.file.exists[file,<types incl>,<types excl>]

 Comprueba si el archivo existe.
 Valor de retorno: 0 - no encontrado; 1 - existe un archivo del tipo especificado.
 Consulte las plataformas compatibles.

Parámetros:

- **file** - la ruta completa al archivo;
- **types incl** - la lista de tipos de archivo a incluir, valores posibles:
  - *file* (archivo regular, valor predeterminado (si `types_excl` no está establecido));
  - *dir* (directorio);

  - *sym* (enlace simbólico);
  - *sock* (socket);
  - *bdev* (dispositivo de bloque);
  - *cdev* (dispositivo de caracteres);
  - *fifo* (FIFO);
  - *dev* (sinónimo de "bdev,cdev");
  - *all* (todos los tipos mencionados, valor predeterminado si `types_excl` está establecido).
- **types excl** - la lista de tipos de archivo a excluir, consulte `types_incl` para ver los valores posibles (de forma predeterminada no se excluye ningún tipo)

Comentarios:

- En Windows, este item sigue rutas UNC, lo que puede suponer un riesgo de seguridad.
- Varios tipos deben separarse con una coma y el conjunto completo debe ir entre comillas "".
- Si el mismo tipo está en ambos <types_incl> y <types_excl>, los archivos de ese tipo se excluyen.
- El límite de tamaño de archivo depende del soporte para archivos grandes.

Ejemplos:

```
vfs.file.exists[/tmp/application.pid]
vfs.file.exists[/tmp/application.pid,"file,dir,sym"]
vfs.file.exists[/tmp/application_dir,dir]
```

vfs.file.get[file]

 Devuelve información sobre un archivo.
 Valor devuelto: *cadena JSON*.
 Consulte plataformas compatibles.

Parámetro:

- **file** - la ruta completa al archivo

Comentarios:

- En Windows, este item sigue rutas UNC, lo que puede ser un riesgo de seguridad.
- Tipos de archivo compatibles en sistemas tipo UNIX: archivo regular, directorio, enlace simbólico, socket, dispositivo de bloque, dispositivo de caracteres, FIFO.
- El límite de tamaño de archivo depende de large file support.

Ejemplo:

```
vfs.file.get[/etc/passwd] #return a JSON with information about the /etc/passwd file (type, user, permissi
```

vfs.file.md5sum[file]

 La suma de comprobación MD5 del archivo.
 Valor devuelto: cadena de caracteres (hash MD5 del archivo).
 Consulte las plataformas compatibles.

Parámetro:

- **file** - la ruta completa al archivo

Comentarios:

- En Windows, este item sigue rutas UNC, lo que puede suponer un riesgo de seguridad.
- El límite de tamaño del archivo depende de la compatibilidad con archivos grandes.

Ejemplo:

```
vfs.file.md5sum[/usr/local/etc/zabbix_agentd.conf]
```

Ejemplo del valor devuelto:

```
b5052decb577e0fffd622d6ddc017e82
```

vfs.file.owner[file,<ownertype>,<resulttype>]

 Obtiene el propietario de un archivo.
 Valor devuelto: *String*.
 Consulte las plataformas compatibles.

Parámetros:

- **file** - la ruta completa al archivo;
- **ownertype** - *user* (predeterminado) o *group* (solo Unix);
- **resulttype** - *name* (predeterminado) o *id*; para id - devuelve uid/gid en Unix, SID en Windows.

Comentarios:

- En Windows, este item sigue rutas UNC, lo que puede suponer un riesgo de seguridad.
- El límite de tamaño de archivo depende del soporte para archivos grandes.

Ejemplos:

```
vfs.file.owner[/tmp/zabbix_server.log] #devuelve el propietario del archivo /tmp/zabbix_server.log
vfs.file.owner[/tmp/zabbix_server.log,,id] #devuelve el ID del propietario del archivo /tmp/zabbix_server.
```

vfs.file.permissions[file]

 Devuelve una cadena de 4 dígitos que contiene el número octal con los permisos UNIX.
 Valor de retorno: *Cadena*.
 Plataformas compatibles: Linux. El elemento puede funcionar en otras plataformas similares a UNIX.

Parámetros:

- **file**: la ruta completa al archivo

El límite de tamaño de archivo depende de la compatibilidad con archivos grandes.

Ejemplo:

```
vfs.file.permissions[/etc/passwd] #devuelve los permisos de /etc/passwd, por ejemplo, '0644'
```

vfs.file.regexp[file,regexp,<encoding>,<start line>,<end line>,<output>]

 Obtiene una cadena en el archivo[7].
 Valor devuelto: La línea que contiene la cadena coincidente, o según lo especificado por el parámetro opcional `output`.
 Consulte plataformas compatibles.

Parámetros:

- **file** - la ruta completa al archivo;
- **regexp** - una expresión regular que describe el patrón requerido;
- **encoding** - el identificador de la página de códigos;
- **start line** - el número de la primera línea que se va a buscar (por defecto, la primera línea del archivo);
- **end line** - el número de la última línea que se va a buscar (por defecto, la última línea del archivo);
- **output** - una plantilla opcional de formato de salida. La secuencia de escape **\0** se reemplaza por la parte coincidente del texto (desde el primer carácter donde comienza la coincidencia hasta el carácter donde termina la coincidencia), mientras que una secuencia de escape **\N** (donde N=1...9) se reemplaza por el N-ésimo grupo coincidente (o una cadena vacía si N supera el número de grupos capturados).

Comentarios:

- En Windows, este item sigue rutas UNC, lo que puede suponer un riesgo de seguridad.
- El límite de tamaño del archivo depende del soporte para archivos grandes.
- Solo se devuelve la primera línea coincidente.
- Se devuelve una cadena vacía si ninguna línea coincide con la expresión.
- La marca de orden de bytes (BOM) se excluye de la salida.
- La extracción de contenido mediante el parámetro `output` se realiza en el agent.

Ejemplos:

```
vfs.file.regexp[/etc/passwd,zabbix]
vfs.file.regexp[/path/to/some/file,"([0-9]+)$",,3,5,\1]
vfs.file.regexp[/etc/passwd,"^zabbix:.:([0-9]+)",,,,\1] → obteniendo el ID del usuario *zabbix*
```

vfs.file.regmatch[file,regexp,<encoding>,<start line>,<end line>]

 Buscar una cadena en el archivo[7].
 Valores devueltos: 0 - no se encontró coincidencia; 1 - encontrado.
 Consulte las plataformas compatibles.

Parámetros:

- **file** - la ruta completa al archivo;
- **regexp** - una expresión regular que describe el patrón requerido;
- **encoding** - el identificador de la página de códigos identifier;
- **start line** - el número de la primera línea a buscar (por defecto, la primera línea del archivo);
- **end line** - el número de la última línea a buscar (por defecto, la última línea del archivo).

Comentarios:

- En Windows, este item sigue rutas UNC, lo que puede suponer un riesgo de seguridad.
- El límite de tamaño del archivo depende del soporte para archivos grandes.
- Se ignora la marca de orden de bytes (BOM).

Ejemplo:

```
vfs.file.regmatch[/var/log/app.log,error]
```

vfs.file.size[file,<mode>]

 El tamaño del archivo (en bytes).
 Valor devuelto: *Integer*.
 Consulte las plataformas compatibles.

Parámetros:

- **file** - la ruta completa al archivo;
- **mode** - valores posibles:
  - *bytes* (predeterminado);
  - *lines* (también se cuentan las líneas vacías).

Comentarios:

- En Windows, este item sigue rutas UNC, lo que puede suponer un riesgo de seguridad.
- El archivo debe tener permisos de lectura para el usuario *zabbix*.
- El límite de tamaño del archivo depende del soporte para archivos grandes.

Ejemplo:

```
vfs.file.size[/var/log/syslog]
```

vfs.file.time[file,<mode>]

 La información de tiempo del archivo.
 Valor devuelto: *Integer* (marca de tiempo Unix).
 Consulte plataformas compatibles.

Parámetros:

- **file** - la ruta completa al archivo;
- **mode** - valores posibles:
  - *modify* (predeterminado) - la última vez que se modificó el contenido del archivo (predeterminado);
  - *access* - la última vez que se leyó el archivo;
  - *change* - la última vez que se cambiaron las propiedades del archivo.

Comentarios:

- En Windows, este item sigue rutas UNC, lo que puede ser un riesgo de seguridad.
- El límite de tamaño de archivo depende del soporte para archivos grandes.

Ejemplo:

```
vfs.file.time[/etc/passwd,modify]
```

vfs.fs.discovery

 La lista de sistemas de archivos montados con su tipo y opciones de montaje. Se utiliza para el descubrimiento de bajo nivel.
 Valor de retorno: *cadena JSON*.
 Plataformas soportadas: Linux, FreeBSD, Solaris, HP-UX, AIX, MacOS X, OpenBSD, NetBSD, Windows.

vfs.fs.get[<mode>,<mountpoint>]

 La lista de sistemas de archivos montados con su tipo y opciones de montaje y, en el modo *full*, el espacio de disco disponible y las estadísticas de inodos. Se puede utilizar para el descubrimiento de bajo nivel.
 Valor de retorno: *cadena JSON*.
 Plataformas compatibles: Linux, FreeBSD, Solaris, HP-UX, AIX, MacOS X, OpenBSD, NetBSD, Windows.

Parámetros:

- **mode** - valores posibles:
  - *full* (predeterminado) - propiedades del sistema de archivos con el espacio de disco y las estadísticas de inodos;
  - *short* - solo las propiedades del sistema de archivos, sin el espacio de disco ni las estadísticas de inodos;
- **mountpoint** - el punto de montaje cuyos datos se devolverán (por ejemplo, `/var` o `C:`). Si no se especifica, se devuelven todos los sistemas de archivos montados.

Comentarios:

- En el modo *short*, solo se devuelven el nombre, el tipo y las opciones de montaje del sistema de archivos (en Windows: nombre, tipo, etiqueta y tipo de unidad). No se consultan estadísticas de los sistemas de archivos; en sistemas similares a Unix solo se lee la lista de sistemas de archivos montados (por ejemplo, `/proc/mounts` en Linux), por lo que la comprobación no activa el automontaje ni se ve afectada por sistemas de archivos de red que no responden. Esto hace que el modo *short* sea adecuado para el descubrimiento de bajo nivel.

- El valor de **mountpoint** debe coincidir exactamente con la propiedad `fsname`. En Windows, la comparación no distingue entre mayúsculas y minúsculas; especifique el punto de montaje sin la barra inversa final (por ejemplo, `C:`).
- Si se especifica **mountpoint**, solo se devuelven los datos correspondientes a ese punto de montaje. Si ningún sistema de archivos montado coincide con él, se devuelve una matriz vacía (`[]`).
- La salida puede contener varias entradas con el mismo `fsname` si hay más de un sistema de archivos montado en el mismo directorio.
- En el modo *full*, el espacio de disco disponible y las estadísticas de inodos solo se devuelven si el usuario del Zabbix agent tiene permiso de recorrido en todos los directorios de la ruta del punto de montaje; de lo contrario, el sistema de archivos se omite de la salida. Esto no se aplica al modo *short*, que, por lo tanto, puede mostrar sistemas de archivos que faltan en la salida del modo *full*.
- Las propiedades `fslabel` y `fsdrivetype` solo se devuelven en Windows; las propiedades `inodes` y `options` no se devuelven en Windows.
- También se notifican los sistemas de archivos cuyo número de inodos es igual a cero, lo que puede ocurrir en sistemas de archivos con inodos dinámicos (por ejemplo, btrfs).
- Consulte también: Descubrimiento de sistemas de archivos montados.

Ejemplos:

```
vfs.fs.get #todos los sistemas de archivos montados con estadísticas; igual que vfs.fs.get[full]
vfs.fs.get[short] #todos los sistemas de archivos montados sin estadísticas
vfs.fs.get[full,/var] #estadísticas únicamente del punto de montaje /var
vfs.fs.get[short,C:] #propiedades únicamente del volumen C: (Windows)
```

vfs.fs.inode[fs,<modo>]

 El número o porcentaje de inodos.
 Valor de retorno: *Entero* - para número; *Flotante* - para porcentaje.
 Consulte las plataformas compatibles.

Parámetros:

- **fs** - el sistema de archivos;
- **modo** - valores posibles: *total* (por defecto), *free*, *used*, *pfree* (libres, porcentaje), o *pused* (usados, porcentaje).

Si el recuento de inodos es igual a cero, lo que puede ocurrir en sistemas de archivos con inodos dinámicos (por ejemplo, btrfs), los valores de pfree/pused se informarán como ”100” y ”0” respectivamente.

Ejemplo:

```
vfs.fs.inode[/,pfree]
```

vfs.fs.size[fs,<mode>]

 El espacio en disco en bytes o en porcentaje del total.
 Valor de retorno: *Entero* - para bytes; *Flotante* - para porcentaje.
 Consulte plataformas compatibles.

Parámetros:

- **fs**: el sistema de archivos;
- **mode**: valores posibles: *total* (por defecto), *free*, *used*, *pfree* (libre, porcentaje) o *pused* (usado, porcentaje).

Comentarios:

- Si el sistema de archivos no está montado, devuelve el tamaño de un sistema de archivos local donde se encuentra el punto de montaje;
- El espacio reservado de un sistema de archivos se tiene en cuenta y no se incluye al usar el modo *free*.

Ejemplo:

```
vfs.fs.size[/tmp,free]
```

vm.memory.size[<mode>]

 El tamaño de la memoria en bytes o en porcentaje del total.
 Valor de retorno: *Integer* - para bytes; *Float* - para porcentaje.
 Consulte las plataformas soportadas.

Parámetro:

- **mode** - valores posibles: *total* (por defecto), *active*, *anon*, *buffers*, *cached*, *exec*, *file*, *free*, *inactive*, *pinned*, *shared*, *slab*, *wired*, *used*, *pused* (usada, porcentaje), *available* o *pavailable* (disponible, porcentaje).

Comentarios:

- Este item acepta tres categorías de parámetros:
  - *total* - cantidad total de memoria;

    - tipos de memoria específicos de la plataforma: *active*, *anon*, *buffers*, *cached*, *exec*, *file*, *free*, *inactive*, *pinned*, *shared*, *slab*, *wired*;
    - estimaciones a nivel de usuario sobre cuánta memoria se utiliza y está disponible: *used*, *pused*, *available*, *pavailable*.
- El parámetro de modo *active* solo es compatible con FreeBSD, HP-UX, MacOS X, OpenBSD, NetBSD.
- Los parámetros de modo *anon*, *exec*, *file* solo son compatibles con NetBSD.
- El parámetro de modo *buffers* solo es compatible con Linux, FreeBSD, OpenBSD, NetBSD.
- El parámetro de modo *cached* solo es compatible con Linux, FreeBSD, AIX, OpenBSD, NetBSD.
- Los parámetros de modo *inactive*, *wired* solo son compatibles con FreeBSD, MacOS X, OpenBSD, NetBSD.
- El parámetro de modo *pinned* solo es compatible con AIX.
- El parámetro de modo *shared* solo es compatible con Linux 2.4, FreeBSD, OpenBSD, NetBSD.
- Consulte también detalles adicionales para este item.

Ejemplo:

```
vm.memory.size[pavailable]
```

web.page.get[host,<path>,<port>,<redirect_limit>]

 Obtiene el contenido de una página web.
 Valor devuelto: código fuente de la página web como texto (incluidos los encabezados).
 Consulte plataformas compatibles.

Parámetros:

- **host**: el nombre de host o la URL (como `scheme://host:port/path`, donde solo *host* es obligatorio). Esquemas de URL permitidos: *http*, *https*[4]. Si falta el esquema, se tratará como *http*. Si se especifica una URL, `path` y `port` deben estar vacíos. Especificar el nombre de usuario y la contraseña al conectarse a servers que requieren autenticación, por ejemplo: `http://user:password@www.example.com`, solo es posible con compatibilidad con cURL [4]. Se admite Punycode en los nombres de host.
- **path**: la ruta a un documento HTML (el valor predeterminado es /);
- **port**: el número de puerto (el valor predeterminado es 80 para HTTP);
- **redirect_limit**: el número máximo de redirecciones que se seguirán [4], de 0 a 1000 (el valor predeterminado es 0: no se siguen las redirecciones).

Comentarios:

- Este item pasa a no compatible si el recurso especificado en `host` no existe o no está disponible, o si `redirect_limit` es mayor que 0 y el número de redirecciones lo supera.
- `host` puede ser un nombre de host, un nombre de dominio o una dirección IPv4 o IPv6. Sin embargo, para las direcciones IPv6, el agent de Zabbix debe compilarse con la compatibilidad con IPv6 habilitada.
- Si se siguen redirecciones, el valor devuelto contiene los encabezados de cada respuesta de redirección, seguidos de los encabezados y el cuerpo de la respuesta final.

Ejemplos:

```
web.page.get[www.example.com,index.php,80]
web.page.get[https://www.example.com]
web.page.get[https://blog.example.com/?s=zabbix]
web.page.get[localhost:80]
web.page.get["[::1]/server-status"]
web.page.get[http://www.example.com,,,5]
```

web.page.perf[host,<path>,<port>,<redirect_limit>]

 El tiempo de carga de una página web completa (en segundos).
 Valor devuelto: *Float*.
 Consulte las plataformas compatibles.

Parámetros:

- **host**: el nombre de host o la URL (como `scheme://host:port/path`, donde solo *host* es obligatorio). Esquemas de URL permitidos: *http*, *https*[4]. Si falta el esquema, se tratará como *http*. Si se especifica una URL, `path` y `port` deben estar vacíos. Especificar el nombre de usuario y la contraseña al conectarse a servidores que requieren autenticación, por ejemplo: `http://user:password@www.example.com`, solo es posible con compatibilidad con cURL [4]. Punycode es compatible con los nombres de host.
- **path**: la ruta a un documento HTML (el valor predeterminado es /);
- **port**: el número de puerto (el valor predeterminado es 80 para HTTP)
- **redirect_limit**: el número máximo de redirecciones que se seguirán [4], de 0 a 1000 (el valor predeterminado es 0: no se siguen las redirecciones).

Comentarios:

- Este item deja de ser compatible si el recurso especificado en `host` no existe o no está disponible, o si `redirect_limit` es mayor que 0 y el número de redirecciones supera dicho límite.
- `host` puede ser un nombre de host, un nombre de dominio o una dirección IPv4 o IPv6. Sin embargo, para una dirección IPv6, el agent de Zabbix debe compilarse con la compatibilidad con IPv6 habilitada.
- Si se siguen redirecciones, el tiempo de carga incluye el tiempo empleado en todas las redirecciones.

Ejemplos:

```
web.page.perf[www.example.com,index.php,80]
web.page.perf[https://www.example.com]
web.page.perf[http://www.example.com,,,5]
```

web.page.regexp[host,<path>,<port>,regexp,<length>,<output>,<redirect_limit>]

 Busca una cadena en la página web.
 Valor de retorno: la cadena coincidente o lo especificado por el parámetro opcional `output`.
 Consulte plataformas compatibles.

Parámetros:

- **host**: el nombre de host o la URL (como `scheme://host:port/path`, donde solo *host* es obligatorio). Esquemas de URL permitidos: *http*, *https*[4]. Si falta el esquema, se tratará como *http*. Si se especifica una URL, `path` y `port` deben estar vacíos. Especificar el nombre de usuario y la contraseña al conectarse a servers que requieren autenticación, por ejemplo: `http://user:password@www.example.com`, solo es posible con compatibilidad con cURL [4]. Punycode es compatible con los nombres de host.
- **path**: la ruta a un documento HTML (el valor predeterminado es /).
- **port**: el número de puerto (el valor predeterminado es 80 para HTTP).
- **regexp**: una expresión regular que describe el patrón requerido.
- **length**: el número máximo de caracteres que se devolverán;
- **output**: una template opcional para dar formato a la salida. La secuencia de escape **\0** se reemplaza por la parte coincidente del texto (desde el primer carácter donde comienza la coincidencia hasta el carácter donde termina), mientras que una secuencia de escape **\N** (donde N=1...9) se reemplaza por el grupo coincidente N (o por una cadena vacía si N supera el número de grupos capturados).
- **redirect_limit**: el número máximo de redirecciones que se seguirán [4], de 0 a 1000 (el valor predeterminado es 0: no se siguen las redirecciones).

Comentarios:

- Este item pasa a no compatible si el recurso especificado en `host` no existe o no está disponible, o si `redirect_limit` es mayor que 0 y el número de redirecciones lo supera.
- `host` puede ser un nombre de host, un nombre de dominio o una dirección IPv4 o IPv6. Sin embargo, para una dirección IPv6, el agent de Zabbix debe compilarse con la compatibilidad con IPv6 habilitada.
- La extracción de contenido mediante el parámetro `output` se realiza en el agent.
- Si se siguen redirecciones, también se buscan los encabezados de cada respuesta de redirección.

Ejemplos:

```
web.page.regexp[www.example.com,index.php,80,OK,2]
web.page.regexp[https://www.example.com,,,OK,2]
web.page.regexp[http://www.example.com,,,OK,2,,5]
```

agent.hostmetadata

 Los metadatos del host del agent.
 Valor devuelto: *String*.
 Consulte plataformas compatibles.

Devuelve el valor de los parámetros `HostMetadata` o `HostMetadataItem`, o una cadena vacía si no se ha definido ninguno.

agent.hostname

 El nombre de host del agent.
 Valor devuelto: *String*.
 Consulte plataformas compatibles.

Devuelve:

- Como comprobación pasiva: el nombre del primer host listado en el parámetro Hostname del archivo de configuración del agent;
- Como comprobación activa: el nombre del hostname actual.

agent.ping

 La comprobación de disponibilidad del agent.
 Valor devuelto: Nada - no disponible; 1 - disponible.
 Consulte supported platforms.

Use la función de trigger `nodata()` para comprobar la indisponibilidad del host.

agent.variant

 La variante del agente Zabbix (agente Zabbix o agente Zabbix 2).
 Valor de retorno: 1 - agente Zabbix; 2 - agente Zabbix 2.
 Consulte las plataformas soportadas.

agent.version

 La versión del agente Zabbix.
 Valor de retorno: *Cadena*.
 Consulte las plataformas compatibles.

Ejemplo de valor devuelto:

```
6.0.3
```

zabbix.stats[<ip>,<port>]

 Devuelve un conjunto de métricas internas del server o proxy de Zabbix. Si se especifican `ip` y `port`, las métricas se obtienen de la instancia remota; de lo contrario, de la instancia local.
 Valor devuelto: *cadena JSON*.
 Consulte las plataformas compatibles.

Parámetros:

- **ip** - la lista de IP/DNS/máscara de red de los servers/proxies que se consultarán de forma remota (el valor predeterminado es `127.0.0.1`);
- **port** - el puerto del server/proxy que se consultará de forma remota (el valor predeterminado es `10051`)

Comentarios:

- Este item devuelve un conjunto seleccionado de métricas internas. Para obtener más detalles, consulte Supervisión remota de las estadísticas de Zabbix.
- Tenga en cuenta que la solicitud de estadísticas solo se aceptará desde las direcciones enumeradas en el parámetro `StatsAllowedIP` de server/proxy en la instancia de destino.

zabbix.stats[<ip>,<port>,queue,<from>,<to>]

 Devuelve el número de items monitorizados en la cola que presentan retraso en el server o proxy de Zabbix. Si se especifican `ip` y `port`, las métricas se obtienen de la instancia remota; de lo contrario, de la instancia local.
 Valor devuelto: *cadena JSON*.
 Consulte las plataformas compatibles.

Parámetros:

- **ip** - la lista de IP/DNS/máscara de red de servers/proxies que se consultarán de forma remota (el valor predeterminado es `127.0.0.1`);
- **port** - el puerto del server/proxy que se consultará de forma remota (el valor predeterminado es `10051`)
- **queue** - constante (debe usarse tal cual)
- **from** - retrasado al menos (el valor predeterminado es 6 segundos)
- **to** - retrasado como máximo (el valor predeterminado es infinito)

Tenga en cuenta que la solicitud de estadísticas solo se aceptará desde las direcciones enumeradas en el parámetro 'StatsAllowedIP' de server/proxy de la instancia de destino.

Notas al pie

[1] Una nota específica de Linux.
Zabbix agent debe tener acceso de solo lectura al sistema de archivos `/proc`.
Los parches del kernel de www.grsecurity.org limitan los derechos de acceso de los usuarios sin privilegios.

[2] `vfs.dev.read[]`, `vfs.dev.write[]`: Zabbix agent terminará las conexiones de dispositivos «obsoletas» si no se accede a los valores del item durante más de 3 horas.
Esto puede ocurrir si un sistema tiene dispositivos con rutas que cambian dinámicamente o si un dispositivo se elimina manualmente.
Tenga en cuenta también que estos items, si utilizan un intervalo de actualización de 3 horas o más, siempre devolverán «0».

[3] `vfs.dev.read[]`, `vfs.dev.write[]`: Si se utiliza el valor predeterminado `all` para el primer parámetro, la clave devolverá estadísticas resumidas, incluidos todos los dispositivos de bloques, como sda y sdb, y sus particiones (sda1, sda2, sdb3...), así como los dispositivos múltiples (RAID MD) basados en esos dispositivos de bloques o particiones y los volúmenes lógicos (LVM) basados en esos dispositivos de bloques o particiones.
En estos casos, los valores devueltos deben considerarse únicamente valores relativos (dinámicos en el tiempo), no valores absolutos.

[4] HTTPS y el seguimiento de redirecciones (`redirect_limit` mayor que 0) solo son compatibles si Zabbix agent se compila con compatibilidad con cURL; de lo contrario, el item pasa a ser no compatible. Zabbix agent 2 no requiere cURL para ninguna de estas funciones.

[5] Los valores `bytes` y `errors` no son compatibles con las interfaces de bucle invertido en sistemas Solaris hasta Solaris 10 6/06, inclusive, ya que las estadísticas de bytes, errores y utilización no se almacenan ni se notifican mediante el kernel.
Sin embargo, si supervisa un sistema Solaris mediante net-snmp, es posible que se devuelvan valores, ya que net-snmp incluye código heredado de cmu-snmp, con una antigüedad de hasta 1997, que, al no poder leer los valores de bytes de las estadísticas de la interfaz, devuelve el contador de paquetes (que sí existe en las interfaces de bucle invertido) multiplicado por un valor arbitrario de 308.
Esto supone que la longitud media de un paquete es de 308 octetos, una estimación muy aproximada, ya que el límite de MTU en los sistemas Solaris para las interfaces de bucle invertido es de 8892 bytes.
No debe suponerse que estos valores sean correctos ni siquiera aproximadamente precisos.
Son estimaciones aproximadas.
Zabbix agent no realiza estimaciones, pero net-snmp devolverá un valor para estos campos.

[6] La línea de comandos en Solaris, obtenida de `/proc/pid/psinfo`, está limitada a 80 bytes y contiene la línea de comandos tal como era cuando se inició el proceso.

[7] Los items `vfs.file.contents[]`, `vfs.file.regexp[]`, `vfs.file.regmatch[]` se pueden utilizar para recuperar el contenido de los archivos.
Si desea restringir el acceso a archivos específicos que contengan información confidencial, ejecute Zabbix agent con un usuario que no tenga permisos de acceso para ver estos archivos.

Uso con utilidades de línea de comandos

Tenga en cuenta que al probar o usar claves de ítem con zabbix_agentd o zabbix_get desde la línea de comandos, también debe considerar la sintaxis de la shell.

Por ejemplo, si un determinado parámetro de la clave debe ir entre comillas dobles, debe escapar explícitamente las comillas dobles; de lo contrario, la shell las eliminará como caracteres especiales y no se pasarán a la utilidad de Zabbix.

Ejemplos:

```
zabbix_agentd -t 'vfs.dir.count[/var/log,,,"file,dir",,0]'

zabbix_agentd -t vfs.dir.count[/var/log,,,\"file,dir\",,0]
```

Configuración de la codificación

Para asegurarse de que los datos adquiridos no se corrompan, puede especificar la codificación correcta para procesar la comprobación (por ejemplo, `vfs.file.contents`) en el parámetro `encoding`. La lista de codificaciones compatibles (identificadores de página de códigos) se encuentra en la documentación de libiconv (GNU Project) o en la documentación del SDK de Microsoft Windows para "Code Page Identifiers". Tenga en cuenta que Microsoft a veces marca algunas páginas de códigos como "disponibles únicamente para aplicaciones administradas", lo que significa que se exponen solo mediante el runtime de .NET y no están disponibles a través de las API nativas de Win32. Zabbix agent implementa su propia lógica de conversión de codificación; por lo tanto, esas páginas de códigos son compatibles con Zabbix agent incluso cuando las funciones nativas de Windows no las proporcionan.

Si no se especifica ninguna codificación en el parámetro `encoding`, se aplican las siguientes estrategias de resolución:

- Si no se especifica la codificación (o es una cadena vacía), se presupone que es UTF-8 y los datos se procesan "tal cual";
- Análisis de BOM: aplicable a los items `vfs.file.contents`, `vfs.file.regexp`, `vfs.file.regmatch`. Se intenta determinar la codificación correcta mediante la marca de orden de bytes (BOM) al principio del archivo. Si no hay BOM, se aplica la resolución estándar (consulte lo anterior).

Solución de problemas de los items del agent

En el caso de las comprobaciones pasivas, para evitar que el item no obtenga ningún valor porque la solicitud del server al agent agote el tiempo de espera primero, debe tenerse en cuenta lo siguiente:

- Cuando la versión del agent es anterior a la versión del server, es posible que el valor de `Timeout` en la configuración del item (o el tiempo de espera global) deba ser superior al valor de `Timeout` en el archivo de configuración del agent.
- Cuando la versión del agent es posterior a la versión del server, es posible que el valor de `Timeout` en el archivo de configuración del server deba ser superior al valor de `Timeout` en el archivo de configuración del agent.

Zabbix agent 2

Zabbix agent 2 admite todas las claves de item compatibles con Zabbix agent en Unix y Windows. Esta página proporciona detalles sobre las claves de item adicionales, que solo puede usar con Zabbix agent 2, agrupadas por el plugin al que pertenecen.

Las claves de item se enumeran sin parámetros ni información adicional. Haga clic en la clave de item para ver los detalles completos.

Item key	Description	Plugin
ceph.df.details	El uso de datos del clúster y su distribución entre pools.	Ceph (cargable)
ceph.osd.stats	Estadísticas agregadas y por OSD.	
ceph.osd.discovery	La lista de OSD descubiertos.	
ceph.osd.dump	Los umbrales de uso y los estados de los OSD.	
ceph.ping	Comprueba si se puede establecer una conexión con Ceph.	
ceph.pool.discovery	La lista de pools descubiertos.	
ceph.status	El estado general del clúster.	
docker.container_info	Información de bajo nivel sobre un contenedor.	Docker
docker.container_stats	Las estadísticas de uso de recursos del contenedor.	
docker.containers	Devuelve la lista de contenedores.	
docker.containers.discovery	Devuelve la lista de contenedores. Se usa para el descubrimiento de bajo nivel.	
docker.data.usage	Información sobre el uso actual de datos.	
docker.images	Devuelve la lista de imágenes.	
docker.images.discovery	Devuelve la lista de imágenes. Se usa para el descubrimiento de bajo nivel.	
docker.info	La información del sistema.	
docker.ping	Comprueba si el daemon de Docker está activo o no.	
ember.get	Devuelve el resultado del dispositivo requerido.	Ember+ (cargable)
memcached.ping	Comprueba si una conexión está activa o no.	Memcached
memcached.stats	Obtiene la salida del comando STATS.	
mongodb.cfg.discovery	Devuelve una lista de servidores de configuración descubiertos.	MongoDB (cargable)
mongodb.collection.stats	Devuelve diversas estadísticas de almacenamiento para una colección determinada.	
mongodb.collections.discovery	Devuelve una lista de colecciones descubiertas.	
mongodb.collections.usage	Devuelve las estadísticas de uso de las colecciones.	
mongodb.connpool.stats	Devuelve información sobre las conexiones salientes abiertas desde la instancia actual de la base de datos hacia otros miembros del clúster fragmentado o del conjunto de réplicas.	
mongodb.db.stats	Devuelve las estadísticas que reflejan el estado de un sistema de base de datos determinado.	
mongodb.db.discovery	Devuelve una lista de bases de datos descubiertas.	
mongodb.jumbo_chunks.count	Devuelve el número de jumbo chunks.	
mongodb.oplog.stats	Devuelve el estado del conjunto de réplicas, usando datos consultados del oplog.	
mongodb.ping	Comprueba si una conexión está activa o no.	
mongodb.rs.config	Devuelve la configuración actual del conjunto de réplicas.	
mongodb.rs.status	Devuelve el estado del conjunto de réplicas desde el punto de vista del miembro donde se ejecuta el método.	
mongodb.server.status	Devuelve el estado de la base de datos.	
mongodb.sh.discovery	Devuelve la lista de shards descubiertos presentes en el clúster.	
mongodb.version	Devuelve la versión del servidor de base de datos.	
mqtt.get	Se suscribe a un tema o a varios temas específicos (con comodines) del broker proporcionado y espera publicaciones.	MQTT
mssql.availability.group.get	Devuelve grupos de disponibilidad.	MSSQL (cargable)
mssql.custom.query	Devuelve el resultado de una consulta personalizada.	
mssql.db.get	Devuelve todas las bases de datos de MSSQL disponibles.	
mssql.job.status.get	Devuelve el estado de los trabajos.	
mssql.last.backup.get	Devuelve la hora del último respaldo para todas las bases de datos.	
mssql.local.db.get	Devuelve las bases de datos que participan en un grupo de disponibilidad Always On y réplica (primaria o secundaria) y que se encuentran en el servidor al que se estableció la conexión.	
mssql.mirroring.get	Devuelve información de mirroring.	
mssql.nonlocal.db.get	Devuelve las bases de datos que participan en un grupo de disponibilidad Always On y réplica (primaria o secundaria) ubicadas en otros servidores (la base de datos no es local a la instancia de SQL Server a la que se estableció la conexión).	
mssql.perfcounter.get	Devuelve los contadores de rendimiento.	
mssql.ping	Comprueba si una conexión está activa o no.	
mssql.quorum.get	Devuelve la información del quórum.	
mssql.quorum.member.get	Devuelve los miembros del quórum.	

Item key	Description	Plugin
mssql.replica.get	Devuelve las réplicas.	
mssql.version	Devuelve la versión de MSSQL.	
mysql.custom.query	Devuelve el resultado de una consulta personalizada.	MySQL
mysql.db.discovery	Devuelve la lista de bases de datos de MySQL.	
mysql.db.size	El tamaño de la base de datos en bytes.	
mysql.get_status_variables	Valores de las variables de estado globales.	
mysql.ping	Comprueba si una conexión está activa o no.	
mysql.replication.discovery	Devuelve la lista de réplicas de MySQL.	
mysql.replication.get_slave_status	El estado de la replicación.	
mysql.version	La versión de MySQL.	
net.dns.get	Realiza una consulta DNS y devuelve información detallada del registro DNS.	Network
nvml.device.count	El número de dispositivos GPU.	NVIDIA GPU (cargable)
nvml.device.decoder.utilization	Utilización del decodificador del dispositivo GPU.	
nvml.device.ecc.mode	Modo ECC del dispositivo GPU.	
nvml.device.encoder.stats.get	Estadísticas del codificador del dispositivo GPU.	
nvml.device.encoder.utilization	Utilización del codificador del dispositivo GPU.	
nvml.device.energy.consumption	Consumo de energía del dispositivo GPU.	
nvml.device.errors.memory	Información sobre errores de memoria ECC del dispositivo GPU.	
nvml.device.errors.register	Información sobre errores de registro ECC del dispositivo GPU.	
nvml.device.fan.speed.avg	Velocidad media del ventilador del dispositivo GPU.	
nvml.device.get	GPUs descubiertas con UUID y nombre.	
nvml.device.graphics.frequency	Velocidad del reloj gráfico del dispositivo GPU.	
nvml.device.memory.bar1.get	Información de memoria BAR1 del dispositivo GPU.	
nvml.device.memory.fb.get	Información de memoria framebuffer del dispositivo GPU.	
nvml.device.memory.frequency	Velocidad del reloj de memoria del dispositivo GPU.	
nvml.device.pci.utilization	Utilización PCI del dispositivo GPU.	
nvml.device.performance.state	Estado de rendimiento del dispositivo GPU.	
nvml.device.power.limit	Límite de energía del dispositivo GPU.	
nvml.device.power.usage	Uso de energía del dispositivo GPU.	
nvml.device.serial	Número de serie del dispositivo GPU.	
nvml.device.sm.frequency	Velocidad del reloj del multiprocesador de streaming del dispositivo GPU.	
nvml.device.temperature	Temperatura del dispositivo GPU.	
nvml.device.utilization	Estadísticas de utilización del dispositivo GPU.	
nvml.device.video.frequency	Velocidad del reloj de video del dispositivo GPU.	
nvml.system.driver.version	La versión del controlador NVIDIA.	
nvml.version	La versión de la biblioteca NVML.	
oracle.diskgroups.stats	Devuelve las estadísticas de los grupos de discos de Automatic Storage Management (ASM).	Oracle
oracle.diskgroups.discovery	Devuelve la lista de grupos de discos ASM.	
oracle.archive.info	Las estadísticas de los archivos de archivo.	
oracle.cdb.info	La información de las Container Databases (CDBs).	
oracle.custom.query	El resultado de una consulta personalizada.	
oracle.datafiles.stats	Devuelve las estadísticas de los archivos de datos.	
oracle.db.discovery	Devuelve la lista de bases de datos.	
oracle.fra.stats	Devuelve las estadísticas de Fast Recovery Area (FRA).	
oracle.instance.info	Las estadísticas de la instancia.	
oracle.pdb.info	La información de las Pluggable Databases (PDBs).	
oracle.pdb.discovery	Devuelve la lista de PDBs.	
oracle.pga.stats	Devuelve las estadísticas de Program Global Area (PGA).	
oracle.ping	Comprueba si se puede establecer una conexión con Oracle.	
oracle.proc.stats	Devuelve las estadísticas de los procesos.	
oracle.redolog.info	La información del archivo de registro desde el archivo de control.	
oracle.sga.stats	Devuelve las estadísticas de System Global Area (SGA).	
oracle.sessions.stats	Devuelve las estadísticas de las sesiones.	
oracle.sys.metrics	Devuelve un conjunto de valores de métricas del sistema.	
oracle.sys.params	Devuelve un conjunto de valores de parámetros del sistema.	
oracle.ts.stats	Devuelve las estadísticas de los tablespaces.	
oracle.ts.discovery	Devuelve una lista de tablespaces.	
oracle.user.info	Devuelve información de usuario de Oracle.	

Item key	Description	Plugin
oracle.version	Devuelve la versión del servidor de base de datos.	
pgsql.autovacuum.count	El número de trabajadores de autovacuum.	PostgreSQL (cargable)
pgsql.archive	La información sobre archivos archivados.	
pgsql.bgwriter	El número combinado de checkpoints para el clúster de bases de datos, desglosado por tipo de checkpoint.	
pgsql.cache.hit	La tasa de aciertos de la caché de búfer de PostgreSQL.	
pgsql.connections	Devuelve conexiones por tipo.	
pgsql.custom.query	Devuelve el resultado de una consulta personalizada.	
pgsql.db.age	La antigüedad del FrozenXID más antiguo de la base de datos.	
pgsql.db.bloating_tables	El número de tablas infladas por base de datos.	
pgsql.db.discovery	La lista de bases de datos de PostgreSQL.	
pgsql.db.size	El tamaño de la base de datos en bytes.	
pgsql.dbstat	Recopila las estadísticas por base de datos.	
pgsql.dbstat.sum	Los datos resumidos de todas las bases de datos de un clúster.	
pgsql.locks	La información sobre bloqueos concedidos por base de datos.	
pgsql.oldest.xid	La antigüedad del XID más antiguo.	
pgsql.ping	Comprueba si una conexión está activa o no.	
pgsql.queries	Métricas de consultas por tiempo de ejecución.	
pgsql.replication.count	El número de servidores en espera.	
pgsql.replication.process	El retraso de flush, el retraso de escritura y el retraso de replay para cada proceso emisor.	
pgsql.replication.process.discovery	El descubrimiento del nombre del proceso de replicación.	
pgsql.replication.recovery_role	El estado de recuperación.	
pgsql.replication.status	El estado de la replicación.	
pgsql.replication_lag.b	El retraso de replicación en bytes.	
pgsql.replication_lag.sec	El retraso de replicación en segundos.	
pgsql.uptime	El tiempo de actividad de PostgreSQL en milisegundos.	
pgsql.version	Devuelve la versión de PostgreSQL.	
pgsql.wal.stat	Las estadísticas de WAL.	
redis.config	Obtiene los parámetros de configuración de una instancia de Redis que coinciden con el patrón.	Redis
redis.info	Obtiene la salida del comando INFO.	
redis.ping	Comprueba si una conexión está activa o no.	
redis.slowlog.count	El número de entradas del registro lento desde que se inició Redis.	
smart.attribute.discovery	Devuelve una lista de atributos de dispositivos S.M.A.R.T.	S.M.A.R.T.
smart.disk.discovery	Devuelve una lista de dispositivos S.M.A.R.T.	
smart.disk.get	Devuelve todas las propiedades disponibles de los dispositivos S.M.A.R.T.	
systemd.unit.get	Devuelve todas las propiedades de una unidad systemd.	Systemd
systemd.unit.info	Información de la unidad systemd.	
systemd.unit.discovery	La lista de unidades systemd y sus detalles.	
web.certificate.get	Valida los certificados y devuelve los detalles del certificado.	Web certificates

Detalles de la clave del elemento

Los parámetros sin corchetes angulares son obligatorios. Los parámetros marcados con corchetes angulares **< >** son opcionales.

ceph.df.details[connString,<user>,<apikey>]

 El uso de datos del clúster y la distribución entre los pools.
 Valor devuelto: *objeto JSON*.

Parámetros:

- **connString** - el URI o nombre de sesión;
- **user, apikey** - el nombre de usuario y la clave API para conectarse a Ceph.

ceph.osd.stats[connString,<user>,<apikey>]

 Estadísticas agregadas y por OSD.
 Valor devuelto: *objeto JSON*.

Parámetros:

- **connString** - el URI o nombre de sesión;
- **user, apikey** - el nombre de usuario y la clave API para conectarse a Ceph.

ceph.osd.discovery[connString,<user>,<apikey>]

 La lista de OSDs descubiertos. Utilizado para descubrimiento de bajo nivel.
 Valor de retorno: *objeto JSON*.

Parámetros:

- **connString** - el URI o nombre de sesión;
- **user, apikey** - el nombre de usuario y la clave API para conectarse a Ceph.

ceph.osd.dump[connString,<user>,<apikey>]

 Los umbrales de uso y los estados de los OSD.
 Valor devuelto: *objeto JSON*.

Parámetros:

- **connString** - el URI o nombre de sesión;
- **user, apikey** - el nombre de usuario y la clave API para conectarse a Ceph.

ceph.ping[connString,<user>,<apikey>]

 Prueba si se puede establecer una conexión con Ceph.
 Valor de retorno: *0* - la conexión está rota (si se presenta algún error, incluidos problemas de AUTH y configuración); *1* - la conexión es exitosa.

Parámetros:

- **connString** - el URI o nombre de la sesión;
- **user, apikey** - el nombre de usuario y la clave API para conectarse a Ceph.

ceph.pool.discovery[connString,<user>,<apikey>]

 La lista de pools descubiertos. Se utiliza para descubrimiento de bajo nivel.
 Valor de retorno: *objeto JSON*.

Parámetros:

- **connString** - el URI o nombre de sesión;
- **user, apikey** - el nombre de usuario y la clave API para conectarse a Ceph.

ceph.status[connString,<user>,<apikey>]

 El estado general del clúster.
 Valor de retorno: *objeto JSON*.

Parámetros:

- **connString** - el URI o nombre de sesión;
- **user, apikey** - el nombre de usuario y la clave API para conectarse a Ceph.

docker.container_info[<ID>,<info>]

 Información de bajo nivel sobre un contenedor.
 Valor de retorno: La salida de la llamada a la API ContainerInspect serializada como JSON.

Parámetros:

- **ID**: el ID o nombre del contenedor;
- **info**: la cantidad de información devuelta. Valores soportados: *short* (por defecto) o *full*.

El usuario del Agente 2 (’zabbix’) debe ser añadido al grupo ’docker’ para tener privilegios suficientes. De lo contrario, la comprobación fallará.

docker.container_stats[<ID>]

 Las estadísticas de uso de recursos del contenedor.
 Valor de retorno: La salida de la llamada a la API ContainerStats y el porcentaje de uso de CPU serializados como JSON.

Parámetro:

- **ID**: el ID o nombre del contenedor.

El usuario del Agente 2 (’zabbix’) debe ser añadido al grupo ’docker’ para tener privilegios suficientes. De lo contrario, la comprobación fallará.

docker.containers

 La lista de contenedores.
 Valor de retorno: La salida de la llamada a la API ContainerList serializada como JSON.

El usuario del Agente 2 (’zabbix’) debe ser añadido al grupo ’docker’ para tener privilegios suficientes. De lo contrario, la comprobación fallará.

docker.containers.discovery[<opciones>]

 Devuelve la lista de contenedores. Se utiliza para el descubrimiento de bajo nivel.
 Valor de retorno: *objeto JSON*.

Parámetro:

- **opciones**: especifica si se deben descubrir todos los contenedores o solo los que están en ejecución. Valores admitidos: *true* - devuelve todos los contenedores; *false* - devuelve solo los contenedores en ejecución (por defecto).

El usuario del Agente 2 (’zabbix’) debe ser añadido al grupo ’docker’ para tener privilegios suficientes. De lo contrario, la comprobación fallará.

docker.data.usage

 Información sobre el uso actual de datos.
 Valor de retorno: La salida de la llamada a la API SystemDataUsage serializada como JSON.

El usuario del Agente 2 (’zabbix’) debe ser añadido al grupo ’docker’ para disponer de privilegios suficientes. De lo contrario, la comprobación fallará.

docker.images

 Devuelve la lista de imágenes.
 Valor de retorno: La salida de la llamada a la API ImageList serializada como JSON.

El usuario del Agente 2 (’zabbix’) debe ser añadido al grupo ’docker’ para tener privilegios suficientes. De lo contrario, la comprobación fallará.

docker.images.discovery

 Devuelve la lista de imágenes. Se utiliza para el descubrimiento de bajo nivel.
 Valor de retorno: *objeto JSON*.

El usuario del Agente 2 (’zabbix’) debe ser añadido al grupo ’docker’ para tener privilegios suficientes. De lo contrario, la comprobación fallará.

docker.info

 La información del sistema.
 Valor de retorno: La salida de la llamada a la API SystemInfo serializada como JSON.

El usuario del Agente 2 (’zabbix’) debe ser añadido al grupo ’docker’ para tener privilegios suficientes. De lo contrario, la comprobación fallará.

docker.ping

 Prueba si el demonio de Docker está activo o no.
 Valor de retorno: *1* - la conexión está activa; *0* - la conexión está rota.

El usuario del Agente 2 (’zabbix’) debe ser añadido al grupo ’docker’ para tener privilegios suficientes. De lo contrario, la comprobación fallará.

ember.get[<uri>,<ruta>]

 Devuelve el resultado del dispositivo requerido.
 Valor de retorno: *objeto JSON*.

Parámetros:

- **uri** - URI del dispositivo Ember+. Por defecto: 127.0.0.1:9998;
- **ruta** - Ruta OID al dispositivo. Vacío por defecto, devuelve los datos de la colección raíz.

memcached.ping[connString,<user>,<password>]

 Prueba si una conexión está activa o no.
 Valor de retorno: *1* - la conexión está activa; *0* - la conexión está rota (si se presenta algún error, incluyendo problemas de AUTH y configuración).

Parámetros:

- **connString** - el URI o nombre de sesión;
- **user, password** - las credenciales de acceso a Memcached.

memcached.stats[connString,<usuario>,<contraseña>,<tipo>]

 Obtiene la salida del comando STATS.
 Valor de retorno: *JSON* - la salida se serializa como JSON.

Parámetros:

- **connString** - el URI o nombre de sesión;
- **usuario, contraseña** - las credenciales de acceso a Memcached;
- **tipo** - tipo de estadística a devolver: *items*, *sizes*, *slabs* o *settings* (vacío por defecto, devuelve estadísticas generales).

mongodb.collection.stats[connString,<usuario>,<contraseña>,<base_de_datos>,colección]

 Devuelve una variedad de estadísticas de almacenamiento para una colección dada.
 Valor de retorno: *objeto JSON*.

Parámetros:

- **connString** - el URI o nombre de sesión;
- **usuario, contraseña** - las credenciales de inicio de sesión de MongoDB;
- **base_de_datos** - el nombre de la base de datos (por defecto: admin);
- **colección** - el nombre de la colección.

mongodb.cfg.discovery[connString,<user>,<password>]

 Devuelve una lista de servidores de configuración descubiertos.
 Valor devuelto: *objeto JSON*.

Parámetros:

- **connString** - el URI o nombre de la sesión;
- **user, password** - las credenciales de inicio de sesión de MongoDB.

mongodb.collections.discovery[connString,<usuario>,<contraseña>]

 Devuelve una lista de colecciones descubiertas. Se utiliza para descubrimiento de bajo nivel.
 Valor de retorno: *objeto JSON*.

Parámetros:

- **connString** - el URI o nombre de sesión;
- **usuario, contraseña** - las credenciales de inicio de sesión de MongoDB.

mongodb.collections.usage[connString,<user>,<password>]

 Devuelve las estadísticas de uso para las colecciones.
 Valor de retorno: *objeto JSON*.

Parámetros:

- **connString**: el URI o nombre de sesión;
- **user, password**: las credenciales de inicio de sesión de MongoDB.

mongodb.connpool.stats[connString,<user>,<password>]

 Devuelve información sobre las conexiones salientes abiertas desde la instancia de base de datos actual a otros miembros del clúster sharded o del conjunto de réplicas.
 Valor devuelto: *objeto JSON*.

Parámetros:

- **connString** - el URI o nombre de la sesión;
- **user, password** - las credenciales de acceso a MongoDB.

mongodb.db.stats[connString,<user>,<password>,<database>]

 Devuelve las estadísticas que reflejan el estado del sistema de una base de datos dada.
 Valor de retorno: *objeto JSON*.

Parámetros:

- **connString** - el URI o nombre de sesión;
- **user, password** - las credenciales de acceso a MongoDB;
- **database** - el nombre de la base de datos (por defecto: admin).

mongodb.db.discovery[connString,<usuario>,<contraseña>]

 Devuelve una lista de bases de datos descubiertas. Se utiliza para descubrimiento de bajo nivel.
 Valor de retorno: *objeto JSON*.

Parámetros:

- **connString**: la URI o el nombre de la sesión;
- **usuario, contraseña**: las credenciales de inicio de sesión de MongoDB.

mongodb.jumbo_chunks.count[connString,<user>,<password>]

 Devuelve el número de jumbo chunks.
 Valor de retorno: *objeto JSON*.

Parámetros:

- **connString** - el URI o nombre de la sesión;
- **user, password** - las credenciales de acceso a MongoDB.

mongodb.oplog.stats[connString,<user>,<password>]

 Devuelve el estado del conjunto de réplicas, utilizando datos obtenidos del oplog.
 Valor de retorno: *objeto JSON*.

Parámetros:

- **connString** - el URI o nombre de la sesión;
- **user, password** - las credenciales de acceso a MongoDB.

mongodb.ping[connString,<user>,<password>]

 Prueba si una conexión está activa o no.
 Valor de retorno: *1* - la conexión está activa; *0* - la conexión está rota (si se presenta algún error, incluyendo problemas de autenticación y configuración).

Parámetros:

- **connString** - el URI o nombre de sesión;
- **user, password** - las credenciales de inicio de sesión de MongoDB.

mongodb.rs.config[connString,<user>,<password>]

 Devuelve la configuración actual del conjunto de réplicas.
 Valor de retorno: *objeto JSON*.

Parámetros:

- **connString** - el URI o nombre de la sesión;
- **user, password** - las credenciales de acceso a MongoDB.

mongodb.rs.status[connString,<user>,<password>]

 Devuelve el estado del conjunto de réplicas desde el punto de vista del miembro donde se ejecuta el método.
 Valor de retorno: *objeto JSON*.

Parámetros:

- **connString** - el URI o nombre de la sesión;
- **user, password** - las credenciales de acceso a MongoDB.

mongodb.server.status[connString,<user>,<password>]

 Devuelve el estado de la base de datos.
 Valor de retorno: *objeto JSON*.

Parámetros:

- **connString** - el URI o nombre de sesión;
- **user, password** - las credenciales de acceso a MongoDB.

mongodb.sh.discovery[connString,<user>,<password>]

 Devuelve la lista de fragmentos descubiertos presentes en el clúster.
 Valor de retorno: *objeto JSON*.

Parámetros:

- **connString** - el URI o nombre de sesión;
- **user, password** - las credenciales de inicio de sesión de MongoDB.

mongodb.version[connString,<user>,<password>]

 Devuelve la versión del servidor de base de datos.
 Valor de retorno: *Cadena*.

Parámetros:

- **connString**: el URI o nombre de sesión;
- **user, password**: las credenciales de acceso a MongoDB.

mqtt.get[<broker url>,topic,<user>,<password>]

 Se suscribe a un tema específico o a temas específicos (con comodines) del broker proporcionado y espera las publicaciones.
 Valor devuelto: Depende del contenido del tema. Si se usan comodines, devuelve el contenido del tema como JSON.

Parámetros:

- **broker url** - la URL del broker MQTT en el formato `protocol://host:port` sin parámetros de consulta (los protocolos admitidos son: `tcp`, `ssl`, `ws`). Si no se especifica ningún valor, el agent utilizará `tcp://localhost:1883`. Si se omite el protocolo o el puerto, se utilizará el protocolo predeterminado (`tcp`) o el puerto predeterminado (1883);
- **topic** - el tema MQTT (obligatorio). Se admiten comodines (+,#);
- **user, password** - las credenciales de autenticación (si son necesarias).

Comentarios:

- El item debe configurarse como una active check (tipo de item 'Zabbix agent (active)');
- Los certificados de cifrado TLS se pueden usar guardándolos en una ubicación predeterminada (por ejemplo, el directorio `/etc/ssl/certs/` para Ubuntu). Para TLS, use el esquema `tls://`.

mssql.availability.group.get[URI,<usuario>,<contraseña>]

 Devuelve los grupos de disponibilidad.
 Valor de retorno: *objeto JSON*.

Parámetros:

- **URI** - URI del servidor MSSQL (el único esquema soportado es `sqlserver://`). Las credenciales incrustadas serán ignoradas. Es posible especificar un nombre de instancia como parte de la URI, es decir: `sqlserver://localhost/InstanceName` (sin puerto). Si se especifica el puerto, el nombre de la instancia se ignora;
- **usuario, contraseña** - nombre de usuario, contraseña para enviar al servidor MSSQL protegido.

Para más información, consulte el archivo readme del plugin MSSQL.

mssql.custom.query[URI,<usuario>,<contraseña>,nombreConsulta,<args...>]

 Devuelve el resultado de una consulta personalizada.
 Valor de retorno: *objeto JSON*.

Parámetros:

- **URI** - URI del servidor MSSQL (el único esquema soportado es `sqlserver://`). Las credenciales incrustadas serán ignoradas. Es posible especificar un nombre de instancia como parte de la URI, es decir: `sqlserver://localhost/InstanceName` (sin puerto). Si se especifica el puerto, el nombre de la instancia se ignora;
- **usuario, contraseña** - nombre de usuario, contraseña para enviar al servidor MSSQL protegido;
- **nombreConsulta** - nombre de una consulta personalizada configurada en `Plugins.MSSQL.CustomQueriesDir` sin la extensión .sql;
- **args** - uno o varios argumentos separados por comas para pasar a la consulta.

Para más información, consulte el readme del plugin MSSQL.

mssql.db.get

 Devuelve todas las bases de datos MSSQL disponibles.
 Valor de retorno: *objeto JSON*.

Para más información, consulte el archivo readme del plugin MSSQL.

mssql.job.status.get

 Devuelve el estado de los trabajos.
 Valor de retorno: *objeto JSON*.

Para más información, consulte el archivo readme del plugin MSSQL.

mssql.last.backup.get

 Devuelve la hora de la última copia de seguridad para todas las bases de datos.
 Valor de retorno: *objeto JSON*.

Para más información, consulte el archivo readme del plugin MSSQL.

mssql.local.db.get

 Devuelve las bases de datos que participan en un grupo de disponibilidad Always On y réplica (primaria o secundaria) y que se encuentran en el servidor al que se estableció la conexión.
 Valor de retorno: *objeto JSON*.

Para más información, consulte el readme del plugin MSSQL.

mssql.mirroring.get

 Devuelve información de mirroring.
 Valor de retorno: *objeto JSON*.

Para más información, consulte el archivo readme del plugin MSSQL.

mssql.nonlocal.db.get

 Devuelve las bases de datos que participan en un grupo de disponibilidad Always On y la réplica (primaria o secundaria) ubicada en otros servidores (la base de datos no es local para la instancia de SQL Server a la que se estableció la conexión).
 Valor de retorno: *objeto JSON*.

Para obtener más información, consulte el archivo readme del plugin MSSQL.

mssql.perfcounter.get

 Devuelve los contadores de rendimiento.
 Valor de retorno: *objeto JSON*.

Para más información, consulte el archivo readme del plugin MSSQL.

mssql.ping

 Hacer ping a la base de datos. Comprueba si la conexión está configurada correctamente.
 Valor de retorno: *1* - activo, *0* - inactivo.

Para más información, consulte el archivo readme del plugin MSSQL.

mssql.quorum.get

 Devuelve la información de quórum.
 Valor de retorno: *objeto JSON*.

Para más información, consulte el archivo readme del plugin MSSQL.

mssql.quorum.member.get

 Devuelve los miembros del quórum.
 Valor de retorno: *objeto JSON*.

Para más información, consulte el archivo readme del plugin MSSQL.

mssql.replica.get

 Devuelve las réplicas.
 Valor de retorno: *objeto JSON*.

Para más información, consulte el archivo readme del plugin MSSQL.

mssql.version

 Devuelve la versión de MSSQL.
 Valor de retorno: *Cadena*.

Para más información, consulte el archivo readme del plugin MSSQL.

mysql.custom.query[connString,<user>,<password>,queryName,<args...>]

 Devuelve el resultado de una consulta personalizada.
 Valor de retorno: *objeto JSON*.

Parámetros:

- **connString** - URI o nombre de sesión;
- **user, password** - credenciales de inicio de sesión de MySQL;
- **queryName** - nombre de una consulta personalizada, debe coincidir con el nombre del archivo SQL sin extensión;
- **args** - uno o varios argumentos separados por comas para pasar a la consulta.

Para más información, consulte el archivo readme del plugin de MySQL, sección *Consultas personalizadas*.

mysql.db.discovery[connString,<usuario>,<contraseña>]

 Devuelve la lista de bases de datos MySQL. Se utiliza para el descubrimiento de bajo nivel.
 Valor de retorno: El resultado de la consulta SQL "show databases" en formato JSON de LLD.

Parámetros:

- **connString** - el URI o nombre de sesión;
- **usuario, contraseña** - las credenciales de acceso a MySQL.

mysql.db.size[connString,<usuario>,<contraseña>,<nombre de base de datos>]

 El tamaño de la base de datos en bytes.
 Valor devuelto: Resultado de la consulta SQL "select coalesce(sum(data_length + index_length),0) as size from information_schema.tables where table_schema=?" para una base de datos específica en bytes.

Parámetros:

- **connString** - el URI o nombre de sesión;
- **usuario, contraseña** - las credenciales de acceso a MySQL;
- **nombre de base de datos** - el nombre de la base de datos.

mysql.get_status_variables[connString,<usuario>,<contraseña>]

 Valores de las variables de estado globales.
 Valor de retorno: Resultado de la consulta SQL "show global status" en formato JSON.

Parámetros:

- **connString** - el URI o nombre de sesión;
- **usuario, contraseña** - las credenciales de acceso a MySQL.

mysql.ping[connString,<user>,<password>]

 Prueba si una conexión está activa o no.
 Valor de retorno: *1* - la conexión está activa; *0* - la conexión está rota (si se presenta algún error, incluyendo problemas de AUTENTICACIÓN y configuración).

Parámetros:

- **connString** - el URI o nombre de sesión;
- **user, password** - las credenciales de acceso a MySQL.

mysql.replication.discovery[connString,<user>,<password>]

 Devuelve la lista de replicaciones de MySQL. Se utiliza para el descubrimiento de bajo nivel.
 Valor de retorno: el resultado de la consulta SQL `SHOW SLAVE STATUS` o `SHOW REPLICA STATUS` en formato JSON de LLD.

Parámetros:

- **connString** - el URI o nombre de sesión;
- **user, password** - las credenciales de acceso a MySQL.

mysql.replication.get_slave_status[connString,<user>,<password>,<master host>]

 El estado de la replicación.
 Valor de retorno: resultado de la consulta SQL `SHOW SLAVE STATUS` o `SHOW REPLICA STATUS` en formato JSON.

Parámetros:

- **connString** - el URI o nombre de sesión;
- **user, password** - las credenciales de acceso a MySQL;
- **master host** - el nombre del host maestro de replicación. Si no se encuentra ninguno, se devuelve un error. Si este parámetro no se especifica, se devuelven todos los hosts.

mysql.version[connString,<user>,<password>]

 La versión de MySQL.
 Valor de retorno: *String* (con la versión de la instancia de MySQL).

Parámetros:

- **connString** - el URI o nombre de sesión;
- **user, password** - las credenciales de acceso a MySQL.

net.dns.get[<ip>,name,<type>,<timeout>,<count>,<protocol>,"<flags>"]

Realiza una consulta DNS y devuelve información detallada del registro DNS.
 Este item es una versión ampliada del item de Zabbix agent `net.dns.record`, con compatibilidad para más tipos de registros y flags personalizables.
 Valores devueltos: *JSON object*

Parámetros:

- **ip** - la dirección IP del servidor DNS (déjelo vacío para usar el servidor DNS predeterminado);
- **name** - el nombre DNS que se va a consultar;
- **type** - el tipo de registro que se va a consultar (el valor predeterminado es *SOA*);
- **timeout** - el tiempo de espera de la solicitud en segundos (el valor predeterminado es 1 segundo);
- **count** - el número de intentos para la solicitud (el valor predeterminado es 2);
- **protocol** - el protocolo usado para realizar consultas DNS: *udp* (predeterminado) o *tcp*;
- **flags** - uno o más argumentos separados por comas que se pasarán a una consulta.

Comentarios:

- Los valores posibles para `type` son: *A*, *NS*, *MD*, *MF*, *CNAME*, *SOA*, *MB*, *MG*, *MR*, *NULL*, *PTR*, *HINFO*, *MINFO*, *MX*, *TXT*, *RP*, *AFSDB*, *X25*, *ISDN*, *RT*, *NSAPPTR*, *SIG*, *KEY*, *PX*, *GPOS*, *AAAA*, *LOC*, *NXT*, *EID*, *NIMLOC*, *SRV*, *ATMA*, *NAPTR*, *KX*, *CERT*, *DNAME*, *OPT*, *APL*, *DS*, *SSHFP*, *IPSECKEY*, *RRSIG*, *NSEC*, *DNSKEY*, *DHCID*, *NSEC3*, *NSEC3PARAM*, *TLSA*, *SMIMEA*, *HIP*, *NINFO*, *RKEY*, *TALINK*, *CDS*, *CDNSKEY*, *OPENPGPKEY*, *CSYNC*, *ZONEMD*, *SVCB*, *HTTPS*, *SPF*, *UINFO*, *UID*, *GID*, *UNSPEC*, *NID*, *L32*, *L64*, *LP*, *EUI48*, *EUI64*, *URI*, *CAA*, *AVC*, *AMTRELAY*. Tenga en cuenta que los valores deben estar solo en mayúsculas; no se admiten valores en minúsculas ni en formato mixto.
- Para búsquedas DNS inversas (cuando `type` se establece en *PTR*), puede proporcionar el nombre DNS tanto en formato invertido como no invertido (consulte los ejemplos a continuación). Tenga en cuenta que, cuando se solicita un registro PTR, el nombre DNS en realidad es una dirección IP.
- Los valores posibles para `flags` son:
  - *cdflag* o *nocdflag* (predeterminado);
  - *rdflag* (predeterminado) o *nordflag*;
  - *dnssec* o *nodnssec* (predeterminado);
  - *nsid* o *nonsid* (predeterminado);
  - *edns0* (predeterminado) o *noedns0*;
  - *aaflag* o *noaaflag* (predeterminado);
  - *adflag* o *noadflag* (predeterminado). Los `flags` *dnssec* y *nsid* no se pueden usar junto con *noedns0*, ya que ambos requieren *edns0*. Tenga en cuenta que los valores deben estar solo en minúsculas; no se admiten valores en mayúsculas ni en formato mixto.
- Los nombres de dominio internacionalizados no son compatibles; utilice en su lugar nombres codificados en IDNA.
- La salida es un objeto que contiene información del registro DNS según los parámetros proporcionados (consulte más detalles).

Ejemplos:

```
net.dns.get[192.0.2.1,example.com,DNSKEY,3,3,tcp,"cdflag,rdflag,nsid"]

net.dns.get[,198.51.100.1,PTR,,,,"cdflag,rdflag,nsid"]
net.dns.get[,1.100.51.198.in-addr.arpa,PTR,,,,"cdflag,rdflag,nsid"]

net.dns.get[,2a00:1450:400f:800::200e,PTR,,,,"cdflag,rdflag,nsid"]
net.dns.get[,e.0.0.2.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.8.0.f.0.0.4.0.5.4.1.0.0.a.2.ip6.arpa,PTR,,,,"cdflag,rdfla
```

nvml.device.count

 El número de dispositivos GPU.
 Valor de retorno: *Entero*.

Para más información, consulte el archivo readme del plugin NVIDIA GPU.

nvml.device.decoder.utilization[<deviceUUID>]

 Utilización del decodificador del dispositivo GPU como porcentaje.
 Valor de retorno: *Entero*.

Parámetro:

- **deviceUUID** - UUID del dispositivo GPU.

Para obtener más información, consulte el archivo readme del plugin NVIDIA GPU.

nvml.device.ecc.mode[<deviceUUID>]

 Información del modo ECC del dispositivo GPU (actual, pendiente).
 Valor de retorno: *JSON*.

Parámetro:

- **deviceUUID** - UUID del dispositivo GPU.

Para más información, consulte el archivo readme del plugin NVIDIA GPU.

nvml.device.encoder.stats.get[<deviceUUID>]

 Estadísticas del codificador del dispositivo GPU.
 Valor de retorno: *JSON*.

Parámetro:

- **deviceUUID** - UUID del dispositivo GPU.

Para más información, consulte el archivo readme del plugin NVIDIA GPU.

nvml.device.encoder.utilization[<deviceUUID>]

 Utilización del codificador del dispositivo GPU como porcentaje.
 Valor de retorno: *Entero*.

Parámetro:

- **deviceUUID** - UUID del dispositivo GPU.

Para obtener más información, consulte el archivo readme del plugin NVIDIA GPU.

nvml.device.energy.consumption[<deviceUUID>]

 Consumo total de energía del dispositivo GPU en milijulios (mJ) desde la última recarga del controlador.
 Valor de retorno: *Entero*.

Parámetro:

- **deviceUUID** - UUID del dispositivo GPU.

Para más información, consulte el archivo readme del plugin NVIDIA GPU.

nvml.device.errors.memory[<deviceUUID>]

 Información de errores de memoria ECC del dispositivo GPU (corregidos, no corregidos).
 Valor de retorno: *JSON*.

Parámetro:

- **deviceUUID** - UUID del dispositivo GPU.

Para más información, consulte el archivo readme del plugin NVIDIA GPU.

nvml.device.errors.register[<deviceUUID>]

 Información de errores de registro ECC del dispositivo GPU (corregidos, no corregidos).
 Valor de retorno: *JSON*.

Parámetro:

- **deviceUUID** - UUID del dispositivo GPU.

Para más información, consulte el archivo readme del plugin NVIDIA GPU.

nvml.device.fan.speed.avg[<deviceUUID>]

 Velocidad media del ventilador del dispositivo GPU como porcentaje de la velocidad máxima.
 Valor devuelto: *Entero*.

Parámetro:

- **deviceUUID** - UUID del dispositivo GPU.

Para más información, consulte el archivo readme del plugin NVIDIA GPU.

nvml.device.get

 GPUs descubiertas con UUID y nombre.
 Valor de retorno: *JSON*.

Para más información, consulte el archivo readme del plugin NVIDIA GPU.

nvml.device.graphics.frequency[<deviceUUID>]

 Velocidad del reloj gráfico del dispositivo GPU en MHz.
 Valor de retorno: *Entero*.

Parámetro:

- **deviceUUID** - UUID del dispositivo GPU.

Para más información, consulte el archivo readme del plugin NVIDIA GPU.

nvml.device.memory.fb.get[<deviceUUID>]

 Estadísticas de memoria framebuffer del dispositivo GPU (total, reservada, libre, usada).
 Valor de retorno: *JSON*.

Parámetro:

- **deviceUUID** - UUID del dispositivo GPU.

Para más información, consulte el archivo readme del plugin NVIDIA GPU.

nvml.device.memory.bar1.get[<deviceUUID>]

 Estadísticas de memoria BAR1 del dispositivo GPU (total, libre, usada).
 Valor de retorno: *JSON*.

Parámetro:

- **deviceUUID** - UUID del dispositivo GPU.

Para más información, consulte el archivo readme del plugin NVIDIA GPU.

nvml.device.memory.frequency[<deviceUUID>]

 Velocidad del reloj de la memoria del dispositivo GPU en MHz.
 Valor de retorno: *Entero*.

Parámetro:

- **deviceUUID** - UUID del dispositivo GPU.

Para más información, consulte el archivo readme del plugin NVIDIA GPU.

nvml.device.pci.utilization[<deviceUUID>]

 Utilización PCI del dispositivo GPU (rendimiento de transmisión/recepción en KBps).
 Valor de retorno: *JSON*.

Parámetro:

- **deviceUUID** - UUID del dispositivo GPU.

Para más información, consulte el archivo readme del plugin NVIDIA GPU.

nvml.device.performance.state[<deviceUUID>]

 Estado de rendimiento del dispositivo GPU.
 Valor de retorno: *Entero*.

Parámetro:

- **deviceUUID** - UUID del dispositivo GPU.

Para más información, consulte el archivo readme del plugin NVIDIA GPU.

nvml.device.power.limit[<deviceUUID>]

 Límite de potencia del dispositivo GPU en milivatios.
 Valor de retorno: *Entero*.

Parámetro:

- **deviceUUID** - UUID del dispositivo GPU.

Para más información, consulte el archivo readme del plugin NVIDIA GPU.

nvml.device.power.usage[<deviceUUID>]

 Consumo actual de energía del dispositivo GPU en milivatios.
 Valor de retorno: *Entero*.

Parámetro:

- **deviceUUID** - UUID del dispositivo GPU.

Para más información, consulte el archivo readme del plugin NVIDIA GPU.

nvml.device.serial[<deviceUUID>]

 Número de serie del dispositivo GPU.
 Valor de retorno: *Cadena*.

Parámetro:

- **deviceUUID** - UUID del dispositivo GPU.

Para más información, consulte el archivo readme del plugin NVIDIA GPU.

nvml.device.sm.frequency[<deviceUUID>]

 Velocidad de reloj del multiprocesador de flujo del dispositivo GPU en MHz.
 Valor de retorno: *Entero*.

Parámetro:

- **deviceUUID** - UUID del dispositivo GPU.

Para más información, consulte el archivo readme del plugin NVIDIA GPU.

nvml.device.temperature[<deviceUUID>]

 Temperatura del dispositivo GPU en grados Celsius.
 Valor de retorno: *Entero*.

Parámetro:

- **deviceUUID** - UUID del dispositivo GPU.

Para más información, consulte el archivo readme del plugin NVIDIA GPU.

nvml.device.utilization[<deviceUUID>]

 Estadísticas de utilización del dispositivo GPU (utilización de GPU/memoria como porcentaje).
 Valor de retorno: *JSON*.

Parámetro:

- **deviceUUID** - UUID del dispositivo GPU.

Para más información, consulte el archivo readme del plugin NVIDIA GPU.

nvml.device.video.frequency[<deviceUUID>]

 Frecuencia de reloj de vídeo del dispositivo GPU en MHz.
 Valor de retorno: *Entero*.

Parámetro:

- **deviceUUID** - UUID del dispositivo GPU.

Para más información, consulte el archivo readme del plugin NVIDIA GPU.

nvml.system.driver.version

 La versión del controlador NVIDIA.
 Valor de retorno: *Cadena*.

Para obtener más información, consulte el archivo readme del plugin NVIDIA GPU.

nvml.version

 La versión de la biblioteca NVML.
 Valor de retorno: *Cadena*.

Para más información, consulte el archivo readme del plugin NVIDIA GPU.

oracle.diskgroups.stats[connString,<user>,<password>,<service>,<diskgroup>]

 Devuelve las estadísticas de los grupos de discos de Automatic Storage Management (ASM).
 Valor devuelto: *objeto JSON*.

Parámetros:

- **connString** - puede ser:
  - el URI - si no se encuentra ninguna sesión con el nombre proporcionado y `Plugins.Oracle.ResolveTNS` está establecido en false. Si contiene un esquema (por ejemplo, "tcp://"), un puerto (por ejemplo, 1521) o ambos, la opción ResolveTNS no se tendrá en cuenta y, en cualquier caso, se tratará como un URI;
  - nombre de sesión - si dicho nombre se encuentra en el archivo de configuración del plugin;
  - clave TNS - si la opción ResolveTNS está establecida en true y no se aplica ninguna de las condiciones anteriores;
  - valor TNS - si comienza con el paréntesis de apertura "(" (se ignoran los espacios iniciales). En este caso, la opción ResolveTNS no se tiene en cuenta.
- **user** - el nombre de usuario de Oracle; admite añadir una de las opciones de inicio de sesión `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` o `as sysrac` en el formato `user as sysdba` (una opción de inicio de sesión no distingue entre mayúsculas y minúsculas y no debe contener un espacio final).
- **password** - la contraseña de Oracle;
- **service** - el nombre del servicio de Oracle;
- **diskgroup** - el nombre del grupo de discos ASM que se va a consultar.

oracle.diskgroups.discovery[connString,<user>,<password>,<service>]

 Devuelve la lista de grupos de discos ASM. Se utiliza para el descubrimiento de bajo nivel.
 Valor devuelto: *objeto JSON*.

Parámetros:

- **connString** - puede ser:
  - el URI: si no se encuentra ninguna sesión con el nombre indicado y `Plugins.Oracle.ResolveTNS` está establecido en false. Si contiene un esquema (por ejemplo, "tcp://"), un puerto (por ejemplo, 1521) o ambos, la opción ResolveTNS no se tendrá en cuenta y, de todos modos, se tratará como URI;
  - nombre de sesión: si dicho nombre se encuentra en el archivo de configuración del plugin;
  - clave TNS: si la opción ResolveTNS está establecida en true y no se cumple ninguna de las condiciones anteriores;
  - valor TNS: si comienza con el paréntesis de apertura "(" (se ignoran los espacios iniciales). En este caso, la opción ResolveTNS no se tiene en cuenta.
- **user** - el nombre de usuario de Oracle; admite añadir una de las opciones de inicio de sesión `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` o `as sysrac` en el formato `user as sysdba` (una opción de inicio de sesión no distingue entre mayúsculas y minúsculas y no debe contener un espacio final).
- **password** - la contraseña de Oracle;
- **service** - el nombre del servicio de Oracle.

oracle.archive.info[connString,<user>,<password>,<service>,<destination>]

 Las estadísticas de los registros de archivo.
 Valor devuelto: *objeto JSON*.

Parámetros:

- **connString** - puede ser:
  - el URI: si no se encuentra ninguna sesión con el nombre proporcionado y `Plugins.Oracle.ResolveTNS` está establecido en false. Si contiene un esquema (por ejemplo, "tcp://"), un puerto (por ejemplo, 1521) o ambos, la opción ResolveTNS no se tendrá en cuenta y, en cualquier caso, se tratará como URI;
  - nombre de sesión: si dicho nombre se encuentra en el archivo de configuración del plugin;
  - clave TNS: si la opción ResolveTNS está establecida en true y no se cumple ninguna de las condiciones anteriores;
  - valor TNS: si comienza con un paréntesis de apertura "(" (ignorando los espacios iniciales). En este caso, la opción ResolveTNS no se tiene en cuenta.
- **user** - el nombre de usuario de Oracle; admite añadir una de las opciones de inicio de sesión `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` o `as sysrac` en el formato `user as sysdba` (una opción de inicio de sesión no distingue entre mayúsculas y minúsculas y no debe contener un espacio final).
- **password** - la contraseña de Oracle;
- **service** - el nombre del servicio de Oracle;
- **destination** - el nombre del destino que se va a consultar.

oracle.cdb.info[connString,<user>,<password>,<service>,<database>]

 La información de las bases de datos contenedor (CDB).
 Valor de retorno: *objeto JSON*.

Parámetros:

- **connString** - puede ser:

    - el URI: si no se encuentra ninguna sesión con el nombre proporcionado y `Plugins.Oracle.ResolveTNS` está establecido en false. Si contiene un esquema (por ejemplo, "tcp://"), un puerto (por ejemplo, 1521) o ambos, la opción ResolveTNS no se tendrá en cuenta y, de todos modos, se tratará como un URI;
    - nombre de sesión: si dicho nombre se encuentra en el archivo de configuración del plugin;
    - clave TNS: si la opción ResolveTNS está establecida en true y no se aplica ninguna de las condiciones anteriores;
    - valor TNS: si comienza con el paréntesis de apertura "(" (se ignoran los espacios iniciales). En este caso, la opción ResolveTNS no se tiene en cuenta.
- **user** - el nombre de usuario de Oracle; admite añadir una de las opciones de inicio de sesión `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` o `as sysrac` en el formato `user as sysdba` (una opción de inicio de sesión no distingue entre mayúsculas y minúsculas y no debe contener un espacio final).
- **password** - la contraseña de Oracle;
- **service** - el nombre del servicio de Oracle;
- **database** - el nombre de la base de datos que se va a consultar.

oracle.custom.query[connString,<user>,<password>,<service>,queryName,<args...>]

 El resultado de una consulta personalizada.
 Valor de retorno: *objeto JSON*.

Parámetros:

- **connString** - puede ser:
    - el URI - si no se encuentra ninguna sesión con el nombre proporcionado y `Plugins.Oracle.ResolveTNS` está establecido en false. Si contiene un esquema (por ejemplo, "tcp://"), un puerto (por ejemplo, 1521) o ambos, la opción ResolveTNS no se tendrá en cuenta y, de todos modos, se tratará como un URI;
    - nombre de sesión - si dicho nombre se encuentra en el archivo de configuración del plugin;
    - clave TNS - si la opción ResolveTNS está establecida en true y no se aplica ninguna de las condiciones anteriores;
    - valor TNS - si comienza con el paréntesis de apertura "(" (se ignoran los espacios iniciales). En este caso, la opción ResolveTNS no se tiene en cuenta.
- **user** - el nombre de usuario de Oracle; admite añadir una de las opciones de inicio de sesión `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` o `as sysrac` en el formato `user as sysdba` (una opción de inicio de sesión no distingue entre mayúsculas y minúsculas y no debe contener un espacio final).
- **password** - la contraseña de Oracle;
- **service** - el nombre del servicio de Oracle;
- **queryName** - el nombre de una consulta personalizada; debe coincidir con el nombre del archivo SQL sin extensión;
- **args** - uno o varios argumentos separados por comas que se pasarán a la consulta.

Comentarios: - Los datos devueltos se convierten automáticamente a JSON. - Evite devolver JSON directamente desde las consultas, ya que se dañará cuando el plugin intente convertirlo de nuevo a JSON.

oracle.datafiles.stats[connString,<user>,<password>,<service>]

 Devuelve las estadísticas de los archivos de datos.
 Valor devuelto: *objeto JSON*.

Parámetros:

- **connString** - puede ser:
    - el URI: si no se encuentra ninguna sesión con el nombre proporcionado y `Plugins.Oracle.ResolveTNS` está establecido en false. Si contiene un esquema (por ejemplo, "tcp://"), un puerto (por ejemplo, 1521) o ambos, la opción ResolveTNS no se tendrá en cuenta y, en cualquier caso, se tratará como un URI;
    - nombre de sesión: si dicho nombre se encuentra en el archivo de configuración del plugin;
    - clave TNS: si la opción ResolveTNS está establecida en true y no se aplica ninguna de las condiciones anteriores;
    - valor TNS: si comienza con el paréntesis de apertura "(" (se ignoran los espacios iniciales). En este caso, la opción ResolveTNS no se tiene en cuenta.
- **user** - el nombre de usuario de Oracle; permite añadir una de las opciones de inicio de sesión `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` o `as sysrac` en el formato `user as sysdba` (una opción de inicio de sesión no distingue entre mayúsculas y minúsculas y no debe contener un espacio final).
- **password** - la contraseña de Oracle;
- **service** - el nombre del servicio de Oracle.

oracle.db.discovery[connString,<user>,<password>,<service>]

 Devuelve la lista de bases de datos. Se utiliza para el descubrimiento de bajo nivel.
 Valor devuelto: *objeto JSON*.

Parámetros:

- **connString**: puede ser:
    - el URI: si no se encuentra ninguna sesión con el nombre proporcionado y `Plugins.Oracle.ResolveTNS` está establecido en false. Si contiene un esquema (por ejemplo, "tcp://"), un puerto (por ejemplo, 1521) o ambos, la opción ResolveTNS no se tendrá en cuenta y, en cualquier caso, se tratará como un URI;

    - nombre de sesión: si dicho nombre se encuentra en el archivo de configuración del plugin;
    - clave TNS: si la opción ResolveTNS está establecida en true y no se cumple ninguna de las condiciones anteriores;
    - valor TNS: si comienza con el paréntesis de apertura "(" (se ignoran los espacios iniciales). En este caso, la opción ResolveTNS no se tiene en cuenta.
- **user**: el nombre de usuario de Oracle; permite añadir una de las opciones de inicio de sesión `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` o `as sysrac` en el formato `user as sysdba` (una opción de inicio de sesión no distingue entre mayúsculas y minúsculas y no debe contener un espacio final).
- **password**: la contraseña de Oracle;
- **service**: el nombre del servicio de Oracle.

oracle.fra.stats[connString,<user>,<password>,<service>]

 Devuelve las estadísticas del Fast Recovery Area (FRA).
 Valor de retorno: *objeto JSON*.

Parámetros:

- **connString**: puede ser:
    - el URI: si no se encuentra ninguna sesión con el nombre proporcionado y `Plugins.Oracle.ResolveTNS` está establecido en false. Si contiene un esquema (por ejemplo, "tcp://"), un puerto (por ejemplo, 1521) o ambos, la opción ResolveTNS no se tendrá en cuenta y, en cualquier caso, se tratará como un URI;
    - nombre de sesión: si dicho nombre se encuentra en el archivo de configuración del plugin;
    - clave TNS: si la opción ResolveTNS está establecida en true y no se cumple ninguna de las condiciones anteriores;
    - valor TNS: si comienza con el paréntesis de apertura "(" (se ignoran los espacios iniciales). En este caso, la opción ResolveTNS no se tiene en cuenta.
- **user**: el nombre de usuario de Oracle; admite añadir una de las opciones de inicio de sesión `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` o `as sysrac` en el formato `user as sysdba` (una opción de inicio de sesión no distingue entre mayúsculas y minúsculas y no debe contener un espacio final).
- **password**: la contraseña de Oracle;
- **service**: el nombre del servicio de Oracle.

oracle.instance.info[connString,<user>,<password>,<service>]

 Las estadísticas de la instancia.
 Valor devuelto: *objeto JSON*.

Parámetros:

- **connString** - puede ser:
    - el URI: si no se encuentra ninguna sesión con el nombre indicado y `Plugins.Oracle.ResolveTNS` está establecido en false. Si contiene un esquema (por ejemplo, "tcp://"), un puerto (por ejemplo, 1521) o ambos, la opción ResolveTNS no se tendrá en cuenta y, en cualquier caso, se tratará como un URI;
    - nombre de sesión: si dicho nombre se encuentra en el archivo de configuración del plugin;
    - clave TNS: si la opción ResolveTNS está establecida en true y no se aplica ninguna de las condiciones anteriores;
    - valor TNS: si comienza con el paréntesis de apertura "(" (se ignoran los espacios iniciales). En este caso, la opción ResolveTNS no se tiene en cuenta.
- **user** - el nombre de usuario de Oracle; admite añadir una de las opciones de inicio de sesión `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` o `as sysrac` en el formato `user as sysdba` (una opción de inicio de sesión no distingue entre mayúsculas y minúsculas y no debe contener un espacio final).
- **password** - la contraseña de Oracle;
- **service** - el nombre del servicio de Oracle.

oracle.pdb.info[connString,<user>,<password>,<service>,<database>]

 La información de las bases de datos conectables (PDB).
 Valor devuelto: *objeto JSON*.

Parámetros:

- **connString** - puede ser:
    - el URI: si no se encuentra ninguna sesión con el nombre especificado y `Plugins.Oracle.ResolveTNS` está establecido en false. Si contiene un esquema (por ejemplo, "tcp://"), un puerto (por ejemplo, 1521) o ambos, la opción ResolveTNS no se tendrá en cuenta y, de todos modos, se tratará como un URI;
    - nombre de sesión: si dicho nombre se encuentra en el archivo de configuración del plugin;
    - clave TNS: si la opción ResolveTNS está establecida en true y no se cumple ninguna de las condiciones anteriores;
    - valor TNS: si comienza con el paréntesis de apertura "(" (se ignoran los espacios iniciales). En este caso, la opción ResolveTNS no se tiene en cuenta.
- **user** - el nombre de usuario de Oracle; admite añadir una de las opciones de inicio de sesión `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` o `as sysrac` en el formato `user as sysdba` (una opción de inicio de sesión no distingue entre mayúsculas y minúsculas y no debe contener un espacio final).
- **password** - la contraseña de Oracle;
- **service** - el nombre del servicio de Oracle;

- **database** - el nombre de la base de datos que se va a consultar.

oracle.pdb.discovery[connString,<user>,<password>,<service>]

 Devuelve la lista de PDB. Se utiliza para el descubrimiento de bajo nivel.
 Valor devuelto: *objeto JSON*.

Parámetros:

- **connString** - puede ser:
  - el URI: si no se encuentra ninguna sesión con el nombre proporcionado y `Plugins.Oracle.ResolveTNS` está establecido en false. Si contiene un esquema (p. ej., "tcp://"), un puerto (p. ej., 1521) o ambos, la opción ResolveTNS no se tendrá en cuenta y, en cualquier caso, se tratará como URI;
  - nombre de sesión: si dicho nombre se encuentra en el archivo de configuración del plugin;
  - clave TNS: si la opción ResolveTNS está establecida en true y no se aplica ninguna de las condiciones anteriores;
  - valor TNS: si comienza con el paréntesis de apertura "(" (se ignoran los espacios iniciales). En este caso, la opción ResolveTNS no se tiene en cuenta.
- **user** - el nombre de usuario de Oracle; admite añadir una de las opciones de inicio de sesión `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` o `as sysrac` en el formato `user as sysdba` (una opción de inicio de sesión no distingue entre mayúsculas y minúsculas y no debe contener un espacio final).
- **password** - la contraseña de Oracle;
- **service** - el nombre del servicio de Oracle.

oracle.pga.stats[connString,<user>,<password>,<service>]

 Devuelve las estadísticas del Program Global Area (PGA).
 Valor devuelto: *objeto JSON*.

Parámetros:

- **connString** - puede ser:
  - el URI: si no se encuentra ninguna sesión con el nombre indicado y `Plugins.Oracle.ResolveTNS` está establecido en false. Si contiene un esquema (por ejemplo, "tcp://"), un puerto (por ejemplo, 1521) o ambos, la opción ResolveTNS no se tendrá en cuenta y, en cualquier caso, se tratará como un URI;
  - nombre de sesión: si dicho nombre se encuentra en el archivo de configuración del plugin;
  - clave TNS: si la opción ResolveTNS está establecida en true y no se aplica ninguna de las condiciones anteriores;
  - valor TNS: si comienza con el paréntesis de apertura "(" (se ignoran los espacios iniciales). En este caso, la opción ResolveTNS no se tiene en cuenta.
- **user** - el nombre de usuario de Oracle; admite añadir una de las opciones de inicio de sesión `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` o `as sysrac` en el formato `user as sysdba` (una opción de inicio de sesión no distingue entre mayúsculas y minúsculas y no debe contener un espacio final).
- **password** - la contraseña de Oracle;
- **service** - el nombre del servicio de Oracle.

oracle.ping[connString,<user>,<password>,<service>]

 Comprueba si se puede establecer una conexión con Oracle.
 Valor devuelto: *1* - la conexión se ha realizado correctamente; *0* - la conexión está interrumpida (si se presenta cualquier error, incluidos problemas de AUTH y de configuración).

Parámetros:

- **connString** - puede ser:
  - el URI - si no se encuentra ninguna sesión con el nombre indicado y `Plugins.Oracle.ResolveTNS` está establecido en false. Si contiene un esquema (por ejemplo, "tcp://"), un puerto (por ejemplo, 1521) o ambos, la opción ResolveTNS no se tendrá en cuenta y, en cualquier caso, se tratará como URI;
  - nombre de sesión - si dicho nombre se encuentra en el archivo de configuración del plugin;
  - clave TNS - si la opción ResolveTNS está establecida en true y no se aplica ninguna de las condiciones anteriores;
  - valor TNS - si comienza con el paréntesis de apertura "(" (se ignoran los espacios iniciales). En este caso, la opción ResolveTNS no se tiene en cuenta.
- **user** - el nombre de usuario de Oracle; admite añadir una de las opciones de inicio de sesión `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` o `as sysrac` en el formato `user as sysdba` (una opción de inicio de sesión no distingue entre mayúsculas y minúsculas y no debe contener un espacio final).
- **password** - la contraseña de Oracle;
- **service** - el nombre del servicio de Oracle.

Ejemplos:

```
oracle.ping[tcp://127.0.0.1:1521,ZABBIX_MON,zabbix,xe]
oracle.ping[localhost,ZABBIX_MON,zabbix,xe]
oracle.ping[zbx_tns_example,ZABBIX_MON,zabbix,xe]
oracle.ping["(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=localhost)(PORT=1521))(CONNECT_DATA=(SERVICE_NAME=x
```

oracle.proc.stats[connString,<user>,<password>,<service>]

 Devuelve las estadísticas de procesos.
 Valor devuelto: *objeto JSON*.

Parámetros:

- **connString** - puede ser:
  - el URI: si no se encuentra ninguna sesión con el nombre proporcionado y `Plugins.Oracle.ResolveTNS` está establecido en false. Si contiene un esquema (por ejemplo, "tcp://"), un puerto (por ejemplo, 1521) o ambos, la opción ResolveTNS no se tendrá en cuenta y, en cualquier caso, se tratará como un URI;
  - nombre de sesión: si dicho nombre se encuentra en el archivo de configuración del plugin;
  - clave TNS: si la opción ResolveTNS está establecida en true y no se aplica ninguna de las condiciones anteriores;
  - valor TNS: si comienza con el paréntesis de apertura "(" (se ignoran los espacios iniciales). En este caso, la opción ResolveTNS no se tiene en cuenta.
- **user** - el nombre de usuario de Oracle; admite añadir una de las opciones de inicio de sesión `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` o `as sysrac` en el formato `user as sysdba` (una opción de inicio de sesión no distingue entre mayúsculas y minúsculas y no debe contener un espacio final).
- **password** - la contraseña de Oracle;
- **service** - el nombre del servicio de Oracle.

oracle.redolog.info[connString,<user>,<password>,<service>]

 La información del archivo de registro del archivo de control.
 Valor devuelto: *objeto JSON*.

Parámetros:

- **connString** - puede ser:
  - el URI: si no se encuentra ninguna sesión con el nombre proporcionado y `Plugins.Oracle.ResolveTNS` está establecido en false. Si contiene un esquema (por ejemplo, "tcp://"), un puerto (por ejemplo, 1521) o ambos, la opción ResolveTNS no se tendrá en cuenta y, de todos modos, se tratará como URI;
  - nombre de sesión: si dicho nombre se encuentra en el archivo de configuración del plugin;
  - clave TNS: si la opción ResolveTNS está establecida en true y no se aplica ninguna de las condiciones anteriores;
  - valor TNS: si comienza con el paréntesis de apertura "(" (se ignoran los espacios iniciales). En este caso, la opción ResolveTNS no se tiene en cuenta.
- **user** - el nombre de usuario de Oracle; admite añadir una de las opciones de inicio de sesión `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` o `as sysrac` en el formato `user as sysdba` (una opción de inicio de sesión no distingue entre mayúsculas y minúsculas y no debe contener un espacio final).
- **password** - la contraseña de Oracle;
- **service** - el nombre del servicio de Oracle.

oracle.sga.stats[connString,<user>,<password>,<service>]

 Devuelve las estadísticas del Área Global del Sistema (SGA).
 Valor devuelto: *objeto JSON*.

Parámetros:

- **connString** - puede ser:
  - el URI: si no se encuentra ninguna sesión con el nombre proporcionado y `Plugins.Oracle.ResolveTNS` está establecido en false. Si contiene un esquema (por ejemplo, "tcp://"), un puerto (por ejemplo, 1521) o ambos, la opción ResolveTNS no se tendrá en cuenta y, en cualquier caso, se tratará como un URI;
  - nombre de sesión: si dicho nombre se encuentra en el archivo de configuración del plugin;
  - clave TNS: si la opción ResolveTNS está establecida en true y no se aplica ninguna de las condiciones anteriores;
  - valor TNS: si comienza con el paréntesis de apertura "(" (se ignoran los espacios iniciales). En este caso, la opción ResolveTNS no se tiene en cuenta.
- **user** - el nombre de usuario de Oracle; admite añadir una de las opciones de inicio de sesión `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` o `as sysrac` en el formato `user as sysdba` (una opción de inicio de sesión no distingue entre mayúsculas y minúsculas y no debe contener un espacio final).
- **password** - la contraseña de Oracle;
- **service** - el nombre del servicio de Oracle.

oracle.sessions.stats[connString,<user>,<password>,<service>,<lockMaxTime>]

 Devuelve las estadísticas de las sesiones.
 Valor devuelto: *objeto JSON*.

Parámetros:

- **connString** - puede ser:
  - el URI: si no se encuentra ninguna sesión con el nombre indicado y `Plugins.Oracle.ResolveTNS` está establecido en false. Si contiene un esquema (por ejemplo, "tcp://"), un puerto (por ejemplo, 1521) o ambos, la opción ResolveTNS no se tendrá en cuenta y, en cualquier caso, se tratará como un URI;

    - nombre de sesión: si dicho nombre se encuentra en el archivo de configuración del plugin;
    - clave TNS: si la opción ResolveTNS está establecida en true y no se cumple ninguna de las condiciones anteriores;
    - valor TNS: si comienza con el paréntesis de apertura "(" (se ignoran los espacios iniciales). En este caso, la opción ResolveTNS no se tiene en cuenta.
- **user** - el nombre de usuario de Oracle; permite añadir una de las opciones de inicio de sesión `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` o `as sysrac` en el formato `user as sysdba` (una opción de inicio de sesión no distingue entre mayúsculas y minúsculas y no debe contener un espacio final).
- **password** - la contraseña de Oracle;
- **service** - el nombre del servicio de Oracle;
- **lockMaxTime** - la duración máxima del bloqueo de la sesión, en segundos, para contar la sesión como bloqueada de forma prolongada. Valor predeterminado: 600 segundos.

oracle.sys.metrics[connString,<user>,<password>,<service>,<duration>]

 Devuelve un conjunto de valores de métricas del sistema.
 Valor devuelto: *objeto JSON*.

Parámetros:

- **connString** - puede ser:
    - la URI: si no se encuentra ninguna sesión con el nombre indicado y `Plugins.Oracle.ResolveTNS` está establecido en false. Si contiene un esquema (por ejemplo, "tcp://"), un puerto (por ejemplo, 1521) o ambos, la opción ResolveTNS no se tendrá en cuenta y, en cualquier caso, se tratará como una URI;
    - nombre de sesión: si se encuentra dicho nombre en el archivo de configuración del plugin;
    - clave TNS: si la opción ResolveTNS está establecida en true y no se cumple ninguna de las condiciones anteriores;
    - valor TNS: si comienza con el paréntesis de apertura "(" (se ignoran los espacios iniciales). En este caso, la opción ResolveTNS no se tendrá en cuenta.
- **user** - el nombre de usuario de Oracle; admite añadir una de las opciones de inicio de sesión `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` o `as sysrac` con el formato `user as sysdba` (la opción de inicio de sesión no distingue entre mayúsculas y minúsculas y no debe contener un espacio final).
- **password** - la contraseña de Oracle;
- **service** - el nombre del servicio de Oracle;
- **duration** - el intervalo de captura (en segundos) de los valores de las métricas del sistema. Valores posibles: *60* - duración larga (predeterminada), *15* - duración corta.

oracle.sys.params[connString,<user>,<password>,<service>]

 Devuelve un conjunto de valores de parámetros del sistema.
 Valor de retorno: *objeto JSON*.

Parámetros:

- **connString** - puede ser:
    - el URI: si no se encuentra ninguna sesión con el nombre indicado y `Plugins.Oracle.ResolveTNS` está establecido en false. Si contiene un esquema (por ejemplo, "tcp://"), un puerto (por ejemplo, 1521) o ambos, la opción ResolveTNS no se tendrá en cuenta y, de todos modos, se tratará como un URI;
    - nombre de sesión: si dicho nombre se encuentra en el archivo de configuración del plugin;
    - clave TNS: si la opción ResolveTNS está establecida en true y no se cumple ninguna de las condiciones anteriores;
    - valor TNS: si comienza con el paréntesis de apertura "(" (se ignoran los espacios iniciales). En este caso, la opción ResolveTNS no se tiene en cuenta.
- **user** - el nombre de usuario de Oracle; admite añadir una de las opciones de inicio de sesión `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` o `as sysrac` en el formato `user as sysdba` (una opción de inicio de sesión no distingue entre mayúsculas y minúsculas y no debe contener un espacio final).
- **password** - la contraseña de Oracle;
- **service** - el nombre del servicio de Oracle.

oracle.ts.stats[connString,<user>,<password>,<service>,<tablespace>,<type>,<conname>]

 Devuelve las estadísticas de los tablespaces.
 Valor devuelto: *objeto JSON*.

Parámetros:

- **connString** - puede ser:
    - el URI: si no se encuentra ninguna sesión con el nombre indicado y `Plugins.Oracle.ResolveTNS` está establecido en false. Si contiene un esquema (por ejemplo, "tcp://"), un puerto (por ejemplo, 1521) o ambos, la opción ResolveTNS no se tendrá en cuenta y, en cualquier caso, se tratará como URI;
    - nombre de sesión: si dicho nombre se encuentra en el archivo de configuración del plugin;
    - clave TNS: si la opción ResolveTNS está establecida en true y no se cumple ninguna de las condiciones anteriores;
    - valor TNS: si comienza con el paréntesis de apertura "(" (ignorando los espacios iniciales). En este caso, la opción ResolveTNS no se tiene en cuenta.

- **user** - el nombre de usuario de Oracle; admite añadir una de las opciones de inicio de sesión `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` o `as sysrac` en el formato `user as sysdba` (una opción de inicio de sesión no distingue entre mayúsculas y minúsculas y no debe contener un espacio final).
- **password** - la contraseña de Oracle;
- **service** - el nombre del servicio de Oracle;
- **tablespace** - nombre del tablespace que se va a consultar. Valor predeterminado (si se deja vacío y `type` está establecido):
  - "TEMP" (si `type` está establecido en "TEMPORARY");
  - "USERS" (si `type` está establecido en "PERMANENT").
- **type** - el tipo de tablespace que se va a consultar. Valor predeterminado (si `tablespace` está establecido): "PERMANENT".
- **conname** - nombre del contenedor para el que se requiere la información.

Si se omite `tablespace`, `type` o `conname`, el item devolverá estadísticas de tablespaces para todos los contenedores coincidentes (incluidos PDB y CDB).

oracle.ts.discovery[connString,<user>,<password>,<service>]

 Devuelve una lista de tablespaces. Se utiliza para el descubrimiento de bajo nivel.
 Valor devuelto: *objeto JSON*.

Parámetros:

- **connString** - puede ser:
  - el URI: si no se encuentra ninguna sesión con el nombre indicado y `Plugins.Oracle.ResolveTNS` está establecido en false. Si contiene un esquema (p. ej., "tcp://"), un puerto (p. ej., `1521`) o ambos, la opción ResolveTNS no se tendrá en cuenta y, en cualquier caso, se tratará como URI;
  - nombre de sesión: si dicho nombre se encuentra en el archivo de configuración del plugin;
  - clave TNS: si la opción ResolveTNS está establecida en true y no se aplica ninguna de las condiciones anteriores;
  - valor TNS: si comienza con el paréntesis de apertura "(" (ignorando los espacios iniciales). En este caso, la opción ResolveTNS no se tiene en cuenta.
- **user** - el nombre de usuario de Oracle; admite añadir una de las opciones de inicio de sesión `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` o `as sysrac` en el formato `user as sysdba` (una opción de inicio de sesión no distingue entre mayúsculas y minúsculas y no debe contener un espacio final).
- **password** - la contraseña de Oracle;
- **service** - el nombre del servicio de Oracle.

oracle.user.info[connString,<user>,<password>,<service>,<username>]

 Devuelve información del usuario de Oracle.
 Valor devuelto: *objeto JSON*.

Parámetros:

- **connString** - puede ser:
  - el URI: si no se encuentra ninguna sesión con el nombre indicado y `Plugins.Oracle.ResolveTNS` está establecido en false. Si contiene un esquema (por ejemplo, "tcp://"), un puerto (por ejemplo, `1521`) o ambos, la opción ResolveTNS no se tendrá en cuenta y, en cualquier caso, se tratará como un URI;
  - nombre de sesión: si dicho nombre se encuentra en el archivo de configuración del plugin;
  - clave TNS: si la opción ResolveTNS está establecida en true y no se cumple ninguna de las condiciones anteriores;
  - valor TNS: si comienza con el paréntesis de apertura "(" (se ignoran los espacios iniciales). En este caso, la opción ResolveTNS no se tiene en cuenta.
- **user** - el nombre de usuario de Oracle; admite añadir una de las opciones de inicio de sesión `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` o `as sysrac` en el formato `user as sysdba` (una opción de inicio de sesión no distingue entre mayúsculas y minúsculas y no debe contener un espacio final).
- **password** - la contraseña de Oracle;
- **service** - el nombre del servicio de Oracle;
- **username** - el nombre de usuario para el que se necesita la información. No se admiten nombres de usuario en minúsculas. Predeterminado: usuario actual.

oracle.version[connString,<user>,<password>,<service>]

 Devuelve la versión del server de base de datos.
 Valor devuelto: *String*.

Parámetros:

- **connString** - puede ser:
  - el URI: si no se encuentra ninguna sesión con el nombre indicado y `Plugins.Oracle.ResolveTNS` está establecido en false. Si contiene un esquema (por ejemplo, "tcp://"), un puerto (por ejemplo, `1521`) o ambos, la opción ResolveTNS no se tendrá en cuenta y, de todos modos, se tratará como un URI;
  - nombre de sesión: si dicho nombre se encuentra en el archivo de configuración del plugin;
  - clave TNS: si la opción ResolveTNS está establecida en true y no se aplica ninguna de las condiciones anteriores;

  - valor TNS: si comienza con el paréntesis de apertura "(" (se ignoran los espacios iniciales). En este caso, la opción ResolveTNS no se tiene en cuenta.
- **user** - el nombre de usuario de Oracle; admite añadir una de las opciones de inicio de sesión `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` o `as sysrac` en el formato `user as sysdba` (una opción de inicio de sesión no distingue entre mayúsculas y minúsculas y no debe contener un espacio final).
- **password** - la contraseña de Oracle;
- **service** - el nombre del servicio de Oracle.

pgsql.autovacuum.count[uri,<username>,<password>,<database name>]

 El número de workers de autovacuum.
 Valor devuelto: *Integer*.

Parámetros:

- **uri** - la URI o el nombre de sesión;
- **username, password** - las credenciales de PostgreSQL;
- **database name** - el nombre de la base de datos (si se omite, se usa el valor de `Plugins.PostgreSQL.Default.Database` de `postgresql.conf`).

pgsql.archive[uri,<username>,<password>,<database name>]

 Información sobre los archivos archivados.
 Valor devuelto: *objeto JSON*.

Parámetros:

- **uri** - el URI o nombre de sesión;
- **username, password** - las credenciales de PostgreSQL;
- **database name** - el nombre de la base de datos (si se omite, se usa el valor de `Plugins.PostgreSQL.Default.Database` de `postgresql.conf`).

pgsql.bgwriter[uri,<username>,<password>,<database name>]

 El número total de checkpoints para el clúster de bases de datos, desglosado por tipo de checkpoint.
 Valor devuelto: *objeto JSON*.

Parámetros:

- **uri** - la URI o el nombre de sesión;
- **username, password** - las credenciales de PostgreSQL;
- **database name** - el nombre de la base de datos (si se omite, se usa el valor de `Plugins.PostgreSQL.Default.Database` de `postgresql.conf`).

pgsql.cache.hit[uri,<username>,<password>,<database name>]

 La tasa de aciertos de la caché de búfer de PostgreSQL.
 Valor devuelto: *Float*.

Parámetros:

- **uri** - la URI o el nombre de sesión;
- **username, password** - las credenciales de PostgreSQL;
- **database name** - el nombre de la base de datos (si se omite, se usa el valor de `Plugins.PostgreSQL.Default.Database` de `postgresql.conf`).

pgsql.connections[uri,<username>,<password>,<database name>]

 Devuelve conexiones por tipo.
 Valor devuelto: *objeto JSON*.

Parámetros:

- **uri** - la URI o el nombre de sesión;
- **username, password** - las credenciales de PostgreSQL;
- **database name** - el nombre de la base de datos (si se omite, se usa el valor de `Plugins.PostgreSQL.Default.Database` de `postgresql.conf`).

pgsql.custom.query[uri,<username>,<password>,<database name>,queryName,<args...>]

 Devuelve el resultado de una consulta personalizada.
 Valor de retorno: *objeto JSON*.

Parámetros:

- **uri** - el URI o nombre de sesión;
- **username, password** - las credenciales de PostgreSQL;
- **database name** - el nombre de la base de datos (si se omite, se usa el valor de Plugins.PostgreSQL.Default.Database de postgresql.conf);
- **queryName** - el nombre de una consulta personalizada, debe coincidir con el nombre del archivo SQL sin la extensión;

- **args** - uno o varios argumentos separados por comas para pasar a una consulta.

pgsql.db.age[uri,<username>,<password>,<database name>]

 La antigüedad del FrozenXID más antiguo de la base de datos (número de transacciones).
 Valor devuelto: *Integer*.

Parámetros:

- **uri** - la URI o el nombre de la sesión;
- **username, password** - las credenciales de PostgreSQL;
- **database name** - el nombre de la base de datos (si se omite, se usa el valor de `Plugins.PostgreSQL.Default.Database` de `postgresql.conf`).

pgsql.db.bloating_tables[uri,<username>,<password>,<database name>]

 El número de tablas con bloat por base de datos.
 Valor de retorno: *Integer*.

Parámetros:

- **uri** - el URI o nombre de sesión;
- **username, password** - las credenciales de PostgreSQL;
- **database name** - el nombre de la base de datos (si se omite, se usa el valor de `Plugins.PostgreSQL.Default.Database` de `postgresql.conf`).

pgsql.db.discovery[uri,<username>,<password>,<database name>]

 La lista de bases de datos de PostgreSQL. Se utiliza para low-level discovery.
 Valor devuelto: *objeto JSON*.

Parámetros:

- **uri** - el URI o nombre de sesión;
- **username, password** - las credenciales de PostgreSQL;
- **database name** - el nombre de la base de datos (si se omite, se usa el valor de `Plugins.PostgreSQL.Default.Database` de `postgresql.conf`).

pgsql.db.size[uri,<username>,<password>,<database name>]

 El tamaño de la base de datos en bytes.
 Valor devuelto: *Integer*.

Parámetros:

- **uri** - la URI o el nombre de sesión;
- **username, password** - las credenciales de PostgreSQL;
- **database name** - el nombre de la base de datos (si se omite, se usa el valor de `Plugins.PostgreSQL.Default.Database` de `postgresql.conf`).

pgsql.dbstat[uri,<username>,<password>,<database name>]

 Recopila las estadísticas por base de datos. Se utiliza para descubrimiento de bajo nivel.
 Valor devuelto: *objeto JSON*.

Parámetros:

- **uri** - el URI o nombre de sesión;
- **username, password** - las credenciales de PostgreSQL;
- **database name** - el nombre de la base de datos (si se omite, se usa el valor de `Plugins.PostgreSQL.Default.Database` de `postgresql.conf`).

pgsql.dbstat.sum[uri,<username>,<password>,<database name>]

 Los datos resumidos de todas las bases de datos en un clúster.
 Valor devuelto: *objeto JSON*.

Parámetros:

- **uri** - la URI o el nombre de sesión;
- **username, password** - las credenciales de PostgreSQL;
- **database name** - el nombre de la base de datos (si se omite, se usa el valor de `Plugins.PostgreSQL.Default.Database` de `postgresql.conf`).

pgsql.locks[uri,<username>,<password>,<database name>]

 Información sobre los bloqueos concedidos por base de datos. Se utiliza para descubrimiento de bajo nivel.
 Valor devuelto: *objeto JSON*.

Parámetros:

- **uri** - el URI o nombre de sesión;
- **username, password** - las credenciales de PostgreSQL;

- **database name** - el nombre de la base de datos (si se omite, se usa el valor de `Plugins.PostgreSQL.Default.Database` de `postgresql.conf`).

pgsql.oldest.xid[uri,<username>,<password>,<database name>]

 La antigüedad del XID más antiguo (número de transacciones).
 Valor devuelto: *Integer*.

Parámetros:

- **uri** - la URI o el nombre de sesión;
- **username, password** - las credenciales de PostgreSQL;
- **database name** - el nombre de la base de datos (si se omite, se usa el valor de `Plugins.PostgreSQL.Default.Database` de `postgresql.conf`).

pgsql.ping[uri,<username>,<password>,<database name>]

 Comprueba si una conexión está activa o no.
 Valor devuelto: *1* - la conexión está activa; *0* - la conexión está rota (si se presenta cualquier error, incluidos problemas de AUTH y de configuración).

Parámetros:

- **uri** - la URI o el nombre de sesión;
- **username, password** - las credenciales de PostgreSQL;
- **database name** - el nombre de la base de datos (si se omite, se usa el valor de `Plugins.PostgreSQL.Default.Database` de `postgresql.conf`).

pgsql.queries[uri,<username>,<password>,<database name>,time period]

 Consulta métricas por tiempo de ejecución.
 Valor devuelto: *objeto JSON*.

Parámetros:

- **uri** - la URI o el nombre de sesión;
- **username, password** - las credenciales de PostgreSQL;
- **database name** - el nombre de la base de datos (si se omite, se usa el valor de `Plugins.PostgreSQL.Default.Database` de `postgresql.conf`);
- **time period** - el límite de tiempo de ejecución para el recuento de consultas lentas (debe ser un entero positivo).

pgsql.replication.count[uri,<usuario>,<contraseña>]

 El número de servidores en espera.
 Valor de retorno: *Entero*.

Parámetros:

- **uri**: el URI o nombre de la sesión;
- **usuario, contraseña**: las credenciales de PostgreSQL.

pgsql.replication.process[uri,<usuario>,<contraseña>]

 El retraso de vaciado, retraso de escritura y retraso de reproducción por cada proceso de envío.
 Valor devuelto: *objeto JSON*.

Parámetros:

- **uri** - la URI o el nombre de la sesión;
- **usuario, contraseña** - las credenciales de PostgreSQL.

pgsql.replication.process.discovery[uri,<usuario>,<contraseña>]

 El descubrimiento del nombre del proceso de replicación.
 Valor de retorno: *objeto JSON*.

Parámetros:

- **uri**: el URI o nombre de la sesión;
- **usuario, contraseña**: las credenciales de PostgreSQL.

pgsql.replication.recovery_role[uri,<usuario>,<contraseña>]

 El estado de recuperación.
 Valor de retorno: *0* - modo maestro; *1* - la recuperación aún está en curso (modo en espera).

Parámetros:

- **uri** - la URI o el nombre de la sesión;
- **usuario, contraseña** - las credenciales de PostgreSQL.

pgsql.replication.status[uri,<usuario>,<contraseña>]

 El estado de la replicación.
 Valor devuelto: *0* - la transmisión está caída; *1* - la transmisión está activa; *2* - modo maestro.

Parámetros:

- **uri** - el URI o nombre de la sesión;
- **usuario, contraseña** - las credenciales de PostgreSQL.

pgsql.replication_lag.b[uri,<usuario>,<contraseña>]

 El retardo de replicación en bytes.
 Valor de retorno: *Entero*.

Parámetros:

- **uri**: el URI o nombre de la sesión;
- **usuario, contraseña**: las credenciales de PostgreSQL.

pgsql.replication_lag.sec[uri,<usuario>,<contraseña>]

 El retardo de replicación en segundos.
 Valor de retorno: *Entero*.

Parámetros:

- **uri** - la URI o el nombre de la sesión;
- **usuario, contraseña** - las credenciales de PostgreSQL.

pgsql.uptime[uri,<username>,<password>,<database name>]

 El tiempo de actividad de PostgreSQL en milisegundos.
 Valor devuelto: *Float*.

Parámetros:

- **uri** - la URI o el nombre de sesión;
- **username, password** - las credenciales de PostgreSQL;
- **database name** - el nombre de la base de datos (si se omite, se usa el valor de `Plugins.PostgreSQL.Default.Database` de `postgresql.conf`).

pgsql.version[uri,<username>,<password>,<database name>]

 Devuelve la versión de PostgreSQL.
 Valor de retorno: *String*.

Parámetros:

- **uri** - el URI o nombre de sesión;
- **username, password** - las credenciales de PostgreSQL;
- **database name** - el nombre de la base de datos (si se omite, se usa el valor de `Plugins.PostgreSQL.Default.Database` de `postgresql.conf`).

pgsql.wal.stat[uri,<username>,<password>,<database name>]

 Las estadísticas de WAL.
 Valor devuelto: *objeto JSON*.

Parámetros:

- **uri** - el URI o nombre de sesión;
- **username, password** - las credenciales de PostgreSQL;
- **database name** - el nombre de la base de datos (si se omite, se usa el valor de `Plugins.PostgreSQL.Default.Database` de `postgresql.conf`).

redis.config[connString,<password>,<pattern>]

 Obtiene los parámetros de configuración de una instancia de Redis que coinciden con el patrón.
 Valor de retorno: *JSON* - si se utilizó un patrón de tipo glob; valor único - si el patrón no contenía ningún carácter comodín.

Parámetros:

- **connString** - el URI o nombre de sesión;
- **password** - la contraseña de Redis;
- **pattern** - un patrón de tipo glob (* por defecto).

Comentario:

- Advertencia de seguridad: `redis.config` ejecuta el comando `CONFIG GET` de Redis y puede devolver parámetros de configuración sensibles (por ejemplo, `requirepass`, `masterauth`, configuraciones relacionadas con TLS y otros secretos). El uso del patrón por defecto (`*`) o patrones amplios puede exponer estos secretos a cualquier persona con permiso para ver los valores de los items en Zabbix.

redis.info[connString,<password>,<section>]

 Obtiene la salida del comando INFO.
 Valor de retorno: *JSON* - la salida se serializa como JSON.

Parámetros:

- **connString** - el URI o nombre de sesión;
- **password** - la contraseña de Redis;
- **section** - la sección de información (*default* por defecto).

redis.ping[connString,<password>]

 Prueba si una conexión está activa o no.
 Valor de retorno: *1* - la conexión está activa; *0* - la conexión está rota (si se presenta algún error, incluyendo problemas de AUTH y configuración).

Parámetros:

- **connString** - el URI o nombre de sesión;
- **password** - la contraseña de Redis.

redis.slowlog.count[connString,<password>]

 El número de entradas de registro lento desde que se inició Redis.
 Valor de retorno: *Entero*.

Parámetros:

- **connString** - el URI o nombre de sesión;
- **password** - la contraseña de Redis.

smart.attribute.discovery

 Devuelve una lista de atributos de dispositivo S.M.A.R.T.
 Valor devuelto: *JSON object*.

Comentarios:

- Se devuelven las siguientes macros y sus valores: `{#NAME}`, `{#DISKTYPE}`, `{#ID}`, `{#ATTRNAME}`, `{#THRESH}`;
- Se admiten tipos de unidades HDD, SSD y NVME. Las unidades pueden estar solas o combinadas en un RAID. `{#NAME}` tendrá un sufijo en caso de RAID, por ejemplo: `{"{#NAME}": "/dev/sda cciss,2"}`.

smart.disk.discovery[<type>]

 Devuelve una lista de dispositivos S.M.A.R.T.
 Valor de retorno: *JSON object*.

Parámetro:

- **type** - especifica un valor para buscar en los discos. Valores posibles: *id*, *name* (predeterminado). *id* no es compatible en Windows y devolverá un error si se selecciona

Comentarios:

- Se devuelven las siguientes macros y sus valores: `{#NAME}`, `{#DISKTYPE}`, `{#MODEL}`, `{#SN}`, `{#PATH}`, `{#ATTRIBUTES}`, `{#RAIDTYPE}`;
- Se admiten los tipos de unidad HDD, SSD y NVME. Si una unidad no pertenece a un RAID, `{#RAIDTYPE}` estará vacío. `{#NAME}` tendrá un complemento en caso de RAID, por ejemplo: `{"{#NAME}": "/dev/sda cciss,2"}`.

smart.disk.get[<path>,<raid type>]

 Devuelve todas las propiedades disponibles de los dispositivos S.M.A.R.T.
 Valor de retorno: *objeto JSON*.

Parámetros:

- **path** - la ruta del disco, se puede usar la macro `{#PATH}` como valor;
- **raid_type** - el tipo de RAID, se puede usar la macro `{#RAID}` como valor

Comentarios:

- Se admiten los tipos de unidad HDD, SSD y NVME. Las unidades pueden estar solas o combinadas en un RAID;
- Los datos incluyen la versión de smartctl y los argumentos de llamada, además de campos adicionales:
disk_name - contiene el nombre con los complementos necesarios para la detección de RAID, por ejemplo: `{"disk_name": "/dev/sda cciss,2"}`
disk_type - contiene el tipo de disco HDD, SSD o NVME, por ejemplo: `{"disk_type": "ssd"}`;
- Si no se especifican parámetros, el item devolverá información sobre todos los discos.

systemd.unit.get[unit name,<interface>]

 Devuelve todas las propiedades de una unidad systemd.
 Valor devuelto: *objeto JSON*.

Parámetros:

- **unit name** - el nombre de la unidad (puede que desee usar la macro {#UNIT.NAME} en el prototipo de item para descubrir el nombre);
- **interface** - el tipo de interfaz de la unidad, valores posibles: *Unit* (predeterminado), *Service*, *Socket*, *Device*, *Mount*, *Automount*, *Swap*, *Target*, *Path*.

Comentarios:

- Este item solo es compatible en la plataforma Linux;
- LoadState, ActiveState y UnitFileState para la interfaz Unit se devuelven como texto e integer: `"ActiveState":{"state":1,"text"`

systemd.unit.info[unit name,<property>,<interface>]

 Información de una unidad de systemd.
 Valor devuelto: *String*.

Parámetros:

- **unit name**: el nombre de la unidad (puede utilizar la macro {#UNIT.NAME} en el prototipo de item para descubrir el nombre);
- **property**: propiedad de la unidad (por ejemplo, ActiveState (predeterminada), LoadState, Description);
- **interface**: tipo de interfaz de la unidad (por ejemplo, Unit (predeterminada), Socket, Service).

Comentarios:

- Este item solo es compatible con la plataforma Linux;
- Este item permite recuperar una propiedad específica de un tipo específico de interfaz, tal como se describe en la API de dbus.

Ejemplos:

```
systemd.unit.info["{#UNIT.NAME}"] #recopilar información sobre el estado activo (active, reloading, inacti
systemd.unit.info["{#UNIT.NAME}",LoadState] #recopilar información sobre el estado de carga de las unidade
systemd.unit.info[mysqld.service,Id] #recuperar el nombre técnico del servicio (mysqld.service)
systemd.unit.info[mysqld.service,Description] #recuperar la descripción del servicio (MySQL Server)
systemd.unit.info[mysqld.service,ActiveEnterTimestamp] #recuperar la última vez que el servicio entró en e
systemd.unit.info[dbus.socket,NConnections,Socket] #recopilar el número de conexiones de esta unidad de so
```

systemd.unit.discovery[<tipo>]

 Lista de unidades de systemd y sus detalles. Se utiliza para descubrimiento de bajo nivel.
 Valor de retorno: *objeto JSON*.

Parámetro:

- **tipo**: valores posibles: *all*, *automount*, *device*, *mount*, *path*, *service* (por defecto), *socket*, *swap*, *target*.

Este elemento solo es compatible con la plataforma Linux.

web.certificate.get[hostname,<port>,<address>]

 Valida los certificados y devuelve los detalles del certificado.
 Valor devuelto: *objeto JSON*.

Parámetros:

- **hostname** - puede ser una IP o DNS.
Puede contener el esquema de la URL (*solo* https), la ruta (se ignorará) y el puerto.
Si se proporciona un puerto tanto en el primer como en el segundo parámetro, sus valores deben coincidir.
Si se especifica address (el 3.er parámetro), hostname solo se usa para SNI y la verificación del hostname;
- **port** - el número de puerto (predeterminado: 443 para HTTPS);
- **address** - puede ser una IP o DNS. Si se especifica, se usará para la conexión, y hostname (el 1.er parámetro) se usará para SNI y la verificación del host. En caso de que el 1.er parámetro sea una IP y el 3.er parámetro sea DNS, el 1.er parámetro se usará para la conexión y el 3.er parámetro se usará para SNI y la verificación del host.

Comentarios:

- Este item pasa a no compatible si el destino especificado en la configuración del host no existe, no está disponible o si el handshake TLS falla con cualquier error excepto un certificado no válido;
- Actualmente, no se admiten la extensión X.509 AIA (Authority Information Access), las CRL y OCSP (incluido OCSP stapling), ni Certificate Transparency;
- Campos de la respuesta JSON:
  - *x509*: contiene los detalles del certificado X.509.
    - *version*: la versión X.509 (por ejemplo, 3).
    - *serial_number*: el número de serie del certificado.
    - *signature_algorithm*: el algoritmo usado para firmar el certificado (por ejemplo, SHA256-RSA).
    - *issuer*: el emisor del certificado.

        * *not_before*: la fecha de inicio de la validez del certificado.
        * *not_after*: la fecha de expiración del certificado.
        * *subject*: el sujeto del certificado.
        * *public_key_algorithm*: el algoritmo usado para la clave pública (por ejemplo, RSA).
        * *alternative_names*: nombres alternativos del sujeto (si están presentes); en caso contrario, `null`.
      - result: contiene el resultado de la validación.
        * *value*: el estado de la validación (consulte los posibles valores más abajo).
        * *message*: mensaje detallado de validación (por ejemplo, `"certificate verified successfully"`).
      - *sha1_fingerprint*: la huella digital SHA-1 del certificado.
      - *sha256_fingerprint*: la huella digital SHA-256 del certificado.
  - El campo `$.result.value` indica el resultado de la validación del certificado. Los valores posibles incluyen:
      - *valid* - el certificado es válido y de confianza.
      - *valid-but-self-signed* - el certificado es válido pero autofirmado, lo que significa que su sujeto coincide con su emisor.
      - *invalid* - el certificado no es válido debido a un problema como expiración, hostname incorrecto o una autoridad de firma desconocida.

Ejemplo:

```
web.certificate.get[example.com,443]
```

Respuesta JSON:


```
{
  "x509": {
    "version": 3,
    "serial_number": "0ad893bafa68b0b7fb7a404f06ecaf9a",
    "signature_algorithm": "ECDSA-SHA384",
    "issuer": "CN=DigiCert Global G3 TLS ECC SHA384 2020 CA1,O=DigiCert Inc,C=US",
    "not_before": {
      "value": "Jan 15 00:00:00 2025 GMT",
      "timestamp": 1736899200
    },
    "not_after": {
      "value": "Jan 15 23:59:59 2026 GMT",
      "timestamp": 1768521599
    },
    "subject": "CN=*.example.com,O=Internet Corporation for Assigned Names and Numbers,L=Los Angeles,ST=Ca
    "public_key_algorithm": "ECDSA",
    "alternative_names": [
      "*.example.com",
      "example.com"
    ]
  },
  "result": {
    "value": "valid",
    "message": "certificate verified successfully"
  },
  "sha1_fingerprint": "310db7af4b2bc9040c8344701aca08d0c69381e3",
  "sha256_fingerprint": "455943cf819425761d1f950263ebf54755d8d684c25535943976f488bc79d23b"
}
```


Windows Zabbix agent

Descripción general

Los items del agente de Zabbix para Windows se presentan en dos listas:

- Items compartidos: las claves de item que se comparten con el agente de Zabbix para UNIX;
- Items específicos de Windows: las claves de item que se admiten **solo** en Windows.

Tenga en cuenta que todas las claves de item admitidas por el agente de Zabbix en Windows también son compatibles con el agente de Zabbix de nueva generación, agent 2. Consulte las claves de item adicionales que puede utilizar solo con el agente 2.

Consulte también: Permisos mínimos para items de Windows

Items compartidos

La tabla siguiente enumera los items de Zabbix agent compatibles en Windows y compartidos con el Zabbix agent de UNIX:

- La clave del item es un enlace a los detalles completos del item de Zabbix agent de UNIX
- Se incluyen comentarios del item relevantes para Windows

Item key	Description	Item group
log	La supervisión de un archivo de registro. Este item no es compatible con Windows Event Log. El parámetro `persistent_dir` no es compatible en Windows.	Log monitoring
log.count	El recuento de líneas coincidentes en un archivo de registro supervisado. Este item no es compatible con Windows Event Log. El parámetro `persistent_dir` no es compatible en Windows.	
logrt	La supervisión de un archivo de registro rotado. Este item no es compatible con Windows Event Log. El parámetro `persistent_dir` no es compatible en Windows.	
logrt.count	El recuento de líneas coincidentes en un archivo de registro rotado supervisado. Este item no es compatible con Windows Event Log. El parámetro `persistent_dir` no es compatible en Windows.	
modbus.get	Lee datos Modbus.	Modbus
net.dns	Comprueba si el servicio DNS está activo. Los parámetros `ip`, `timeout` y `count` se ignoran en Windows, salvo que se use Zabbix agent 2.	Network
net.dns.perf	Comprueba el rendimiento de un servicio DNS. Los parámetros `ip`, `timeout` y `count` se ignoran en Windows, salvo que se use Zabbix agent 2.	
net.dns.record	Realiza una consulta DNS. Los parámetros `ip`, `timeout` y `count` se ignoran en Windows, salvo que se use Zabbix agent 2.	
net.if.discovery	La lista de interfaces de red. Algunas versiones de Windows (por ejemplo, Server 2008) pueden requerir que estén instaladas las últimas actualizaciones para admitir caracteres no ASCII en los nombres de interfaz.	
net.if.in	Las estadísticas del tráfico entrante en una interfaz de red. En Windows, el item obtiene valores de contadores de 64 bits si están disponibles. Los contadores de estadísticas de interfaz de 64 bits se introdujeron en Windows Vista y Windows Server 2008. Si no hay contadores de 64 bits disponibles, el agent usa contadores de 32 bits. Se admiten nombres de interfaz multibyte en Windows. Puede obtener descripciones de interfaces de red en Windows con los items net.if.discovery o net.if.list.	
net.if.out	Las estadísticas del tráfico saliente en una interfaz de red. En Windows, el item obtiene valores de contadores de 64 bits si están disponibles. Los contadores de estadísticas de interfaz de 64 bits se introdujeron en Windows Vista y Windows Server 2008. Si no hay contadores de 64 bits disponibles, el agent usa contadores de 32 bits. Se admiten nombres de interfaz multibyte en Windows. Puede obtener descripciones de interfaces de red en Windows con los items net.if.discovery o net.if.list.	
net.if.total	La suma de las estadísticas del tráfico entrante y saliente en una interfaz de red. En Windows, el item obtiene valores de contadores de 64 bits si están disponibles. Los contadores de estadísticas de interfaz de 64 bits se introdujeron en Windows Vista y Windows Server 2008. Si no hay contadores de 64 bits disponibles, el agent usa contadores de 32 bits. Se admiten nombres de interfaz multibyte en Windows. Puede obtener descripciones de interfaces de red en Windows con los items net.if.discovery o net.if.list.	
net.tcp.listen	Comprueba si este puerto TCP está en estado LISTEN.	
net.tcp.port	Comprueba si es posible establecer una conexión TCP con el puerto especificado.	
net.tcp.service	Comprueba si un servicio está en ejecución y acepta conexiones TCP. La comprobación de LDAP y HTTPS en Windows solo es compatible con Zabbix agent 2.	

Item key	Description	Item group
net.tcp.service.perf	Comprueba el rendimiento de un servicio TCP. La comprobación de LDAP y HTTPS en Windows solo es compatible con Zabbix agent 2.	
net.tcp.socket.count	Devuelve el número de sockets TCP que coinciden con los parámetros. Este item es compatible en Linux con Zabbix agent, pero en Windows solo es compatible con Zabbix agent 2 en Windows de 64 bits.	
net.udp.service	Comprueba si un servicio está en ejecución y responde a solicitudes UDP.	
net.udp.service.perf	Comprueba el rendimiento de un servicio UDP.	
net.udp.socket.count	Devuelve el número de sockets UDP que coinciden con los parámetros. Este item es compatible en Linux con Zabbix agent, pero en Windows solo es compatible con Zabbix agent 2 en Windows de 64 bits.	
proc.get	La lista de procesos del sistema operativo y sus parámetros. El parámetro `cmdline` no es compatible en Windows.	Processes
proc.num	El número de procesos. En Windows, solo se admiten los parámetros `name` y `user`.	
system.cpu.discovery	La lista de CPU/núcleos de CPU detectados.	System
system.cpu.load	La carga de CPU. Cuando se inicia un proceso colector en Zabbix agent, se inicializan los siguientes contadores de rendimiento y luego se usan para este item: `\System\Processor Queue Length`	
system.cpu.num	El número de CPU.	
system.cpu.util	El porcentaje de utilización de CPU. El valor se obtiene mediante el contador de rendimiento *Processor Time*. Tenga en cuenta que, desde Windows 8, su Administrador de tareas muestra la utilización de CPU basada en el contador de rendimiento *Processor Utility*, mientras que en versiones anteriores era el contador *Processor Time* (consulte más detalles). *system* es el único parámetro `type` compatible en Windows.	
system.hostname	El nombre de host del sistema. El valor se obtiene mediante las funciones `GetComputerName()` (para **netbios**), `GetComputerNameExA()` (para **fqdn**) o `gethostname()` (para **host**) en Windows. Consulte también una descripción más detallada.	
system.localtime	La hora del sistema.	
system.run	Ejecuta el comando especificado en el host.	
system.sw.arch	La información de la arquitectura del software.	
system.sw.os	La información del sistema operativo.	
system.sw.os.get	Información detallada sobre el sistema operativo (versión, tipo, nombre de la distribución, versión menor y mayor, etc.).	
system.swap.size	El tamaño del espacio de intercambio en bytes o en porcentaje del total. El parámetro de tipo `pused` es compatible en Linux con Zabbix agent, pero en Windows solo es compatible con Zabbix agent 2. Tenga en cuenta que esta clave puede informar un tamaño/porcentaje incorrecto del espacio de intercambio en plataformas Windows virtualizadas (VMware ESXi, VirtualBox). En este caso, puede usar la clave `perf_counter[\700(_Total)\702]` para obtener el porcentaje correcto del espacio de intercambio.	
system.uname	Identificación del sistema. En Windows, el valor de este item se obtiene de las clases WMI `Win32_OperatingSystem` y `Win32_Processor`. El nombre del SO (incluida la edición) puede traducirse al idioma de visualización del usuario. En algunas versiones de Windows contiene símbolos de marca registrada y espacios adicionales.	
system.uptime	El tiempo de actividad del sistema en segundos.	
vfs.dir.count	El recuento de entradas del directorio. En Windows, se omiten los enlaces simbólicos de directorio y los enlaces duros se cuentan solo una vez.	Virtual file systems
vfs.dir.get	La lista de entradas del directorio. En Windows, se omiten los enlaces simbólicos de directorio y los enlaces duros se cuentan solo una vez.	
vfs.dir.size	El tamaño del directorio. En Windows, se omite cualquier enlace simbólico y los enlaces duros se tienen en cuenta solo una vez.	

Item key	Description	Item group
vfs.file.cksum	La suma de comprobación del archivo, calculada mediante el algoritmo UNIX cksum.	
vfs.file.contents	Obtención del contenido de un archivo.	
vfs.file.exists	Comprueba si el archivo existe. En Windows, las comillas dobles deben escaparse con barra invertida '\' y toda la clave del item debe ir entre comillas dobles al usar la utilidad de línea de comandos para llamar a `zabbix_get.exe` o `agent2`. Tenga en cuenta que el item puede pasar a no compatible en Windows si se busca un directorio dentro de un directorio inexistente, por ejemplo, `vfs.file.exists[C:\no\dir,dir]` (donde 'no' no existe).	
vfs.file.get	Devuelve información sobre un archivo. Tipos de archivo compatibles en Windows: archivo regular, directorio, enlace simbólico	
vfs.file.md5sum	La suma de comprobación MD5 del archivo.	
vfs.file.owner	Recupera el propietario de un archivo.	
vfs.file.regexp	Recupera una cadena en el archivo.	
vfs.file.regmatch	Busca una cadena en el archivo.	
vfs.file.size	El tamaño del archivo.	
vfs.file.time	La información de tiempo del archivo. En Windows XP `vfs.file.time[file,change]` puede ser igual a `vfs.file.time[file,access]`.	
vfs.fs.discovery	La lista de sistemas de archivos montados con su tipo y opciones de montaje. La macro {#FSLABEL} es compatible en Windows.	
vfs.fs.get	La lista de sistemas de archivos montados con su tipo, espacio de disco disponible, estadísticas de inodos y opciones de montaje. La macro {#FSLABEL} es compatible en Windows.	
vfs.fs.size	El espacio en disco en bytes o en porcentaje del total.	
vm.memory.size	El tamaño de la memoria en bytes o en porcentaje del total.	Virtual memory
web.page.get	Obtiene el contenido de una página web.	Web monitoring
web.page.perf	El tiempo de carga de una página web completa.	
web.page.regexp	Busca una cadena en la página web.	
agent.hostmetadata	Los metadatos del host del agent.	Zabbix
agent.hostname	El nombre de host del agent.	
agent.ping	La comprobación de disponibilidad del agent.	
agent.variant	La variante de Zabbix agent (Zabbix agent o Zabbix agent 2).	
agent.version	La versión de Zabbix agent.	
zabbix.stats	Devuelve de forma remota un conjunto de métricas internas del server o proxy de Zabbix.	
zabbix.stats (queue)	Devuelve de forma remota el número de items supervisados en la cola que están retrasados en el server o proxy de Zabbix.	

Métricas específicas de Windows

La tabla proporciona detalles sobre las claves de métricas que son compatibles **solo** con el agente Zabbix para Windows.

Las métricas específicas de Windows a veces son una contraparte aproximada de una métrica de agente similar, por ejemplo, `proc_info`, compatible con Windows, corresponde aproximadamente a la métrica `proc.mem`, que no es compatible con Windows.

La clave de la métrica es un enlace a los detalles completos de la clave de la métrica.

Clave de la métrica	Descripción	Grupo de métricas
eventlog	La monitorización del registro de eventos de Windows.	Monitorización de registros
eventlog.count	El recuento de líneas en el registro de eventos de Windows.	
net.if.list	La lista de interfaces de red (incluye tipo de interfaz, estado, dirección IPv4, descripción).	Red

Clave de la métrica	Descripción	Grupo de métricas
perf_counter	El valor de cualquier contador de rendimiento de Windows.	Contadores de rendimiento
perf_counter_en	El valor de cualquier contador de rendimiento de Windows en inglés.	
perf_instance.discovery	La lista de instancias de objetos de contadores de rendimiento de Windows.	
perf_instance_en.discovery	La lista de instancias de objetos de contadores de rendimiento de Windows, descubiertas utilizando los nombres de objetos en inglés.	
proc_info	Información variada sobre procesos específicos.	Procesos
registry.data	Devuelve los datos para el nombre de valor especificado en la clave del Registro de Windows.	Registro
registry.get	La lista de valores o claves del Registro de Windows ubicados en la clave dada.	
service.discovery	La lista de servicios de Windows.	Servicios
service.info	Información sobre un servicio.	
services	El listado de servicios.	
vm.vmemory.size	El tamaño de la memoria virtual en bytes o en porcentaje del total.	Memoria virtual
wmi.get	Ejecuta una consulta WMI y devuelve el primer objeto seleccionado.	WMI
wmi.getall	Ejecuta una consulta WMI y devuelve toda la respuesta.	

Detalles de la clave del item

Los parámetros sin corchetes angulares son obligatorios. Los parámetros marcados con corchetes angulares **< >** son opcionales.

eventlog[name,<regexp>,<severity>,<source>,<eventid>,<maxlines>,<mode>]

 La supervisión del registro de eventos.
 Valor devuelto: *Log*.

Parámetros:

- **name** - el nombre del canal del registro de eventos (*Log Name* en la interfaz gráfica de Event Viewer);
- **regexp** - una expresión regular que describe el patrón requerido (distingue mayúsculas de minúsculas);
- **severity** - una expresión regular que describe la severidad (no distingue mayúsculas de minúsculas). Este parámetro acepta una expresión regular basada en los siguientes valores: ”Information”, ”Warning”, ”Error”, ”Critical”, ”Verbose” (en ejecución en Windows Vista o posterior).
- **source** - una expresión regular que describe el identificador de origen (no distingue mayúsculas de minúsculas);
- **eventid** - una expresión regular que describe el/los identificador(es) del evento (distingue mayúsculas de minúsculas);
- **maxlines** - el número máximo de nuevas líneas por segundo que el agent enviará al server o proxy de Zabbix. Este parámetro anula el valor de `MaxLinesPerSecond` en `zabbix_agentd.conf`.
- **mode** - valores posibles:
  - *all* (predeterminado);
  - *skip* - omite el procesamiento de datos más antiguos (afecta solo a los items creados recientemente).

Comentarios:

- El item debe configurarse como una comprobación activa;
- El agent no puede enviar eventos desde el registro ”Forwarded events”;
- Se admite Windows Eventing 6.0;
- Seleccionar para este item un tipo de información que no sea Log provocará la pérdida de la marca de tiempo local, así como de la severidad del registro y la información de origen;
- Consulte también información adicional sobre la supervisión de registros.

Ejemplos:

```
eventlog[Application]
eventlog[Microsoft-Windows-Application-Experience/Program-Compatibility-Assistant]
eventlog[Security,,"Failure Audit",,^(529|680)$]
eventlog[System,,"Warning|Error"]
eventlog[System,,,,^1$]
eventlog[Windows PowerShell,,,,,,skip]
eventlog[System,,,,@TWOSHORT] #aquí se hace referencia a una expresión regular personalizada llamada `TWOS
```

eventlog.count[name,<regexp>,<severity>,<source>,<eventid>,<maxproclines>,<mode>]

 El recuento de líneas en el registro de eventos de Windows.
 Valor devuelto: *Integer*.

Parámetros:

- **name** - el nombre del canal del registro de eventos (*Log Name* en la interfaz gráfica de Event Viewer);
- **regexp** - una expresión regular que describe el patrón requerido (sensible a mayúsculas y minúsculas);
- **severity** - una expresión regular que describe la severidad (no sensible a mayúsculas y minúsculas). Este parámetro acepta una expresión regular basada en los siguientes valores: ”Information”, ”Warning”, ”Error”, ”Critical”, ”Verbose” (en ejecución en Windows Vista o posterior).
- **source** - una expresión regular que describe el identificador de origen (no sensible a mayúsculas y minúsculas);
- **eventid** - una expresión regular que describe el identificador o identificadores del evento (sensible a mayúsculas y minúsculas);
- **maxproclines** - el número máximo de nuevas líneas por segundo que el agent analizará (no puede superar 10000). El valor predeterminado es 10*`MaxLinesPerSecond` en `zabbix_agentd.conf`.
- **mode** - valores posibles:
  - *all* (predeterminado);
  - *skip* - omite el procesamiento de datos más antiguos (afecta solo a los items creados recientemente).

Comentarios:

- El item debe configurarse como una active check;
- El agent no puede enviar eventos desde el registro ”Forwarded events”;
- Se admite Windows Eventing 6.0;
- Seleccionar para este item un type of information que no sea Log provocará la pérdida de la marca de tiempo local, así como de la severidad del registro y la información de origen;
- Consulte también información adicional sobre log monitoring.

Ejemplos:

```
eventlog.count[System,,"Warning|Error"]
eventlog.count[Windows PowerShell,,,,,,skip]
```

net.if.list

 La lista de interfaces de red (incluye tipo de interfaz, estado, dirección IPv4, descripción).
 Valor de retorno: *Texto*.

Comentarios:

- Se admiten nombres de interfaz multibyte;
- Las interfaces deshabilitadas no se enumeran;
- Habilitar/deshabilitar algunos componentes puede cambiar su orden en el nombre de la interfaz de Windows;
- Algunas versiones de Windows (por ejemplo, Server 2008) pueden requerir que se instalen las últimas actualizaciones para admitir caracteres no ASCII en los nombres de las interfaces.

perf_counter[counter,<interval>]

 El valor de cualquier contador de rendimiento de Windows.
 Valor devuelto: *Integer*, *float*, *string* o *text* (según la solicitud).

Parámetros:

- **counter** - la ruta al contador;
- **interval** - los últimos N segundos para almacenar el valor promedio. El `interval` debe estar entre 1 y 900 segundos (incluidos) y el valor predeterminado es 1.

Comentarios:

- Este item sigue rutas UNC, lo que puede suponer un riesgo de seguridad;
- `interval` se usa para contadores que requieren más de una muestra (como la utilización de CPU), por lo que la comprobación devuelve un valor promedio de los últimos ”interval” segundos cada vez;
- Performance Monitor puede usarse para obtener la lista de contadores disponibles.
- Véase también: Contadores de rendimiento de Windows.

perf_counter_en[counter,<interval>]

 El valor de cualquier contador de rendimiento de Windows en inglés.
 Valor devuelto: *Integer*, *float*, *string* o *text* (según la solicitud).

Parámetros:

- **counter** - la ruta al contador en inglés;
- **interval** - los últimos N segundos para almacenar el valor promedio. El `interval` debe estar entre 1 y 900 segundos (incluidos) y el valor predeterminado es 1.

Comentarios:

- Este item sigue rutas UNC, lo que puede suponer un riesgo de seguridad;

- `interval` se usa para contadores que requieren más de una muestra (como la utilización de CPU), por lo que la comprobación devuelve un valor promedio de los últimos "interval" segundos cada vez;
- Este item solo es compatible con **Windows Server 2008/Vista** y versiones posteriores;
- Puede encontrar la lista de cadenas en inglés consultando la siguiente clave del registro: `HKEY_LOCAL_MACHINE\SOFTWARE\Microsof NT\CurrentVersion\Perflib\009`.

perf_instance.discovery[object]

 La lista de instancias de objetos de los contadores de rendimiento de Windows. Se utiliza para el descubrimiento de bajo nivel.
 Valor de retorno: *objeto JSON*.

Parámetro:

- **object** - el nombre del objeto (localizado).

perf_instance_en.discovery[object]

 La lista de instancias de objetos de contadores de rendimiento de Windows, descubiertas utilizando los nombres de los objetos en inglés. Se utiliza para el descubrimiento de bajo nivel.
 Valor de retorno: *objeto JSON*.

Parámetro:

- **object** - el nombre del objeto (en inglés).

proc_info[process,<attribute>,<type>]

 Varias informaciones sobre uno o varios procesos específicos.
 Valor devuelto: *Float*.

Parámetros:

- **process** - el nombre del proceso;
- **attribute** - el atributo del proceso solicitado;
- **type** - el tipo de representación (es relevante cuando existe más de un proceso con el mismo nombre)

Comentarios:

- Se admiten los siguientes `attributes`:
  - *vmsize* (predeterminado) - tamaño de la memoria virtual del proceso en Kbytes;
  - *wkset* - tamaño del conjunto de trabajo del proceso (cantidad de memoria física utilizada por el proceso) en Kbytes;
  - *pf* - número de fallos de página;
  - *ktime* - tiempo de kernel del proceso en milisegundos;
  - *utime* - tiempo de usuario del proceso en milisegundos;
  - *io_read_b* - número de bytes leídos por el proceso durante las operaciones de E/S;
  - *io_read_op* - número de operaciones de lectura realizadas por el proceso;
  - *io_write_b* - número de bytes escritos por el proceso durante las operaciones de E/S;
  - *io_write_op* - número de operaciones de escritura realizadas por el proceso;
  - *io_other_b* - número de bytes transferidos por el proceso durante operaciones distintas de lectura y escritura;
  - *io_other_op* - número de operaciones de E/S realizadas por el proceso, distintas de lectura y escritura;
  - *gdiobj* - número de objetos GDI utilizados por el proceso;
  - *userobj* - número de objetos USER utilizados por el proceso.
- Los `types` válidos son:
  - *avg* (predeterminado) - valor promedio de todos los procesos llamados <process>;
  - *min* - valor mínimo entre todos los procesos llamados <process>;
  - *max* - valor máximo entre todos los procesos llamados <process>;
  - *sum* - suma de los valores de todos los procesos llamados <process>.
- En un sistema de 64 bits, se requiere un agent de Zabbix de 64 bits para que este item funcione correctamente.

Ejemplos:

```
proc_info[iexplore.exe,wkset,sum] #recupera la cantidad de memoria física utilizada por todos los procesos
proc_info[iexplore.exe,pf,avg] #recupera el número promedio de fallos de página para los procesos de Inter
```

registry.data[key,<value name>]

 Devuelve los datos del nombre de valor especificado en la clave del Registro de Windows.
 Valor devuelto: *Integer*, *string* o *text* (según el tipo de valor)

Parámetros:

- **key** - la clave del registro, incluida la clave raíz; se permiten abreviaturas de raíz (por ejemplo, HKLM);
- **value name** - el nombre del valor del registro en la clave (cadena vacía "" de forma predeterminada). Se devuelve el valor predeterminado si no se especifica el nombre del valor.

Comentarios:

- Abreviaturas de raíz admitidas:
  - HKCR - HKEY_CLASSES_ROOT
  - HKCC - HKEY_CURRENT_CONFIG
  - HKCU - HKEY_CURRENT_USER
  - HKCULS - HKEY_CURRENT_USER_LOCAL_SETTINGS
  - HKLM - HKEY_LOCAL_MACHINE
  - HKPD - HKEY_PERFORMANCE_DATA
  - HKPN - HKEY_PERFORMANCE_NLSTEXT
  - HKPT - HKEY_PERFORMANCE_TEXT
  - HKU - HKEY_USERS
- Las claves con espacios deben ir entre comillas dobles.

Ejemplos:

```
registry.data["HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\Windows Error Reporting"] #return the data of
registry.data["HKLM\SOFTWARE\Microsoft\Windows\Windows Error Reporting","EnableZip"] #return the data of t
```

registry.get[key,<mode>,<name regexp>]

 La lista de valores o claves del Registro de Windows ubicados en la clave indicada.
 Valor devuelto: *objeto JSON*.

Parámetros:

- **key** - la clave del registro, incluida la clave raíz; se permiten abreviaturas de la raíz (por ejemplo, HKLM) (consulte los comentarios de `registry.data[]` para ver la lista completa de abreviaturas);
- **mode** - valores posibles:
values (predeterminado) o *keys*;
- **name regexp** - descubre solo los valores cuyos nombres coincidan con la regexp (predeterminado: descubrir todos los valores). Solo se permite con *values* como *mode*.

Las claves con espacios deben ir entre comillas dobles.

Ejemplos:

```
registry.get[HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall,values,"^DisplayName|DisplayVersion$
El JSON incluirá detalles de la clave, la última subclave, el nombre del valor, el tipo de valor y los dat
registry.get[HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall,values] #devuelve los datos de todos
El JSON incluirá detalles de la clave, la última subclave, el nombre del valor, el tipo de valor y los dat
registry.get[HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall,keys] #devuelve todas las subclaves
El JSON incluirá detalles de la clave y de la última subclave.
```

service.discovery

 La lista de servicios de Windows. Se utiliza para el descubrimiento de bajo nivel.
 Valor de retorno: *objeto JSON*.

service.info[service,<param>]

 Información sobre un servicio.
 Valor devuelto: - *Integer* - con `param` como *state*, *startup*;

 Específicamente para *state*: - 0 - en ejecución - 1 - en pausa - 2 - inicio pendiente - 3 - pausa pendiente - 4 - continuación pendiente - 5 - detención pendiente - 6 - detenido - 7 - desconocido - 255 - no existe tal servicio

 Específicamente para *startup*: - 0 - automático - 1 - automático diferido - 2 - manual - 3 - deshabilitado - 4 - desconocido - 5 - inicio automático por trigger - 6 - inicio automático diferido por trigger - 7 - inicio manual por trigger - *String* - con `param` como *displayname*, *path*, *user*; - *Text* - con `param` como *description*.

Parámetros:

- **service** - un nombre de servicio real o su nombre para mostrar, tal como aparece en el complemento Servicios de MMC;
- **param** - *state* (predeterminado), *displayname*, *path*, *user*, *startup* o *description*.

Comentarios:

- Los elementos como `service.info[service,state]` y `service.info[service]` devolverán la misma información;
- Solo con `param` como *state* este item devuelve un valor para servicios inexistentes (255).

Ejemplos:

```
service.info[SNMPTRAP] - estado del servicio SNMPTRAP;
service.info[SNMP Trap] - estado del mismo servicio, pero con el nombre para mostrar especificado;
service.info[EventLog,startup] - el tipo de inicio del servicio EventLog
```

services[<type>,<state>,<exclude>]

 El listado de servicios.
 Valor de retorno: *0* - si está vacío; *Texto* - la lista de servicios separados por una nueva línea.

Parámetros:

- **type** - *all* (por defecto), *automatic*, *manual* o *disabled*;
- **state** - *all* (por defecto), *stopped*, *started*, *start_pending*, *stop_pending*, *running*, *continue_pending*, *pause_pending* o *paused*;
- **exclude** - los servicios a excluir del resultado. Los servicios excluidos deben listarse entre comillas dobles, separados por comas, sin espacios.

Ejemplos:

```
services[,started] #devuelve la lista de servicios iniciados;
services[automatic, stopped] #devuelve la lista de servicios detenidos que deberían estar en ejecución;
services[automatic, stopped, "service1,service2,service3"] #devuelve la lista de servicios detenidos que d
```

vm.vmemory.size[<type>]

 El tamaño de la memoria virtual en bytes o en porcentaje del total.
 Valor devuelto: *Integer* - para bytes; *float* - para porcentaje.

Parámetro:

- **type** - valores posibles: *available* (memoria virtual disponible), *pavailable* (memoria virtual disponible, en porcentaje), *pused* (memoria virtual utilizada, en porcentaje), *total* (memoria virtual total, por defecto), o *used* (memoria virtual utilizada)

Comentarios:

- El monitoreo de las estadísticas de memoria virtual se basa en:
  - Memoria virtual total en Windows (física total + tamaño del archivo de paginación);
  - La cantidad máxima de memoria que el agent de Zabbix puede comprometer;
  - El límite actual de memoria comprometida para el sistema o el agent de Zabbix, el que sea menor.

Ejemplo:

```
vm.vmemory.size[pavailable] #devuelve la memoria virtual disponible, en porcentaje
```

wmi.get[<namespace>,<query>]

 Ejecuta una consulta WMI y devuelve el primer objeto seleccionado.
 Valor devuelto: *Entero*, *flotante*, *cadena* o *texto* (dependiendo de la consulta).

Parámetros:

- **namespace** - el espacio de nombres WMI;
- **query** - la consulta WMI que devuelve un solo objeto.

Las consultas WMI se realizan con WQL.

Ejemplo:

```
wmi.get[root\cimv2,select status from Win32_DiskDrive where Name like '%PHYSICALDRIVE0%'] #devuelve el est
```

wmi.getall[<namespace>,<query>]

 Ejecuta una consulta WMI y devuelve la respuesta completa. Puede utilizarse para el descubrimiento de bajo nivel.
 Valor devuelto: *objeto JSON*

Parámetros:

- **namespace** - el espacio de nombres de WMI;
- **query** - la consulta WMI.

Comentarios:

- Las consultas WMI se realizan con WQL.
- El preprocesamiento de JSONPath puede utilizarse para apuntar a valores más específicos en el JSON devuelto.

Ejemplo:

```
wmi.getall[root\cimv2,select * from Win32_DiskDrive where Name like '%PHYSICALDRIVE%'] #returns status inf
```

Supervisión de servicios de Windows

Este tutorial proporciona instrucciones paso a paso para configurar la supervisión de servicios de Windows. Se asume que el server y el agent de Zabbix están configurados y en funcionamiento.

Paso 1

Obtenga el nombre del servicio.

Puede obtener el nombre del servicio yendo al complemento MMC de Servicios y abriendo las propiedades del servicio. En la pestaña *General* debería ver un campo llamado *Service name*. El valor que aparece a continuación es el nombre que utilizará al configurar un item para la supervisión. Por ejemplo, si quisiera supervisar el servicio "workstation", entonces su servicio podría ser: **lanmanworkstation**.

Paso 2

Configure un item para monitorizar el servicio.

El item `service.info[service,<param>]` recupera información sobre un servicio en particular. Dependiendo de la información que necesite, especifique la opción `param`, que acepta los siguientes valores: *displayname*, *state*, *path*, *user*, *startup* o *description*. El valor predeterminado es *state* si no se especifica `param` (`service.info[service]`).

El tipo de valor de retorno depende del `param` elegido: entero para *state* y *startup*; cadena de caracteres para *displayname*, *path* y *user*; texto para *description*.

Ejemplo:

- *Clave:* `service.info[lanmanworkstation]`
- *Tipo de información:* Numérico (sin signo)

El item `service.info[lanmanworkstation]` recuperará información sobre el estado del servicio como un valor numérico. Para asignar un valor numérico a una representación de texto en el frontend ("0" como "Running", "1" como "Paused", etc.), puede configurar un mapeo de valores en el host en el que está configurado el item. Para ello, puede enlazar la template *Windows services by Zabbix agent* o *Windows services by Zabbix agent active* al host, o configurar en el host un nuevo mapeo de valores basado en el mapeo de valores *Windows service state* configurado en las templates mencionadas.

Tenga en cuenta que ambas templates mencionadas tienen configurada una regla de descubrimiento que descubrirá los servicios automáticamente. Si no desea esto, puede desactivar la regla de descubrimiento a nivel de host una vez que la template haya sido enlazada al host.

Descubrimiento de servicios de Windows

El descubrimiento de bajo nivel proporciona una forma de crear automáticamente items, triggers y gráficos para diferentes entidades en un equipo. Zabbix puede comenzar a monitorizar automáticamente los servicios de Windows en su máquina, sin necesidad de conocer el nombre exacto de un servicio o crear items para cada servicio manualmente. Se puede utilizar un filtro para generar items, triggers y gráficos reales solo para los servicios de interés.

Supervisión de archivos de registro

Resumen

Zabbix se puede usar para la supervisión y el análisis centralizados de archivos de registro con o sin compatibilidad con rotación de registros.

Las notificaciones se pueden usar para advertir a los usuarios cuando un archivo de registro contiene determinadas cadenas o patrones de cadenas.

Para supervisar un archivo de registro, debe tener:

- Zabbix agent en ejecución en el host.
- Un item de supervisión de registros configurado.

Attention:
El límite de tamaño de un archivo de registro supervisado depende de la compatibilidad con archivos grandes.

Configuración

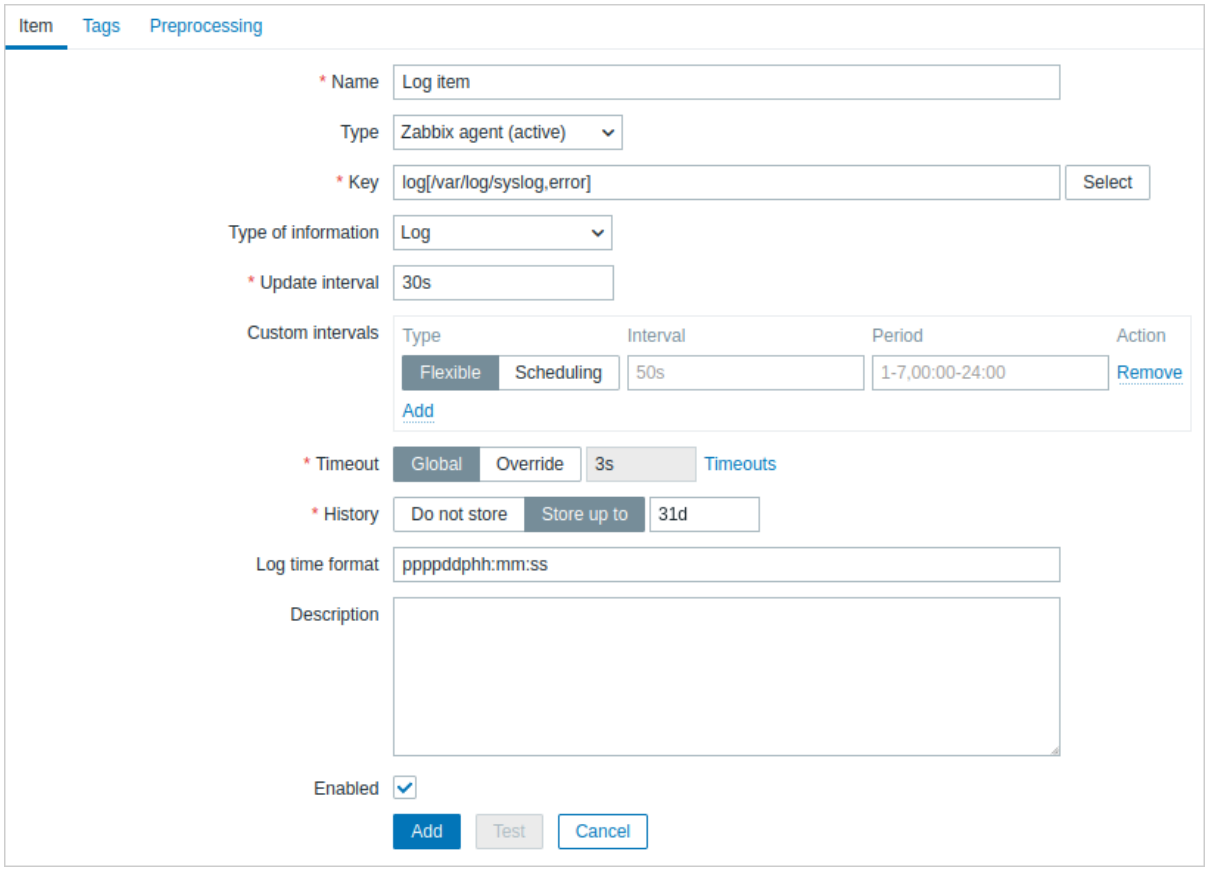
Verificar los parámetros del agent

Asegúrese de que en el archivo de configuración del agent:

- El parámetro `Hostname` coincida con el nombre del host en el frontend.
- Los servers en el parámetro `ServerActive` estén especificados para el procesamiento de comprobaciones activas.

Configuración del item

Configure un item de monitorización de logs.



Todos los campos de entrada obligatorios están marcados con un asterisco rojo.

Específicamente, para los items de monitorización de logs, introduzca:

Tipo	Seleccione **Zabbix agent (active)** aquí.
Clave	Utilice una de las siguientes claves de item: `log[]` o `logrt[]`: Estas dos claves de item permiten monitorizar logs y filtrar las entradas de log por la expresión regular del contenido, si se especifica. Por ejemplo: `log[/var/log/syslog,error]`. Asegúrese de que el archivo tenga permisos de lectura para el usuario 'zabbix'; de lo contrario, el estado del item se establecerá en 'unsupported'. **log.count[]** o **logrt.count[]**: Estas dos claves de item permiten devolver únicamente el número de líneas coincidentes. Consulte la sección de claves de items de Zabbix agent compatibles para obtener información detallada sobre el uso de estas claves de item y sus parámetros.
Tipo de información	Se completa automáticamente: Para items `log[]` o `logrt[]`: *Log*; Para items `log.count[]` o `logrt.count[]`: *Numeric (unsigned)*. Si utiliza opcionalmente el parámetro `output`, puede seleccionar manualmente un tipo de información adecuado distinto de *Log*. Tenga en cuenta que elegir un tipo de información distinto de Log provocará la pérdida de la marca de tiempo local.
Intervalo de actualización (en segundos)	El parámetro define con qué frecuencia Zabbix agent comprobará si hay cambios en el archivo de log. Establecerlo en 1 segundo garantizará que reciba los nuevos registros lo antes posible.

Formato de hora del log	En este campo puede especificar opcionalmente el patrón para analizar la marca de tiempo de la línea de log. Marcadores de posición compatibles: * **y**: *Año (1970-2038)* * **M**: *Mes (01-12)* * **d**: *Día (01-31)* * **h**: *Hora (00-23)* * **m**: *Minuto (00-59)* * **s**: *Segundo (00-59)* Si se deja vacío, la marca de tiempo se establecerá en 0 en tiempo Unix, lo que representa `January 1, 1970`. Por ejemplo, considere la siguiente línea del archivo de log de Zabbix agent: `23480:20100328:154718.045 Zabbix agent started. Zabbix 1.8.2 (revision 11211).` Comienza con seis posiciones de caracteres para el PID, seguidas de la fecha, la hora y el resto del mensaje. El formato de hora del log para esta línea sería "ppppppp:yyyyMMdd:hhmmss". Tenga en cuenta que los caracteres "p" y ":" son marcadores de posición y pueden ser cualquier carácter excepto "yMdhms".

Notas importantes

- El server y el agent mantienen el rastro del tamaño de un log monitorizado y de la hora de la última modificación (para `logrt`) en dos contadores. Además:
  - El agent también usa internamente números de inode (en UNIX/GNU/Linux), índices de archivo (en Microsoft Windows) y sumas MD5 de los primeros 512 bytes del archivo de log para mejorar las decisiones cuando los archivos de log se truncan y se rotan.
  - En sistemas UNIX/GNU/Linux se asume que los sistemas de archivos donde se almacenan los archivos de log informan números de inode, que pueden usarse para rastrear archivos.
  - En Microsoft Windows Zabbix agent determina el tipo de sistema de archivos en el que residen los archivos de log y usa:
    * En sistemas de archivos NTFS, índices de archivo de 64 bits.
    * En sistemas de archivos ReFS (solo desde Microsoft Windows Server 2012), identificadores de archivo de 128 bits.
    * En sistemas de archivos donde los índices de archivo cambian (por ejemplo, FAT32, exFAT) se usa un algoritmo alternativo para adoptar un enfoque razonable en condiciones inciertas cuando la rotación de archivos de log da como resultado varios archivos de log con la misma hora de última modificación.
  - Los números de inode, los índices de archivo y las sumas MD5 se recopilan internamente por Zabbix agent. No se transmiten a Zabbix server y se pierden cuando Zabbix agent se detiene.
  - No modifique la hora de última modificación de un archivo de log (por ejemplo, con `touch`), y no reemplace un archivo de log monitorizado copiando un archivo de vuelta a su nombre original (esto crea un nuevo inode). En cualquiera de los casos, Zabbix puede tratar el archivo como un archivo diferente y volver a leerlo desde el principio, lo que puede producir alertas duplicadas.
  - Si hay varios archivos de log coincidentes para el item `logrt[]` y Zabbix agent está siguiendo el más reciente de ellos y este archivo de log más reciente se elimina, se registra un mensaje de advertencia `"there are no files matching "<regexp mask>" in "<directory>"`. Zabbix agent ignora los archivos de log con una hora de modificación menor que la hora de modificación más reciente vista por el agent para el item `logrt[]` que se está comprobando.
- El agent comienza a leer el archivo de log desde el punto en que se detuvo la vez anterior.
- El número de bytes ya analizados (el contador de tamaño) y la hora de última modificación (el contador de tiempo) se almacenan en la base de datos de Zabbix y se envían al agent para asegurarse de que el agent comience a leer el archivo de log desde este punto en los casos en que el agent acaba de iniciarse o ha recibido items que anteriormente estaban deshabilitados o no eran compatibles. Sin embargo, si el agent ha recibido un contador de tamaño distinto de cero desde server, pero el item `logrt[]` o `logrt.count[]` no puede encontrar archivos coincidentes, el contador de tamaño se restablece a 0 para analizar desde el principio si los archivos aparecen más tarde.
- Siempre que el archivo de log se vuelva más pequeño que el contador de tamaño de log conocido por el agent, el contador se restablece a cero y el agent comienza a leer el archivo de log desde el principio teniendo en cuenta el contador de tiempo.
- Si hay varios archivos coincidentes con la misma hora de última modificación en el directorio, entonces el agent intenta analizar correctamente todos los archivos de log con la misma hora de modificación y evitar omitir datos o analizar los mismos datos dos veces, aunque no puede garantizarse en todas las situaciones. El agent no asume ningún esquema particular de rotación de archivos de log ni determina uno. Cuando se presentan varios archivos de log con la misma hora de última modificación, el agent los procesará en orden lexicográfico descendente. Así, para algunos esquemas de rotación los archivos de log se analizarán y notificarán en su orden original. Para otros esquemas de rotación no se respetará el orden original de los archivos de log, lo que puede provocar que los registros de archivos de log coincidentes se notifiquen en un

orden alterado (el problema no ocurre si los archivos de log tienen distintas horas de última modificación).

- Zabbix agent procesa nuevos registros de un archivo de log una vez cada *Update interval* segundos.
- Zabbix agent no envía más de `maxlines` de un archivo de log por segundo. El límite evita la sobrecarga de los recursos de red y CPU y anula el valor predeterminado proporcionado por el parámetro `MaxLinesPerSecond` en el archivo de configuración del agent.
- Para encontrar la cadena requerida, Zabbix procesará 10 veces más líneas nuevas que las establecidas en `MaxLinesPerSecond`. Así, por ejemplo, si un item `log[]` o `logrt[]` tiene un *Update interval* de 1 segundo, de forma predeterminada el agent analizará no más de 200 registros de archivo de log y enviará no más de 20 registros coincidentes a Zabbix server en una comprobación. Al aumentar `MaxLinesPerSecond` en el archivo de configuración del agent o establecer el parámetro `maxlines` en la clave del item, el límite puede incrementarse hasta 10000 registros de archivo de log analizados y 1000 registros coincidentes enviados a Zabbix server en una comprobación. Si el *Update interval* se establece en 2 segundos, los límites para una comprobación serían 2 veces mayores que con un *Update interval* de 1 segundo.
- Además, los valores de `log` y `log.count` siempre están limitados al 50% del tamaño del búfer de envío del agent, incluso si no hay valores que no sean de log en él. Por lo tanto, para que los valores de `maxlines` se envíen en una sola conexión (y no en varias), el parámetro `BufferSize` del agent debe ser al menos `maxlines` x 2. Zabbix agent puede cargar datos durante la recopilación de logs y así liberar el búfer, mientras que Zabbix agent 2 detendrá la recopilación de logs hasta que los datos se carguen y el búfer se libere, lo cual se realiza de forma asíncrona.
- En ausencia de items de log, todo el tamaño del búfer del agent se usa para valores que no son de log. Cuando llegan valores de log, reemplazan los valores anteriores que no son de log según sea necesario, hasta el 50% designado.
- Para registros de archivo de log de más de 256 kB, solo los primeros 256 kB se comparan con la expresión regular y el resto del registro se ignora. Sin embargo, si Zabbix agent se detiene mientras está procesando un registro largo, el estado interno del agent se pierde y el registro largo puede analizarse de nuevo y de forma diferente después de que el agent se inicie otra vez.
- Nota especial para separadores de ruta `\`: si `file_format` es `file\.log`, entonces no debe existir un directorio `file`, ya que no es posible definir de forma inequívoca si `.` está escapado o si es el primer símbolo del nombre del archivo.
- Las expresiones regulares para `logrt` solo se admiten en el nombre de archivo; no se admite la coincidencia de expresiones regulares en el directorio.
- En plataformas UNIX, un item `logrt[]` pasa a ser NOTSUPPORTED si no existe un directorio donde se espera encontrar los archivos de log.
- En Microsoft Windows, si un directorio no existe, el item no pasará a ser NOTSUPPORTED (por ejemplo, si el directorio está mal escrito en la clave del item).
- La ausencia de archivos de log para el item `logrt[]` no lo convierte en NOTSUPPORTED. Los errores de lectura de archivos de log para el item `logrt[]` se registran como advertencias en el archivo de log de Zabbix agent, pero no hacen que el item sea NOTSUPPORTED.
- El archivo de log de Zabbix agent puede ser útil para averiguar por qué un item `log[]` o `logrt[]` pasó a ser NOTSUPPORTED. Zabbix puede monitorizar el archivo de log de su agent, excepto cuando está en `DebugLevel=4` o `DebugLevel=5`.
- Buscar un signo de interrogación mediante una expresión regular, por ejemplo `\?`, puede dar falsos positivos si el archivo de texto contiene símbolos NUL, ya que Zabbix los reemplaza por `?` para continuar procesando la línea hasta el carácter de nueva línea.

Extracción de la parte coincidente de una expresión regular

A veces es posible que queramos extraer únicamente el valor de interés de un archivo de destino, en lugar de devolver toda la línea cuando se encuentra una coincidencia con una expresión regular.

Los items de registro pueden extraer los valores deseados de las líneas coincidentes. Esto se consigue mediante el parámetro adicional **output** en los items `log` y `logrt`.

El uso del parámetro `output` permite indicar el «grupo de captura» de la coincidencia que nos interesa.

Por ejemplo:

```
log[/path/to/the/file,"large result buffer allocation.*Entries: ([0-9]+)",,,,\1]
```

debería permitir devolver el número de entradas encontrado en el contenido de:

```
Fr Feb 07 2014 11:07:36.6690 */ Thread Id 1400 (GLEWF) large result
buffer allocation - /Length: 437136/Entries: 5948/Client Ver: >=10/RPC
ID: 41726453/User: AUser/Form: CFG:ServiceLevelAgreement
```

Solo se devolverá el número, porque `\1` hace referencia al primer y único grupo de captura: `([0-9]+)`.

Además, gracias a la posibilidad de extraer y devolver un número, el valor puede utilizarse para definir triggers.

Uso del parámetro maxdelay

El parámetro `maxdelay` en los items de log permite ignorar algunas líneas más antiguas de los archivos de log para que las líneas más recientes se analicen dentro de los `maxdelay` segundos.

Warning:
Especificar `maxdelay` > 0 puede provocar **que se ignoren registros importantes del archivo de log y se pierdan alertas**. Úselo con cuidado y bajo su propia responsabilidad, solo cuando sea necesario.

De forma predeterminada, los items para la supervisión de logs siguen todas las nuevas líneas que aparecen en los archivos de log. Sin embargo, hay aplicaciones que en algunas situaciones empiezan a escribir una cantidad enorme de mensajes en sus archivos de log. Por ejemplo, si una base de datos o un servidor DNS no está disponible, estas aplicaciones inundan los archivos de log con miles de mensajes de error casi idénticos hasta que se restablece el funcionamiento normal. De forma predeterminada, todos esos mensajes se analizarán diligentemente y las líneas coincidentes se enviarán al server tal como se haya configurado en los items `log` y `logrt`.

La protección integrada contra sobrecarga consiste en un parámetro configurable `maxlines` (protege al server de demasiadas líneas de log coincidentes entrantes) y un límite de 10*`maxlines` (protege la CPU y la E/S del host frente a la sobrecarga causada por el agent en una sola comprobación). Aun así, existen 2 problemas con la protección integrada. Primero, se informa al server de una gran cantidad de mensajes potencialmente poco informativos y estos consumen espacio en la base de datos. Segundo, debido al número limitado de líneas analizadas por segundo, el agent puede quedarse rezagado respecto a los registros de log más recientes durante horas. Muy probablemente, preferirá recibir antes información sobre la situación actual en los archivos de log en lugar de revisar registros antiguos durante horas.

La solución a ambos problemas es usar el parámetro `maxdelay`. Si se especifica `maxdelay` > 0, durante cada comprobación se mide el número de bytes procesados, el número de bytes restantes y el tiempo de procesamiento. A partir de estos números, el agent calcula un retraso estimado: cuántos segundos tardaría en analizar todos los registros restantes de un archivo de log.

Si el retraso no supera `maxdelay`, entonces el agent continúa analizando el archivo de log como de costumbre.

Si el retraso es mayor que `maxdelay`, entonces el agent **ignora un fragmento de un archivo de log "saltándolo"** hasta una nueva posición estimada, de modo que las líneas restantes puedan analizarse dentro de `maxdelay` segundos.

Tenga en cuenta que el agent ni siquiera lee las líneas ignoradas en el búfer, sino que calcula una posición aproximada a la que saltar dentro de un archivo.

El hecho de omitir líneas del archivo de log se registra en el archivo de log del agent de la siguiente manera:

```
14287:20160602:174344.206 item:"logrt["/home/zabbix32/test[0-9].log",ERROR,,1000,,,120.0]"
logfile:"/home/zabbix32/test1.log" skipping 679858 bytes
(from byte 75653115 to byte 76332973) to meet maxdelay
```

El número "to byte" es aproximado porque, después del "jump", el agent ajusta la posición en el archivo al comienzo de una línea de log, que puede estar más adelante en el archivo o más atrás.

Dependiendo de cómo se compare la velocidad de crecimiento con la velocidad de análisis del archivo de log, puede que no vea "jumps", "jumps" raros o frecuentes, "jumps" grandes o pequeños, o incluso un "jump" pequeño en cada comprobación. Las fluctuaciones en la carga del sistema y la latencia de red también afectan al cálculo del retraso y, por tanto, al "jump" hacia adelante para mantenerse al día con el parámetro `maxdelay`.

No se recomienda establecer `maxdelay` < `update interval` (puede dar lugar a "jumps" pequeños y frecuentes).

Notas sobre el manejo de la rotación de archivos de log `copytruncate`

`logrt` con la opción `copytruncate` asume que los distintos archivos de log tienen registros diferentes (al menos, sus marcas de tiempo son distintas), por lo tanto, las sumas MD5 de los bloques iniciales (hasta los primeros 512 bytes) serán diferentes. Dos archivos con las mismas sumas MD5 de los bloques iniciales significan que uno de ellos es el original y el otro, una copia.

`logrt` con la opción `copytruncate` hace un esfuerzo para procesar correctamente las copias de archivos de log sin informar duplicados. Sin embargo, no se recomiendan situaciones como generar varias copias del archivo de log con la misma marca de tiempo, rotar el archivo de log con más frecuencia que el intervalo de actualización del item `logrt[]`, o reiniciar el agent con frecuencia. El agent intenta gestionar razonablemente bien todas estas situaciones, pero no se pueden garantizar buenos resultados en todos los casos.

Notas sobre archivos persistentes para items log*[]

Propósito de los archivos persistentes

Cuando se inicia Zabbix agent, recibe una lista de comprobaciones activas desde Zabbix server o proxy. Para las métricas `log*[]`, recibe el tamaño del registro procesado y la hora de modificación para determinar desde dónde empezar la supervisión del archivo de registro. Según el tamaño real del archivo de registro y la hora de modificación informados por el sistema de archivos, el agent decide si continuar la supervisión del archivo de registro desde el tamaño de registro procesado o volver a analizar el archivo de registro desde el principio.

Un agent en ejecución mantiene un conjunto más amplio de atributos para hacer seguimiento de todos los archivos de registro supervisados entre comprobaciones. Este estado en memoria se pierde cuando el agent se detiene.

El nuevo parámetro opcional `persistent_dir` especifica un directorio para almacenar este estado del item `log[]`, `log.count[]`, `logrt[]` o `logrt.count[]` en un archivo. El estado del item de registro se restaura desde el archivo persistente después de reiniciar Zabbix agent.

El caso de uso principal es la supervisión de un archivo de registro ubicado en un sistema de archivos reflejado. Hasta cierto momento, el archivo de registro se escribe en ambos espejos. Luego, los espejos se separan. En la copia activa, el archivo de registro sigue creciendo y recibe nuevos registros. Zabbix agent lo analiza y envía el tamaño del registro procesado y la hora de modificación a server. En la copia pasiva, el archivo de registro permanece igual, muy por detrás de la copia activa. Más tarde, el sistema operativo y Zabbix agent se reinician desde la copia pasiva. El tamaño del registro procesado y la hora de modificación que Zabbix agent recibe desde server pueden no ser válidos para la situación en la copia pasiva. Para continuar la supervisión del archivo de registro desde el punto en que el agent se quedó en el momento de la separación del espejo del sistema de archivos, el agent restaura su estado desde el archivo persistente.

Operación del agent con archivo persistente

Al iniciarse, Zabbix agent no sabe nada acerca de los archivos persistentes. Solo después de recibir una lista de comprobaciones activas desde Zabbix server (proxy), el agent ve que algunos items de log deben respaldarse con archivos persistentes en los directorios especificados.

Durante la operación del agent, los archivos persistentes se abren para escritura (con `fopen(filename, "w")`) y se sobrescriben con los datos más recientes. La probabilidad de perder datos del archivo persistente si la sobrescritura y la división del espejo del sistema de archivos ocurren al mismo tiempo es muy pequeña; no se requiere un tratamiento especial.

La escritura en el archivo persistente NO va seguida de una sincronización forzada con el medio de almacenamiento (`fsync()` no se llama).

La sobrescritura con los datos más recientes se realiza después de informar correctamente a Zabbix server del registro del archivo de log coincidente o de los metadatos (tamaño de log procesado y hora de modificación). Eso puede ocurrir tan a menudo como en cada comprobación del item si el archivo de log sigue cambiando.

No se realizan acciones especiales durante el apagado del agent.

Después de recibir una lista de comprobaciones activas, el agent marca los archivos persistentes obsoletos para su eliminación. Un archivo persistente se vuelve obsoleto si:

1. El elemento de log correspondiente ya no se supervisa.
2. Un elemento de log se reconfigura con una ubicación `persistent_dir` diferente a la anterior.

La eliminación se realiza con un retraso de 24 horas porque los archivos de log en estado NOTSUPPORTED no se incluyen en la lista de comprobaciones activas, pero pueden pasar a SUPPORTED más adelante y sus archivos persistentes serán útiles.

Si el agent se detiene antes de que transcurran 24 horas, entonces los archivos obsoletos no se eliminarán, ya que Zabbix agent ya no recibe información sobre su ubicación desde Zabbix server.

Warning:
Reconfigurar `persistent_dir` de un elemento de log de nuevo a la ubicación antigua de `persistent_dir` mientras el agent está detenido, sin que el usuario elimine el archivo persistente antiguo, provocará la restauración del estado del agent a partir del archivo persistente antiguo, lo que dará lugar a mensajes omitidos o alertas falsas.

Nombres y ubicación de los archivos persistentes

Zabbix agent distingue las comprobaciones activas mediante sus claves. Por ejemplo, `logrt[/home/zabbix/test.log]` y `logrt[/home/zabbix/test.log,]` son items diferentes. Modificar el item `logrt[/home/zabbix/test.log,,,10]` en el frontend a `logrt[/home/zabbix/test.log,,,20]` provocará que el item `logrt[/home/zabbix/test.log,,,10]` se elimine de la lista de comprobaciones activas del agent y que se cree el item `logrt[/home/zabbix/test.log,,,20]` (algunos atributos se conservan durante la modificación en el frontend/server, pero no en el agent).

El nombre del archivo se compone de la suma MD5 de la clave del item, con la longitud de la clave del item añadida para reducir la posibilidad de colisiones. Por ejemplo, el estado del item `logrt[/home/zabbix50/test.log,,,,,,,,/home/zabbix50/agent_privat` se conservará en el archivo persistente c963ade4008054813bbc0a650bb8e09266.

Varios items de registro pueden utilizar el mismo valor de `persistent_dir`.

`persistent_dir` se especifica teniendo en cuenta las disposiciones específicas del sistema de archivos, los puntos y las opciones de montaje, así como la configuración de duplicación del almacenamiento: el archivo persistente debe encontrarse en el mismo sistema de archivos duplicado que el archivo de registro supervisado.

Si el directorio persistent_dir no se puede crear o no existe, o si el Zabbix agent no tiene permisos suficientes para crear, escribir, leer o eliminar archivos, el item de registro pasa a ser NOTSUPPORTED.

Si se eliminan los permisos de acceso a los archivos de almacenamiento persistente durante el funcionamiento del agent o se producen otros errores (por ejemplo, si el disco está lleno), los errores se registran en el archivo de registro del agent, pero el item de registro no pasa a ser NOTSUPPORTED.

Carga en I/O

El archivo persistente del item se actualiza después del envío correcto de cada lote de datos (que contiene los datos del item) al server. Por ejemplo, el valor predeterminado de BufferSize es 100. Si un item de log ha encontrado 70 registros coincidentes, entonces los primeros 50 registros se enviarán en un lote, el archivo persistente se actualizará, luego los 20 registros restantes se enviarán (tal vez con algún retraso cuando se acumule más datos) en el segundo lote, y el archivo persistente se actualizará nuevamente.

Acciones si falla la comunicación entre agent y server

Cada línea coincidente de un item log[] y logrt[] y el resultado de cada comprobación de un item log.count[] y logrt.count[] requiere un espacio libre en el área designada del 50% en el búfer de envío del agent. Los elementos del búfer se envían regularmente al server (o proxy) y los espacios del búfer vuelven a quedar libres.

Mientras haya espacios libres en el área designada de logs en el búfer de envío del agent y falle la comunicación entre agent y server (o proxy), los resultados de monitorización de logs se acumulan en el búfer de envío. Esto ayuda a mitigar fallos de comunicación breves.

Durante fallos de comunicación más prolongados, todos los espacios de logs se ocupan y se toman las siguientes acciones:

- Las comprobaciones de items log[] y logrt[] se detienen. Cuando se restablece la comunicación y hay espacios libres disponibles en el búfer, las comprobaciones se reanudan desde la posición anterior. No se pierden líneas coincidentes; simplemente se informan más tarde.
- Las comprobaciones log.count[] y logrt.count[] se detienen si maxdelay = 0 (valor predeterminado). El comportamiento es similar al de los items log[] y logrt[] descritos anteriormente. Tenga en cuenta que esto puede afectar a los resultados de log.count[] y logrt.count[]: por ejemplo, una comprobación cuenta 100 líneas coincidentes en un archivo de log, pero como no hay espacios libres en el búfer, la comprobación se detiene. Cuando se restablece la comunicación, el agent cuenta las mismas 100 líneas coincidentes y también 70 nuevas líneas coincidentes. Ahora el agent envía count = 170 como si se hubieran encontrado en una sola comprobación.
- Las comprobaciones log.count[] y logrt.count[] con maxdelay > 0: si no hubo un "salto" durante la comprobación, el comportamiento es similar al descrito anteriormente. Si se produjo un "salto" sobre líneas del archivo de log, se conserva la posición después del "salto" y se descarta el resultado contado. Así, el agent intenta mantenerse al día con un archivo de log en crecimiento incluso en caso de fallo de comunicación.

Gestión de errores de compilación y de ejecución de expresiones regulares

Si una expresión regular utilizada en un item log[], logrt[], log.count[] o logrt.count[] no puede ser compilada por la biblioteca PCRE o PCRE2, el item pasa al estado NOTSUPPORTED con un mensaje de error. Para continuar supervisando el item de registro, se debe corregir la expresión regular.

Si la expresión regular se compila correctamente, pero falla durante la ejecución (en algunos o en todos los registros), el item de registro permanece en estado compatible y la supervisión continúa. El error de ejecución se registra en el archivo de registro del Zabbix agent (sin el registro del archivo de registro).

La frecuencia de registro está limitada a un error de ejecución por comprobación para permitir que el Zabbix agent supervise su propio archivo de registro. Por ejemplo, si se analizan 10 registros y 3 fallan con un error de ejecución de la expresión regular, se genera un registro en el registro del agent.

Excepción: si MaxLinesPerSecond=1 y update interval=1 (lo que permite analizar un solo registro por comprobación), los errores de ejecución de la expresión regular no se registran.

zabbix_agentd registra la clave del item en caso de un error de ejecución; zabbix_agent2 registra el ID del item para ayudar a identificar qué item de registro presenta errores de ejecución. Se recomienda rediseñar la expresión regular en caso de errores de ejecución.

Comprobación simple

Descripción general

Las comprobaciones simples se utilizan normalmente para comprobaciones remotas sin agent de servicios.

Tenga en cuenta que no se necesita el agent de Zabbix para las comprobaciones simples. El server/proxy de Zabbix es responsable del procesamiento de las comprobaciones simples (realizando conexiones externas, etc.).

Ejemplos de uso de comprobaciones simples:

```
net.tcp.service[ftp,,155]
net.tcp.service[http]
net.tcp.service.perf[http,,8080]
net.udp.service.perf[ntp]
```

Note:
Los campos *Nombre de usuario* y *Contraseña* (limitados a 255 caracteres) en la configuración del item de comprobación simple se utilizan para los items de monitorización de VMware; se ignoran en otros casos.

Comprobaciones compatibles

Las claves de item se enumeran sin parámetros opcionales ni información adicional. Haga clic en la clave de item para ver los detalles completos.

Consulte también Claves de item de supervisión de VMware.

Clave de item	Descripción
icmpping	La accesibilidad del host mediante ping ICMP.
icmppingloss	El porcentaje de paquetes perdidos.
icmppingretry	La accesibilidad del host mediante ping ICMP con reintentos.
icmppingsec	El tiempo de respuesta del ping ICMP.
net.tcp.service	Comprueba si un servicio está en ejecución y acepta conexiones TCP.
net.tcp.service.perf	Comprueba el rendimiento de un servicio TCP.
net.udp.service	Comprueba si un servicio está en ejecución y responde a solicitudes UDP.
net.udp.service.perf	Comprueba el rendimiento de un servicio UDP.

Detalles de la clave del item

Los parámetros sin corchetes angulares son obligatorios. Los parámetros marcados con corchetes angulares **< >** son opcionales.

icmpping[<target>,<packets>,<interval>,<size>,<timeout>,<options>]

 La accesibilidad del host mediante ping ICMP.
 Valor de retorno: *0* - el ping ICMP falla; *1* - el ping ICMP tiene éxito.

Parámetros:

- **target** - la IP o el nombre DNS del host;
- **packets** - el número de paquetes;
- **interval** - el tiempo entre paquetes sucesivos en milisegundos;
- **size** - el tamaño del paquete en bytes;
- **timeout** - el tiempo de espera en milisegundos;
- **options** - se usa para permitir redirecciones: si está vacío (valor predeterminado), las respuestas redirigidas se tratan como si el host de destino estuviera caído; si se establece en *allow_redirect*, las respuestas redirigidas se tratan como si el host de destino estuviera activo.

Consulte también la tabla de valores predeterminados.

Comentario:

- Si se devuelve al menos uno de los paquetes definidos por el parámetro `packets`, el item devuelve 1. Si no se devuelve ninguno de los paquetes, el item devuelve 0.

Ejemplo:

```
icmpping[,4] #Si se devuelve al menos uno de los cuatro paquetes, el item devuelve 1.
```

icmppingloss[<target>,<packets>,<interval>,<size>,<timeout>,<options>]

 El porcentaje de paquetes perdidos.
 Valor de retorno: *Float*.

Parámetros:

- **target** - la IP o el nombre DNS del equipo;
- **packets** - el número de paquetes;
- **interval** - el tiempo entre paquetes sucesivos en milisegundos;
- **size** - el tamaño del paquete en bytes;
- **timeout** - el tiempo de espera en milisegundos;

- **options** - se utiliza para permitir redirección: si está vacío (valor por defecto), las respuestas redirigidas se tratan como equipo de destino caído; si se establece en *allow_redirect*, las respuestas redirigidas se tratan como equipo de destino activo.

Consulte también la tabla de valores por defecto.

icmppingretry[<target>,<retries>,<backoff>,<size>,<timeout>,<options>]

 La accesibilidad del host por ICMP ping con reintentos. Si el primer paquete tiene éxito, se detiene; si el paquete falla, se vuelve a intentar hasta que se alcance el número máximo de intentos definido en el parámetro `retries`. Este item es útil para reducir el número de paquetes enviados a través de la red.
 Valor de retorno: *0* - ICMP ping falla; *1* - ICMP ping exitoso.

Parámetros:

- **target** - la IP o el nombre DNS del host;
- **retries** - el número de veces que se intentará hacer ping a un destino, sin incluir el primer intento (0 o más; por defecto 1);
- **backoff** - el número por el que se multiplica el tiempo de espera en cada solicitud sucesiva (rango 1.0-5.0; por defecto 1.0);
- **size** - el tamaño del paquete en bytes;
- **timeout** - el tiempo de espera en milisegundos;
- **options** - se utiliza para permitir la redirección: si está vacío (valor por defecto), las respuestas redirigidas se tratan como host de destino caído; si se establece en *allow_redirect*, las respuestas redirigidas se tratan como host de destino activo.

Consulte también la tabla de valores por defecto.

icmppingsec[<objetivo>,<paquetes>,<intervalo>,<tamaño>,<timeout>,<modo>,<opciones>]

 El tiempo de respuesta de ping ICMP (en segundos).
 Valor de retorno: *Float*.

Parámetros:

- **objetivo**: la IP o el nombre DNS del equipo;
- **paquetes**: el número de paquetes;
- **intervalo**: el tiempo entre paquetes sucesivos en milisegundos;
- **tamaño**: el tamaño del paquete en bytes;
- **timeout**: el tiempo de espera en milisegundos;
- **modo**: valores posibles: *min*, *max* o *avg* (por defecto);
- **opciones**: se utiliza para permitir redirección: si está vacío (valor por defecto), las respuestas redirigidas se tratan como equipo de destino caído; si se establece en *allow_redirect*, las respuestas redirigidas se tratan como equipo de destino activo.

Comentarios:

- Los paquetes que se pierden o agotan el tiempo de espera no se utilizan en el cálculo;
- Si el equipo no está disponible (se alcanza el tiempo de espera), la métrica devolverá 0;
- Si el valor de retorno es menor que 0.0001 segundos, el valor se establecerá en 0.0001 segundos;
- Consulte también la tabla de valores por defecto.

net.tcp.service[service,<ip>,<port>]

 Comprueba si un servicio está en ejecución y acepta conexiones TCP.
 Valor de retorno: *0* - el servicio está caído; *1* - el servicio está en ejecución.

Parámetros:

- **service** - valores posibles: *ssh*, *ldap*, *smtp*, *ftp*, *http*, *pop*, *nntp*, *imap*, *tcp*, *https*, *telnet* (ver detalles);
- **ip** - la dirección IP o el nombre DNS (por defecto se utiliza la IP/DNS del equipo);
- **port** - el número de puerto (por defecto se utiliza el número de puerto estándar del servicio).

Comentarios:

- Tenga en cuenta que con el servicio *tcp* es obligatorio indicar el puerto;
- Estas comprobaciones pueden generar mensajes adicionales en los archivos de registro del sistema (normalmente se registran sesiones SMTP y SSH);
- Actualmente no se admite la comprobación de protocolos cifrados (como IMAP en el puerto 993 o POP en el puerto 995). Como solución alternativa, utilice `net.tcp.service[tcp,<ip>,port]` para comprobaciones como estas.

Ejemplo:

```
net.tcp.service[ftp,,45] #Esta métrica puede utilizarse para comprobar la disponibilidad de un servidor FT
```

Attention:
Si SELinux se está ejecutando en modo forzado, las comprobaciones simples TCP/UDP personalizadas pueden ser bloqueadas por la política. Para verificar y permitir la nueva conexión saliente, revise las denegaciones de auditoría: con `grep denied /var/log/audit/audit.log`

net.tcp.service.perf[service,<ip>,<port>]

 Comprueba el rendimiento de un servicio TCP.
 Valor de retorno: *Float*: *0.000000* - el servicio está caído; *segundos* - el número de segundos empleados al conectar con el servicio.

Parámetros:

- **service** - valores posibles: *ssh*, *ldap*, *smtp*, *ftp*, *http*, *pop*, *nntp*, *imap*, *tcp*, *https*, *telnet* (ver detalles);
- **ip** - la dirección IP o el nombre DNS (por defecto se utiliza la IP/DNS del equipo);
- **port** - el número de puerto (por defecto se utiliza el número de puerto estándar del servicio).

Comentarios:

- Tenga en cuenta que con el servicio *tcp* es obligatorio indicar el puerto;
- Actualmente no se admite la comprobación de protocolos cifrados (como IMAP en el puerto 993 o POP en el puerto 995). Como solución alternativa, utilice `net.tcp.service[tcp,<ip>,port]` para comprobaciones como estas.

Ejemplo:

```
net.tcp.service.perf[ssh] #Esta métrica puede utilizarse para probar la velocidad de la respuesta inicial
```

net.udp.service[service,<ip>,<puerto>]

 Comprueba si un servicio está en ejecución y responde a solicitudes UDP.
 Valor de retorno: *0* - el servicio está caído; *1* - el servicio está en ejecución.

Parámetros:

- **service** - valores posibles: *ntp* (ver detalles);
- **ip** - la dirección IP o el nombre DNS (por defecto se utiliza la IP/DNS del equipo);
- **puerto** - el número de puerto (por defecto se utiliza el número de puerto estándar del servicio).

Ejemplo:

```
net.udp.service[ntp,,45] #Esta métrica puede utilizarse para comprobar la disponibilidad del servicio NTP
```

net.udp.service.perf[service,<ip>,<port>]

 Comprueba el rendimiento de un servicio UDP.
 Valor de retorno: *Float*: *0.000000* - el servicio está caído; *segundos* - el número de segundos que se ha esperado la respuesta del servicio.

Parámetros:

- **service** - valores posibles: *ntp* (ver detalles);
- **ip** - la dirección IP o el nombre DNS (por defecto se utiliza la IP/DNS del equipo);
- **port** - el número de puerto (por defecto se utiliza el número de puerto estándar del servicio).

Ejemplo:

```
net.udp.service.perf[ntp] #Esta métrica puede utilizarse para probar el tiempo de respuesta del servicio N
```

> **Attention:**
> Para admitir SourceIP en comprobaciones simples de LDAP (por ejemplo, `net.tcp.service[ldap]`), se requiere OpenLDAP versión 2.6.1 o superior.

Procesamiento de timeout

Los timeouts flexibles de item, aunque son compatibles con las comprobaciones simples, no se aplican a los items `icmpping*` y VMware. Consulte Soporte de timeout flexible.

ICMP pings

Zabbix utiliza la utilidad externa **fping** para procesar ICMP pings (**icmpping**, **icmppingloss**, **icmppingretry**, **icmppingsec**).

Instalación

fping no está incluido con Zabbix y debe instalarse por separado:

- Varias plataformas basadas en Unix tienen el paquete fping en sus repositorios predeterminados, pero no está preinstalado. En este caso, puede utilizar el gestor de paquetes para instalar fping.
- Zabbix proporciona paquetes de fping para RHEL y sus derivados. Tenga en cuenta que estos paquetes se proporcionan sin soporte oficial.
- fping también puede compilarse desde el código fuente.

Configuración

Especifique la ubicación de fping en el parámetro *FpingLocation* del archivo de configuración de Zabbix server/proxy (o el parámetro *Fping6Location* para usar direcciones IPv6).

fping debe ser ejecutable por el usuario con el que se ejecuta Zabbix server/proxy, y este usuario debe tener permisos suficientes.

Consulte también: Problemas conocidos para el procesamiento de comprobaciones simples con versiones de fping anteriores a 3.10.

Valores predeterminados

Valores predeterminados, límites y descripción de los valores de los parámetros de comprobación ICMP:

Parámetro	Unidad	Descripción	Indicador de fping	Valores predeterminados establecidos por		Límites permitidos por Zabbix	
				fping	**Zabbix**	**mín.**	**máx.**
packets	número	Número de paquetes de solicitud enviados a un destino	-C		3	1	10000
interval	milisegundos	Tiempo de espera entre paquetes sucesivos enviados a un destino individual	-p	1000		20	ilimitado
size	bytes	Tamaño del paquete en bytes 56 bytes en x86, 68 bytes en x86_64	-b	56 o 68		24	65507
timeout	milisegundos	**fping v3.x**: tiempo de espera después del envío del último paquete, afectado por el indicador `-C` **fping v4.x**: tiempo de espera individual para cada paquete	-t	**fping v3.x**: 500 **fping v4.x** y posteriores: heredado del indicador `-p`, pero no superior a 2000		50	ilimitado
retries	número	Número de veces que se intentará hacer ping a un destino, sin incluir el primer intento	-r	3	1	0	ilimitado
backoff factor	número	Número por el que se multiplica el tiempo de espera en cada solicitud sucesiva	-B	1.5	1.0	1.0	5.0

Los valores predeterminados pueden variar ligeramente según la plataforma y la versión.

Además, Zabbix utiliza las opciones de fping `-i interval ms` (no debe confundirse con el parámetro `interval` del item mencionado en la tabla anterior, que corresponde a la opción `-p` de fping) y `-S source IP address` (o `-I` en versiones anteriores de fping). Estas opciones se detectan automáticamente ejecutando comprobaciones con distintas combinaciones de opciones. Zabbix intenta detectar el valor mínimo en milisegundos que fping permite para la opción `-i` probando tres valores: 0, 1 y 10. El primer valor que funciona se utiliza posteriormente para las comprobaciones ICMP. Este proceso lo realiza individualmente cada proceso de ICMP pinger.

Las opciones de fping detectadas automáticamente se invalidan cada hora y se vuelven a detectar en el siguiente intento de realizar una comprobación ICMP. Establezca DebugLevel>=4 para ver los detalles de este proceso en el archivo de registro del server o del proxy.

Zabbix escribe las direcciones IP que se comprobarán mediante cualquiera de las claves `icmpping*` en un archivo temporal, que posteriormente se pasa a fping. Si los items tienen distintos parámetros de clave, solo se escriben en un mismo archivo aquellos que tienen parámetros de clave idénticos. fping comprobará en paralelo todas las direcciones IP escritas en el mismo archivo, por lo que el proceso de ICMP pinger de Zabbix empleará una cantidad de tiempo fija, independientemente del número de direcciones IP del archivo.

Claves de item de monitorización de VMware

La lista de claves de métricas de monitorización de VMware se ha trasladado a la sección de monitorización de VMware.

SNMP agent

Resumen

Es posible que desee utilizar la supervisión SNMP en dispositivos como impresoras, conmutadores de red, routers o UPS que normalmente tienen SNMP habilitado y en los que sería poco práctico intentar instalar sistemas operativos completos y agentes de Zabbix.

Para poder recuperar los datos proporcionados por los agentes SNMP en estos dispositivos, Zabbix server debe estar configurado inicialmente con compatibilidad SNMP, especificando la opción `--with-net-snmp`. Se recomienda también instalar archivos MIB para garantizar que los valores de los item se muestren en el formato correcto. Sin los archivos MIB, pueden producirse problemas de formato, como mostrar valores en HEX en lugar de UTF-8 o viceversa.

Las comprobaciones SNMP se realizan únicamente sobre el protocolo UDP.

Zabbix server y los demonios proxy registran líneas similares a la siguiente si reciben una respuesta SNMP incorrecta:

```
SNMP response from host "gateway" does not contain all of the requested variable bindings
```

Aunque no cubren todos los casos problemáticos, son útiles para identificar dispositivos SNMP individuales para los que deben deshabilitarse las solicitudes combinadas.

Zabbix server/proxy reintentará hasta 5 veces los item `walk` y `get` de SNMP. El mecanismo de reintento no se aplica a los fallos de resolución DNS.

Para las comprobaciones SNMP heredadas (número OID único o cadena), Zabbix server/proxy reintentará al menos una vez después de un intento de consulta fallido: ya sea mediante el mecanismo de reintento de la biblioteca SNMP o mediante el mecanismo interno de procesamiento combinado.

Warning:
Si supervisa dispositivos SNMPv3, asegúrese de que msgAuthoritativeEngineID (también conocido como snmpEngineID o "Engine ID") nunca se comparta entre dos dispositivos. Según RFC 2571 (sección 3.1.1.1), debe ser único para cada dispositivo.

Warning:
RFC3414 requiere que los dispositivos SNMPv3 conserven sus engineBoots. Algunos dispositivos no lo hacen, lo que provoca que sus mensajes SNMP se descarten como obsoletos después de reiniciarse. En tal situación, es necesario limpiar manualmente la caché SNMP en un server/proxy (usando -R snmp_cache_reload) o reiniciar el server/proxy.

Note:
Zabbix almacena en caché las asignaciones SNMPv3 EngineID → IP y reutiliza los EngineID almacenados en caché para comprobaciones posteriores en lugar de enviar una sonda cada vez, reduciendo así el tráfico de red. Si no se puede reutilizar un EngineID, se realizará un reintento con una sonda para descubrir el nuevo EngineID.

Configuración de la monitorización SNMP

Para comenzar a monitorizar un dispositivo a través de SNMP, deben realizarse los siguientes pasos:

Paso 1

Averigüe la cadena SNMP (o OID) del item que desea supervisar.

Para obtener una lista de cadenas SNMP, use el comando `snmpwalk` (parte del software net-snmp que debería haber instalado como parte de la instalación de Zabbix) o una herramienta equivalente:

```
snmpwalk -v 2c -c public <host IP> .
```

Como 2c aquí representa la versión de SNMP, también puede sustituirlo por 1, para indicar la versión 1 de SNMP en el dispositivo.

Esto debería devolverle una lista de cadenas SNMP y su último valor. Si no es así, es posible que la "community" de SNMP sea distinta de la estándar `public`, en cuyo caso deberá averiguar cuál es.

A continuación, puede recorrer la lista hasta encontrar la cadena que desea supervisar; por ejemplo, si quisiera supervisar los bytes entrantes en su switch por el puerto 3, usaría la cadena `IF-MIB::ifHCInOctets.3` de esta línea:

```
IF-MIB::ifHCInOctets.3 = Counter64: 3409739121
```

Ahora puede usar el comando `snmpget` para averiguar el OID numérico de `IF-MIB::ifHCInOctets.3`:

```
snmpget -v 2c -c public -On <host IP> IF-MIB::ifHCInOctets.3
```

Tenga en cuenta que el último número de la cadena es el número de puerto que desea supervisar. Véase también: Índices dinámicos.

Esto debería devolverle algo como lo siguiente:

```
.1.3.6.1.2.1.31.1.1.1.6.3 = Counter64: 3472126941
```

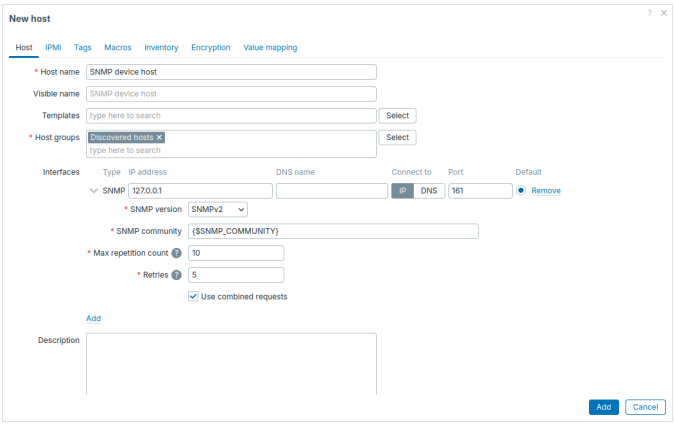
De nuevo, el último número del OID es el número de puerto.

Note:
Algunos de los OID SNMP más utilizados son traducidos automáticamente a una representación numérica por Zabbix.

En el último ejemplo anterior, el tipo de valor es "Counter64", que internamente corresponde al tipo ASN_COUNTER64. La lista completa de tipos admitidos es ASN_COUNTER, ASN_COUNTER64, ASN_UINTEGER, ASN_UNSIGNED64, ASN_INTEGER, ASN_INTEGER64, ASN_FLOAT, ASN_DOUBLE, ASN_TIMETICKS, ASN_GAUGE, ASN_IPADDRESS, ASN_OCTET_STR y ASN_OBJECT_ID. Estos tipos corresponden aproximadamente a "Counter32", "Counter64", "UInteger32", "INTEGER", "Float", "Double", "Timeticks", "Gauge32", "IpAddress", "OCTET STRING", "OBJECT IDENTIFIER" en la salida de **snmpget**, pero también pueden mostrarse como "STRING", "Hex-STRING", "OID" y otros, según la presencia de una pista de visualización.

Paso 2

Crear un host correspondiente a un dispositivo.



Añada una interfaz SNMP para el host:

1. Introduzca la dirección IP/nombre DNS y el número de puerto.
2. Seleccione la *versión SNMP* en la lista desplegable.
3. Añada las credenciales de la interfaz según la versión SNMP seleccionada:
    - SNMPv1 y v2 requieren solo la comunidad (normalmente 'public').
    - SNMPv3 requiere opciones más específicas; consulte SNMPv3.
4. Especifique el valor máximo de repeticiones (predeterminado: 10) para solicitudes bulk SNMP nativas (GetBulkRequest-PDUs); solo para items `discovery[]` y `walk[]` en SNMPv2 y v3. Tenga en cuenta que establecer este valor demasiado alto puede provocar que se agote el tiempo de espera de la comprobación del agent SNMP.
5. Marque la casilla *Usar solicitudes combinadas* para permitir el procesamiento combinado de solicitudes SNMP (no relacionado con las solicitudes bulk SNMP nativas "walk" y "get").

Puede usar una de las templates SNMP proporcionadas, que añadirá automáticamente un conjunto de items. Antes de usar una template, verifique que sea compatible con el host.

Haga clic en *Añadir* para guardar el host.

SNMPv3

Los siguientes parámetros son obligatorios para SNMPv3:

- *Context name* - Introduzca el nombre de contexto para identificar el item en la subred SNMP.
Las macros de usuario se resuelven en este campo.
- *Security name* - Introduzca el nombre de seguridad.
Las macros de usuario se resuelven en este campo.

- *Security level* - Seleccione el nivel de seguridad:
  - *noAuthNoPriv* - no se usan protocolos de autenticación ni de privacidad;
  - *AuthNoPriv* - se usa el protocolo de autenticación, pero no el de privacidad;
  - *AuthPriv* - se usan tanto el protocolo de autenticación como el de privacidad.
- *Authentication protocol* - Seleccione el protocolo de autenticación: *MD5*, *SHA1*; con net-snmp 5.8 y versiones posteriores, *SHA224*, *SHA256*, *SHA384* o *SHA512*.
- *Authentication passphrase* - Introduzca la frase de contraseña de autenticación.
Las macros de usuario se resuelven en este campo.
- *Privacy protocol* - Seleccione el protocolo de privacidad: *DES*, *AES128*, *AES192*, *AES256*, *AES192C* (Cisco) o *AES256C* (Cisco).
Consulte las notas sobre compatibilidad con el protocolo de privacidad.
- *Privacy passphrase* - Introduzca la frase de contraseña de privacidad.
Las macros de usuario se resuelven en este campo.

En caso de credenciales SNMPv3 incorrectas (security name, authentication protocol/passphrase, privacy protocol):

- Zabbix recibe un ERROR de net-snmp, excepto en el caso de una *Privacy passphrase* incorrecta, en cuyo caso Zabbix recibe un error TIMEOUT de net-snmp.
- La disponibilidad de la interfaz SNMP cambiará a rojo (no disponible).

Note:
Los cambios en *Authentication protocol*, *Authentication passphrase*, *Privacy protocol* o *Privacy passphrase*, realizados sin cambiar el *Security name*, normalmente se aplican automáticamente cuando la interfaz SNMPv3 correspondiente se actualiza en Zabbix. En los casos en que también se cambia el *Security name*, todos los parámetros se actualizarán inmediatamente.

Compatibilidad con protocolos de privacidad

Dependiendo de su sistema operativo y de la configuración de net-snmp, es posible que algunos protocolos de privacidad no estén disponibles:

- En algunos sistemas operativos más recientes (por ejemplo, RHEL9), se ha eliminado la compatibilidad con DES para el paquete `net-snmp`.
- Los protocolos de cifrado AES192 y superiores no se admiten de forma predeterminada en sistemas operativos anteriores a RHEL 8, CentOS 8, Oracle Linux 8, Debian 12, Ubuntu LTS 22.04 y openSUSE Leap 15.5.

Para comprobar si la biblioteca net-snmp admite AES192+, use una de las siguientes opciones:

1. `net-snmp-config`:

```
net-snmp-config --configure-options
```

Si la salida contiene `--enable-blumenthal-aes`, AES192+ es compatible.

Tenga en cuenta que `net-snmp-config` forma parte del paquete de desarrollo para SNMP (`libsnmp-dev` para Debian/Ubuntu, `net-snmp-devel` para CentOS/RHEL/OL/SUSE) y puede que no esté instalado de forma predeterminada.

2. `snmpget`:

```
snmpget -v 3 -x AES-256
```

Si la salida contiene `Invalid privacy protocol specified after -3x flag: AES-256`, AES192+ no es compatible. Si la salida contiene `No hostname specified.`, AES192+ no es compatible.

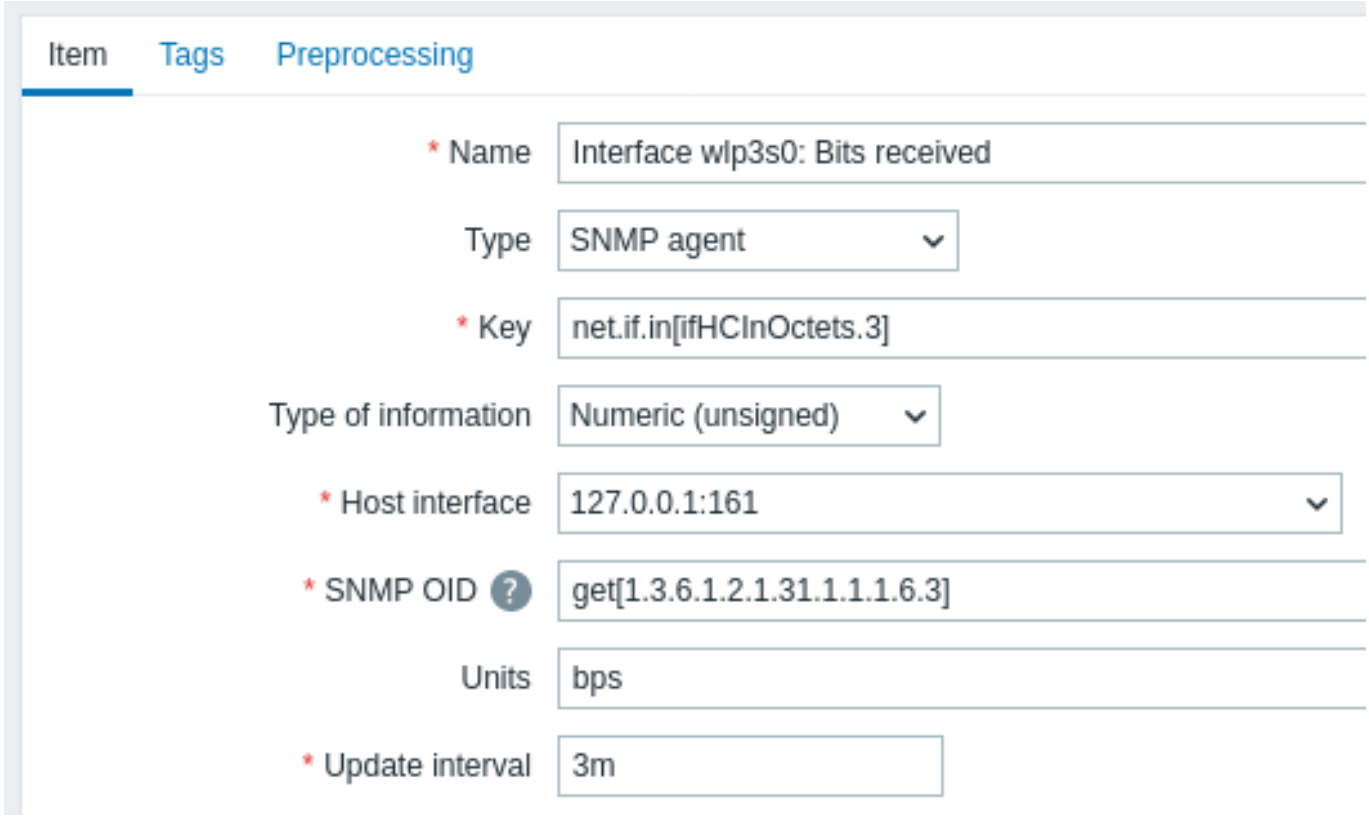
Si su biblioteca net-snmp no admite los protocolos AES192 y superiores, vuelva a compilar net-snmp con la opción `--enable-blumenthal-ae` y, a continuación, vuelva a compilar el server de Zabbix especificando la opción `--with-net-snmp=/home/user/yourcustomnetsnmp/bi`

Paso 3

Cree un item para la monitorización.

Ahora, vuelva a Zabbix y haga clic en *Items* para el host SNMP que creó anteriormente. Dependiendo de si utilizó un template o no al crear el host, tendrá una lista de items SNMP asociados a su host o simplemente una lista vacía. Supondremos que va a crear el item usted mismo utilizando la información que acaba de recopilar con snmpwalk y snmpget, así que haga clic en *Create item*.

Complete los parámetros obligatorios en el formulario del nuevo item:



Parameter	Description
Name	Introduzca el nombre del item.
Type	Seleccione **SNMP agent** aquí.
Key	Introduzca una clave significativa.
Host interface	Asegúrese de seleccionar la interfaz SNMP, por ejemplo, la de su switch/router.

Parameter	Description
SNMP OID	Utilice uno de los formatos compatibles para introducir los valores de OID: **walk[OID1,OID2,...]**: recupera un subárbol de valores. Por ejemplo: `walk[1.3.6.1.2.1.2.2.1.2,1.3.6.1.2.1.2.2.1.3]`. Esta opción utiliza solicitudes bulk nativas de SNMP (GetBulkRequest-PDU) de forma asíncrona. La configuración del tiempo de espera para este item puede establecerse en el formulario de configuración del item. Considere establecer un valor de tiempo de espera bajo para evitar demoras prolongadas si el dispositivo no está disponible, ya que se realizarán hasta 5 reintentos si los anteriores agotan el tiempo de espera o fallan (por ejemplo, un tiempo de espera de 3 segundos puede resultar en una espera de 15 segundos). Puede utilizarlo como item maestro, con items dependientes que extraigan datos del item maestro mediante el preprocesamiento. Es posible especificar varios OID en un único snmp walk, como `walk[OID1,OID2,...]`, para procesar cada OID de forma asíncrona. Si la solicitud bulk no devuelve resultados, se intentará recuperar un único registro sin una solicitud bulk. Los nombres MIB son compatibles como parámetros; por lo tanto, `walk[1.3.6.1.2.1.2.2.1.2]` y `walk[ifDescr]` devolverán el mismo resultado. Si se especifican varios OID/MIB, es decir, `walk[ifDescr,ifType,ifPhysAddress]`, el resultado será una lista concatenada. Las solicitudes GetBulk se utilizan con interfaces SNMPv2 y v3, y GetNext con interfaces SNMPv1; las repeticiones máximas para las solicitudes bulk se configuran en el nivel de la interfaz. El parámetro de repeticiones máximas afecta a las solicitudes bulk al determinar el número máximo de OID devueltos en una única respuesta bulk. Un valor más alto produce respuestas bulk más grandes, lo que reduce el número de transmisiones necesarias. Sin embargo, es posible que algunos dispositivos no admitan valores muy altos, lo que podría causar problemas. Este item devuelve el resultado de la utilidad snmpwalk con los parámetros -Oe -Ot -On. Puede utilizar este item como item maestro en el descubrimiento SNMP. **get[OID]**: recupera un valor *único* de forma asíncrona. Por ejemplo: `get[1.3.6.1.2.1.31.1.1.1.6.3]` La configuración del tiempo de espera para este item puede establecerse en el formulario de configuración del item. Considere establecer un valor de tiempo de espera bajo para evitar demoras prolongadas si el dispositivo no está disponible, ya que se realizarán hasta 5 reintentos si los anteriores agotan el tiempo de espera o fallan (por ejemplo, un tiempo de espera de 3 segundos puede resultar en una espera de 15 segundos). **OID**: (heredado) introduzca un único OID textual o numérico para recuperar un valor único de forma síncrona, opcionalmente combinado con otros valores. Por ejemplo: `1.3.6.1.2.1.31.1.1.1.6.3`. Para esta opción, el tiempo de espera de la comprobación del item será igual al valor establecido en el archivo de configuración del server. Se **recomienda** utilizar items `walk[OID]` y `get[OID]` para obtener un mejor rendimiento. Todos los items `walk[OID]` y `get[OID]` se ejecutan de forma asíncrona; no es necesario recibir la respuesta a una solicitud antes de iniciar otras comprobaciones. La resolución DNS también es asíncrona. La concurrencia máxima de las comprobaciones asíncronas es 1000 (definida por MaxConcurrentChecksPerPoller). El número de pollers SNMP asíncronos se define mediante el parámetro StartSNMPPollers. Tenga en cuenta que, para las estadísticas de tráfico de red devueltas por cualquiera de los métodos, debe añadirse un paso *Change per second* en la pestaña *Preprocessing*; de lo contrario, obtendrá el valor acumulado del dispositivo SNMP en lugar del cambio más reciente.

Todos los campos de entrada obligatorios están marcados con un asterisco rojo.

Ahora guarde el item y vaya a *Monitoring > Latest data* para consultar los datos SNMP.

Ejemplo 1

Ejemplo general:

Parámetro	Descripción
OID	1.2.3.45.6.7.8.0 (o .1.2.3.45.6.7.8.0)
Clave	<Cadena única que se utilizará como referencia para disparadores> Por ejemplo, ”mi_param”.

Tenga en cuenta que el OID puede darse en forma numérica o de cadena. Sin embargo, en algunos casos, el OID de cadena debe convertirse a representación numérica. Se puede utilizar la utilidad snmpget para este propósito:

```
snmpget -On localhost public enterprises.ucdavis.memory.memTotalSwap.0
```

Ejemplo 2

Supervisión del tiempo de actividad:

Parámetro	Descripción
OID	MIB::sysUpTime.0
Clave	router.uptime
Tipo de valor	Float
Unidades	uptime
Paso de preprocesamiento: Multiplicador personalizado	0.01

Solicitudes bulk nativas de SNMP

El item **walk[OID1,OID2,...]** permite utilizar la funcionalidad nativa de SNMP para solicitudes bulk (GetBulkRequest-PDU), disponible en las versiones 2/3 de SNMP.

Una solicitud GetBulk en SNMP ejecuta varias solicitudes GetNext y devuelve el resultado en una única respuesta. Esto puede utilizarse tanto para items SNMP normales como para el descubrimiento SNMP, con el fin de minimizar los viajes de ida y vuelta por la red.

El item SNMP **walk[OID1,OID2,...]** puede utilizarse como el item maestro que recopila los datos en una sola solicitud, junto con items dependientes que analizan la respuesta según sea necesario mediante el preprocesamiento.

Tenga en cuenta que el uso de solicitudes bulk nativas de SNMP no está relacionado con la opción de combinar solicitudes SNMP, que es la forma propia de Zabbix de combinar varias solicitudes SNMP (consulte la siguiente sección).

Se realizarán hasta cinco reintentos para los items bulk de SNMP, con el fin de evitar errores si se pierde uno de los paquetes. El tiempo de espera para los items SNMP con `get` y `walk` (establecido en el formulario de configuración del item) se aplica a toda la sesión. El tiempo de espera se aplica independientemente de si los datos se recuperan por completo; si los datos se reciben parcialmente (por ejemplo, si se recopilan correctamente solo los datos de uno de varios OID), el item pasa a ser no soportado y muestra el mensaje ”Only partial data received”. Si se alcanza el tiempo de espera, se realizará un reintento, se restablecerá el tiempo de espera y se volverá a enviar la última solicitud, lo que permitirá que la sesión continúe desde la última solicitud si se pierde un único paquete o llega demasiado tarde. Considere establecer un valor de tiempo de espera bajo para evitar retrasos prolongados si el dispositivo no está disponible, ya que se realizarán hasta 5 reintentos si los anteriores agotan el tiempo de espera o fallan (por ejemplo, un tiempo de espera de 3 segundos puede provocar una espera de 15 segundos).

Funcionamiento interno del procesamiento combinado

El server y el proxy de Zabbix pueden consultar dispositivos SNMP para obtener varios valores en una sola solicitud. Esto afecta a varios tipos de items SNMP:

- items SNMP normales
- items SNMP con índices dinámicos
- reglas de descubrimiento de bajo nivel SNMP

Todos los items SNMP de una misma interfaz con parámetros idénticos se programan para consultarse al mismo tiempo. Los dos primeros tipos de items son tomados por los pollers en lotes de un máximo de 128 items, mientras que las reglas de descubrimiento de bajo nivel se procesan individualmente, como antes.

En un nivel inferior, se realizan dos tipos de operaciones para consultar valores: obtener varios objetos especificados y recorrer un árbol de OID.

Para la operación de ”obtención”, se utiliza un GetRequest-PDU con un máximo de 128 vinculaciones de variables. Para la operación de ”recorrido”, se utiliza un GetNextRequest-PDU para SNMPv1 y un GetBulkRequest con un campo ”max-repetitions” de un máximo de 128 para SNMPv2 y SNMPv3.

Por lo tanto, los beneficios del procesamiento combinado para cada tipo de item SNMP son los siguientes:

- los items SNMP normales se benefician de las mejoras de la ”obtención”;
- los items SNMP con índices dinámicos se benefician de las mejoras tanto de la ”obtención” como del ”recorrido”: la ”obtención” se utiliza para verificar el índice y el ”recorrido” para crear la caché;
- las reglas de descubrimiento de bajo nivel SNMP se benefician de las mejoras del ”recorrido”.

Sin embargo, existe un problema técnico: no todos los dispositivos pueden devolver 128 valores por solicitud. Algunos siempre devuelven una respuesta correcta, pero otros responden con un error ”tooBig(1)” o no responden en absoluto cuando la respuesta potencial supera cierto límite.

Para encontrar el número óptimo de objetos que se deben consultar para un dispositivo determinado, Zabbix utiliza la siguiente estrategia. Comienza de forma prudente consultando 1 valor por solicitud. Si la operación tiene éxito, consulta 2 valores por solicitud. Si vuelve a tener éxito, consulta 3 valores por solicitud y continúa de forma similar multiplicando por 1,5 el número de objetos consultados, lo que da como resultado la siguiente secuencia de tamaños de solicitud: 1, 2, 3, 4, 6, 9, 13, 19, 28, 42, 63, 94, 128.

Sin embargo, cuando un dispositivo se niega a proporcionar una respuesta correcta (por ejemplo, para 42 variables), Zabbix hace dos cosas.

En primer lugar, para el lote de items actual divide por la mitad el número de objetos de una sola solicitud y consulta 21 variables. Si el dispositivo está activo, la consulta debería funcionar en la gran mayoría de los casos, porque se sabe que 28 variables funcionaban y 21 es una cantidad significativamente menor. Sin embargo, si esto también falla, Zabbix vuelve a consultar los valores uno por uno. Si en este punto sigue fallando, el dispositivo definitivamente no responde y el tamaño de la solicitud no es el problema.

Lo segundo que hace Zabbix para los lotes de items posteriores es comenzar con el último número de variables que tuvo éxito (28 en nuestro ejemplo) y continuar incrementando el tamaño de las solicitudes en 1 hasta alcanzar el límite. Por ejemplo, suponiendo que el mayor tamaño de respuesta es de 32 variables, las solicitudes posteriores tendrán tamaños de 29, 30, 31, 32 y 33. La última solicitud fallará y Zabbix no volverá a emitir una solicitud de tamaño 33. A partir de ese momento, Zabbix consultará como máximo 32 variables para este dispositivo.

Si las consultas grandes fallan con este número de variables, puede significar una de dos cosas. No se pueden conocer los criterios exactos que utiliza un dispositivo para limitar el tamaño de la respuesta, pero intentamos aproximarlos utilizando el número de variables. La primera posibilidad es que este número de variables esté cerca del límite real de tamaño de respuesta del dispositivo en el caso general: a veces la respuesta es menor que el límite y otras veces es mayor. La segunda posibilidad es que simplemente se haya perdido un paquete UDP en cualquiera de las dos direcciones. Por estos motivos, si Zabbix recibe una consulta fallida, reduce el número máximo de variables que intentará obtener para acercarse más al rango cómodo del dispositivo, pero solo hasta dos veces.

En el ejemplo anterior, si una consulta con 32 variables falla, Zabbix reducirá el número a 31. Si esa consulta también falla, Zabbix reducirá el número a 30. Sin embargo, Zabbix no reducirá el número por debajo de 30, porque asumirá que los fallos posteriores se deben a la pérdida de paquetes UDP y no al límite del dispositivo.

No obstante, si un dispositivo no puede gestionar correctamente las solicitudes combinadas por otros motivos y la heurística descrita anteriormente no funciona, existe una opción *Use combined requests* para cada interfaz que permite desactivar las solicitudes combinadas para ese dispositivo.

Si las solicitudes combinadas provocan respuestas parciales o con formato incorrecto que dan lugar a cálculos incorrectos por segundo (delta) (por ejemplo, picos aparentes en los contadores de las interfaces), desactive *Use combined requests* para la interfaz afectada a fin de forzar consultas independientes por item; esto suele evitar los picos falsos. Como alternativa, considere utilizar items `get[]` o `walk[]` asíncronos, que se ejecutan de forma asíncrona y no están sujetos al procesamiento por lotes de *Use combined requests* por interfaz; pueden utilizarse en lugar de las comprobaciones de OID síncronas heredadas para evitar problemas relacionados con las solicitudes combinadas. Busque entradas en los registros del server/proxy similares a la mostrada en la sección Descripción general para ayudar a identificar los dispositivos afectados.

Además, si la interfaz deja de estar disponible con frecuencia, puede ser necesario aumentar el parámetro `UnavailableDelay` en los archivos de configuración del server de Zabbix o del proxy de Zabbix para reducir la frecuencia de las solicitudes. Los items pueden dejar de ser compatibles si se reciben datos parciales durante el descubrimiento o los recorridos de OID.

Índices dinámicos

Descripción general

Aunque puede encontrar el número de índice requerido (por ejemplo, de una interfaz de red) entre los OID de SNMP, a veces no puede confiar por completo en que el número de índice siempre permanezca igual.

Los números de índice pueden ser dinámicos: pueden cambiar con el tiempo y, como consecuencia, su item puede dejar de funcionar.

Para evitar este escenario, es posible definir un OID que tenga en cuenta la posibilidad de que cambie un número de índice.

Por ejemplo, si necesita recuperar el valor del índice para añadirlo a **ifInOctets** que corresponde a la interfaz **GigabitEthernet0/1** en un dispositivo Cisco, utilice el siguiente OID:

```
ifInOctets["index","ifDescr","GigabitEthernet0/1"]
```

La sintaxis

Se utiliza una sintaxis especial para OID:

<OID de datos>[”index”,”<OID base del índice>”,”<cadena a buscar>”]

Parámetro	Descripción
OID de datos	OID principal que se utilizará para recuperar datos en el item.
index	Método de procesamiento. Actualmente se admite un método: **index** - buscar el índice y añadirlo al OID de datos
OID base del índice	Este OID se buscará para obtener el valor del índice correspondiente a la cadena.
cadena a buscar	La cadena que se utilizará para una coincidencia exacta con un valor al realizar la búsqueda. Distingue entre mayúsculas y minúsculas.

Ejemplo

Obtener el uso de memoria del proceso *apache*.

Si se utiliza esta sintaxis OID:

```
HOST-RESOURCES-MIB::hrSWRunPerfMem["index","HOST-RESOURCES-MIB::hrSWRunPath", "/usr/sbin/apache2"]
```

el número de índice se buscará aquí:

```
...
HOST-RESOURCES-MIB::hrSWRunPath.5376 = STRING: "/sbin/getty"
HOST-RESOURCES-MIB::hrSWRunPath.5377 = STRING: "/sbin/getty"
HOST-RESOURCES-MIB::hrSWRunPath.5388 = STRING: "/usr/sbin/apache2"
HOST-RESOURCES-MIB::hrSWRunPath.5389 = STRING: "/sbin/sshd"
...
```

Ahora tenemos el índice, 5388. El índice se añadirá al OID de datos para recibir el valor que nos interesa:

```
HOST-RESOURCES-MIB::hrSWRunPerfMem.5388 = INTEGER: 31468 KBytes
```

Almacenamiento en caché de búsqueda de índices

Cuando se solicita un item de índice dinámico, Zabbix recupera y almacena en caché la tabla SNMP completa bajo el OID base del índice, incluso si se encontrara una coincidencia antes. Esto se hace por si otro item hiciera referencia al mismo OID base más adelante: Zabbix buscaría el índice en la caché, en lugar de consultar de nuevo al host monitorizado. Tenga en cuenta que cada proceso poller utiliza una caché independiente.

En todas las operaciones posteriores de recuperación de valores, solo se verifica el índice encontrado. Si no ha cambiado, se solicita el valor. Si ha cambiado, la caché se reconstruye: cada poller que encuentre un índice modificado recorre de nuevo la tabla SNMP del índice.

OIDs especiales

Algunos de los OIDs SNMP más utilizados se traducen automáticamente a una representación numérica por Zabbix. Por ejemplo, **ifIndex** se traduce a **1.3.6.1.2.1.2.2.1.1**, **ifIndex.0** se traduce a **1.3.6.1.2.1.2.2.1.1.0**.

La tabla contiene la lista de los OIDs especiales.

Special OID	Identifier	Description
ifIndex	1.3.6.1.2.1.2.2.1.1	Un valor único para cada interfaz.
ifDescr	1.3.6.1.2.1.2.2.1.2	Una cadena de texto que contiene información sobre la interfaz. Esta cadena debe incluir el nombre del fabricante, el nombre del producto y la versión de la interfaz de hardware.

Special OID	Identifier	Description
ifType	1.3.6.1.2.1.2.2.1.3	El tipo de interfaz, distinguido según el/los protocolo(s) físico(s)/de enlace inmediatamente 'debajo' de la capa de red en la pila de protocolos.
ifMtu	1.3.6.1.2.1.2.2.1.4	El tamaño del datagrama más grande que puede enviarse/recibirse en la interfaz, especificado en octetos.
ifSpeed	1.3.6.1.2.1.2.2.1.5	Una estimación del ancho de banda actual de la interfaz en bits por segundo.
ifPhysAddress	1.3.6.1.2.1.2.2.1.6	La dirección de la interfaz en la capa de protocolo inmediatamente 'debajo' de la capa de red en la pila de protocolos.
ifAdminStatus	1.3.6.1.2.1.2.2.1.7	El estado administrativo actual de la interfaz.
ifOperStatus	1.3.6.1.2.1.2.2.1.8	El estado operativo actual de la interfaz.
ifInOctets	1.3.6.1.2.1.2.2.1.10	El número total de octetos recibidos en la interfaz, incluidos los caracteres de encapsulación.
ifInUcastPkts	1.3.6.1.2.1.2.2.1.11	El número de paquetes unicast de subred entregados a un protocolo de capa superior.
ifInNUcastPkts	1.3.6.1.2.1.2.2.1.12	El número de paquetes no unicast (es decir, broadcast o multicast de subred) entregados a un protocolo de capa superior.
ifInDiscards	1.3.6.1.2.1.2.2.1.13	El número de paquetes entrantes que se eligieron para descartar, aunque no se detectaron errores, para evitar que pudieran entregarse a un protocolo de capa superior. Una posible razón para descartar un paquete de este tipo podría ser liberar espacio en el búfer.
ifInErrors	1.3.6.1.2.1.2.2.1.14	El número de paquetes entrantes que contenían errores que impedían que pudieran entregarse a un protocolo de capa superior.
ifInUnknownProtos	1.3.6.1.2.1.2.2.1.15	El número de paquetes recibidos a través de la interfaz que se descartaron debido a un protocolo desconocido o no compatible.
ifOutOctets	1.3.6.1.2.1.2.2.1.16	El número total de octetos transmitidos fuera de la interfaz, incluidos los caracteres de encapsulación.
ifOutUcastPkts	1.3.6.1.2.1.2.2.1.17	El número total de paquetes que los protocolos de nivel superior solicitaron transmitir y que no estaban dirigidos a una dirección multicast o broadcast en esta subcapa, incluidos los que se descartaron o no se enviaron.
ifOutNUcastPkts	1.3.6.1.2.1.2.2.1.18	El número total de paquetes que los protocolos de nivel superior solicitaron transmitir y que estaban dirigidos a una dirección multicast o broadcast en esta subcapa, incluidos los que se descartaron o no se enviaron.
ifOutDiscards	1.3.6.1.2.1.2.2.1.19	El número de paquetes salientes que se eligieron para descartar, aunque no se detectaron errores, para evitar que fueran transmitidos. Una posible razón para descartar un paquete de este tipo podría ser liberar espacio en el búfer.
ifOutErrors	1.3.6.1.2.1.2.2.1.20	El número de paquetes salientes que no pudieron transmitirse debido a errores.
ifOutQLen	1.3.6.1.2.1.2.2.1.21	La longitud de la cola de paquetes de salida (en paquetes).

Archivos MIB

Introducción

MIB significa Base de Información de Gestión. Los archivos MIB permiten utilizar representaciones textuales de los OID (identificadores de objetos). Es posible utilizar OID sin formato al monitorizar dispositivos SNMP con Zabbix, pero si se siente más cómodo utilizando una representación textual, debe instalar los archivos MIB.

Por ejemplo,

`ifHCOutOctets`

es la representación textual del OID

`1.3.6.1.2.1.31.1.1.1.10`

Instalación de archivos MIB

En sistemas basados en Debian:

```
apt install snmp-mibs-downloader
download-mibs
```

En sistemas basados en RedHat:

```
dnf install net-snmp-libs
```

Habilitar archivos MIB

En sistemas basados en RedHat, los archivos MIB deberían estar habilitados por defecto. En sistemas basados en Debian, debe editar el archivo `/etc/snmp/snmp.conf` y comentar la línea que dice `mibs :`

```
# Como los paquetes snmp vienen sin archivos MIB por razones de licencia, la carga
# de MIBs está deshabilitada por defecto. Si ha añadido los MIBs puede volver a habilitar
# su carga comentando la siguiente línea.
mibs :
```

Probar archivos MIB

Las pruebas de MIB SNMP se pueden realizar utilizando la utilidad `snmpwalk`. Si no la tiene instalada, utilice las siguientes instrucciones.

En sistemas basados en Debian:

```
apt install snmp
```

En sistemas basados en RedHat:

```
dnf install net-snmp-utils
```

Después de eso, el siguiente comando no debe dar error al consultar un dispositivo de red:

```
$ snmpwalk -v 2c -c public <IP DEL DISPOSITIVO DE RED> ifInOctets
IF-MIB::ifInOctets.1 = Counter32: 176137634
IF-MIB::ifInOctets.2 = Counter32: 0
IF-MIB::ifInOctets.3 = Counter32: 240375057
IF-MIB::ifInOctets.4 = Counter32: 220893420
[...]
```

Uso de MIBs en Zabbix

Lo más importante a tener en cuenta es que los procesos de Zabbix no se informan de los cambios realizados en los archivos MIB. Por lo tanto, después de cada cambio, debe reiniciar el servidor o proxy de Zabbix, por ejemplo:

```
systemctl restart zabbix-server
```

Después de eso, los cambios realizados en los archivos MIB entrarán en vigor.

Uso de archivos MIB personalizados

Existen archivos MIB estándar que vienen con cada distribución GNU/Linux. Pero algunos proveedores de dispositivos proporcionan los suyos propios.

Supongamos que desea utilizar el archivo MIB CISCO-SMI. Las siguientes instrucciones lo descargarán e instalarán:

```
wget ftp://ftp.cisco.com/pub/mibs/v2/CISCO-SMI.my -P /tmp
mkdir -p /usr/local/share/snmp/mibs
grep -q '^mibdirs +/usr/local/share/snmp/mibs' /etc/snmp/snmp.conf 2>/dev/null || echo "mibdirs +/usr/loca
cp /tmp/CISCO-SMI.my /usr/local/share/snmp/mibs
```

Ahora debería poder utilizarlo. Intente traducir el nombre del objeto *ciscoProducts* del archivo MIB a OID:

```
snmptranslate -IR -On CISCO-SMI::ciscoProducts
.1.3.6.1.4.1.9.1
```

Si recibe errores en lugar del OID, asegúrese de que todos los comandos anteriores no hayan devuelto ningún error.

La traducción del nombre del objeto funcionó, está listo para usar el archivo MIB personalizado. Tenga en cuenta el prefijo del nombre MIB (*CISCO-SMI::*) utilizado en la consulta. Necesitará esto al usar herramientas de línea de comandos, así como Zabbix.

No olvide reiniciar el servidor/proxy de Zabbix antes de usar este archivo MIB en Zabbix.

> **Attention:**
> Tenga en cuenta que los archivos MIB pueden tener dependencias. Es decir, un MIB puede requerir otro. Para satisfacer estas dependencias, debe instalar todos los archivos MIB afectados.

SNMP trap

Resumen

Recibir traps SNMP es lo opuesto a consultar dispositivos habilitados para SNMP.

En este caso, la información se envía desde un dispositivo habilitado para SNMP a `snmptrapd` y Zabbix server o Zabbix proxy la recopila o la "captura" desde un archivo.

Por lo general, los traps se envían cuando cambia alguna condición y el agent se conecta al server en el puerto 162 (a diferencia del puerto 161 del lado del agent, que se usa para consultas). El uso de traps puede detectar algunos problemas breves que ocurren entre intervalos de consulta y que podrían pasar desapercibidos en los datos obtenidos por consulta.

La recepción de traps SNMP en Zabbix está diseñada para funcionar con `snmptrapd` y uno de los mecanismos para pasar los traps a Zabbix: un script Bash o Perl, o SNMPTT.

> **Note:**
> La forma más sencilla de configurar la supervisión de traps después de configurar Zabbix es usar la solución con script Bash, porque Perl y SNMPTT suelen faltar en las distribuciones modernas y requieren una configuración más compleja. Sin embargo, esta solución usa un script configurado como `traphandle`. Para un mejor rendimiento en sistemas de producción, use la solución con Perl integrado (ya sea un script con la opción `do perl` o SNMPTT).

El flujo de trabajo para recibir un trap es el siguiente:

1. `snmptrapd` recibe un trap.
2. `snmptrapd` pasa el trap al script receptor (Bash, Perl) o a SNMPTT.
3. El receptor analiza, da formato y escribe el trap en un archivo.
4. El SNMP trapper de Zabbix lee y analiza el archivo de traps.
5. Para cada trap, Zabbix encuentra todos los items de SNMP trapper con interfaces de host que coincidan con la dirección del trap recibido. Tenga en cuenta que durante la coincidencia solo se usa el *IP* o *DNS* seleccionado en la interfaz del host.
6. Para cada item encontrado, el trap se compara con la regexp en `snmptrap[regexp]`. El trap se establece como valor de **todos** los items coincidentes. Si no se encuentra ningún item coincidente y existe un item `snmptrap.fallback`, el trap se establece como valor de ese item.
7. Si el trap no se estableció como valor de ningún item, Zabbix registra de forma predeterminada el trap no coincidente. (Esto se configura mediante *Registrar traps SNMP no coincidentes* en *Administración* > *General* > *Otros*.)

Notas sobre la conmutación por error en HA

Durante el cambio de nodo en alta disponibilidad (HA), Zabbix continuará procesando después del último registro dentro de la última marca de tiempo ISO 8601; si no se encuentra el mismo registro, entonces solo se utilizará la marca de tiempo para identificar la última posición.

Configuración de traps SNMP

Este tipo de item requiere la siguiente configuración del frontend:

Paso 1: Crear una interfaz SNMP para el host

1. En *Recopilación de datos* > *Hosts*, cree un host o edite uno existente.
2. En el campo *Interfaces*, seleccione el tipo de interfaz *SNMP*.
3. Introduzca la dirección IP/nombre DNS y el número de puerto.
4. Seleccione la *versión SNMP* en la lista desplegable: (*SNMPv1*, *SNMPv2*, o *SNMPv3*) y añada las credenciales de la interfaz según la versión SNMP seleccionada.

La dirección de cada trap recibido se comparará con las direcciones IP y DNS de todas las interfaces SNMP para encontrar los hosts correspondientes.

Paso 2: Configurar el item

1. Para el host, cree un item o edite uno existente.
2. En el campo *Key*, use una de las claves de trap SNMP:
   - `snmptrap[regexp]` - captura todos los traps SNMP que coincidan con la expresión regular especificada en el parámetro `regexp`; si `regexp` no se especifica, captura cualquier trap.
 Se admiten macros de usuario y expresiones regulares globales en el parámetro.
 Valor devuelto: trap SNMP.
 Este item solo se puede configurar para interfaces SNMP.
   - `snmptrap.fallback` - captura todos los traps SNMP que no hayan sido capturados por ninguno de los items `snmptrap[]` de esa interfaz.
 Valor devuelto: trap SNMP.
 Este item solo se puede configurar para interfaces SNMP.

Note:
Actualmente no se admite la coincidencia de expresiones regulares multilínea.

3. Establezca *Tipo de información* en *Log* para que se analicen las marcas de tiempo. También se aceptan otros formatos, como *Numérico*, pero podrían requerir un controlador de traps personalizado.

Configuración de la monitorización de traps SNMP

Configuración de Zabbix server/proxy

Para leer las traps, Zabbix server o proxy debe configurarse para iniciar el proceso SNMP trapper y apuntar al archivo de traps que está siendo escrito por SNMPTT o por un receptor de traps Bash/Perl. Para ello, edite el archivo de configuración (`zabbix_server.conf` o `zabbix_proxy.conf`):

```
StartSNMPTrapper=1
SNMPTrapperFile=[TRAP FILE]
```

Warning:
Si se utiliza el parámetro de systemd `PrivateTmp`, es poco probable que este archivo funcione en `/tmp`.

Configuración del receptor de traps Bash

Requisitos: solo snmptrapd.

Se puede usar un script receptor de traps Bash para pasar traps a Zabbix server desde snmptrapd usando el archivo trapper. Para configurarlo, añada la opción `traphandle` al archivo de configuración de snmptrapd (`snmptrapd.conf`), consulte el ejemplo.

Note:
Es posible que sea necesario reiniciar snmptrapd para que tome los cambios de su configuración.

Configuración del receptor de traps de Perl

Requisitos: Perl, Net-SNMP compilado con `--enable-embedded-perl` (habilitado de forma predeterminada desde Net-SNMP 5.4)

Se puede usar un receptor de traps de Perl (busque `misc/snmptrap/zabbix_trap_receiver.pl`) para pasar traps a Zabbix server directamente desde snmptrapd. Para configurarlo:

- Añada el script de Perl al archivo de configuración de snmptrapd (`snmptrapd.conf`), por ejemplo:

```
perl do "[FULL PATH TO PERL RECEIVER SCRIPT]";
```

- Configure el receptor, por ejemplo:

```
$SNMPTrapperFile = '[TRAP FILE]';
$DateTimeFormat = '[DATE TIME FORMAT]';
```

Note:
Es posible que sea necesario reiniciar snmptrapd para que aplique los cambios en su configuración.

Note:
Si el nombre del script no está entre comillas, snmptrapd se negará a iniciarse y mostrará mensajes similares a estos:


```
Regexp modifiers "/l" and "/a" are mutually exclusive at (eval 2) line 1, at end of line
Regexp modifier "/l" may not appear twice at (eval 2) line 1, at end of line
```

Configuración de SNMPTT

En primer lugar, `snmptrapd` debe configurarse para usar SNMPTT.

Note:
Para obtener el mejor rendimiento, SNMPTT debe configurarse como un daemon usando `snmptthandler-embedded` para pasarle los traps. Consulte las instrucciones para configurar SNMPTT.

Cuando SNMPTT esté configurado para recibir los traps, configure `snmptt.ini`:

1. Habilite el uso del módulo Perl del paquete `NET-SNMP`:

```
net_snmp_perl_enable = 1
```

2. Registre los traps en el archivo de traps que será leído por Zabbix:

```
log_enable = 1
log_file = [TRAP FILE]
```

3. Establezca el formato de fecha y hora:

```
date_time_format = %Y-%m-%dT%H:%M:%S%z
```

> **Warning:**
> El paquete `net-snmp-perl` se eliminó en RHEL 8.0-8.2; se volvió a agregar en RHEL 8.3. Para obtener más información, consulte los problemas conocidos.

Ahora formatee los traps para que Zabbix los reconozca (edite `snmptt.conf`):

1. Cada instrucción `FORMAT` debe comenzar con `ZBXTRAP [address]`, donde `[address]` se comparará con las direcciones IP y DNS de las interfaces SNMP en Zabbix. Por ejemplo:

```
EVENT coldStart .1.3.6.1.6.3.1.1.5.1 "Status Events" Normal
FORMAT ZBXTRAP $aA Device reinitialized (coldStart)
```

2. Consulte más información sobre el formato de traps SNMP a continuación.

> **Attention:**
> No use traps desconocidos: Zabbix no podrá reconocerlos. Los traps desconocidos se pueden gestionar definiendo un evento general en `snmptt.conf`:

> `EVENT general .* "General event" Normal`

Formato de trap SNMP

Todos los receptores de traps Perl personalizados y la configuración de traps de SNMPTT deben formatear el trap de la siguiente manera:

```
[timestamp] [the trap, part 1] ZBXTRAP [address] [the trap, part 2]
```

donde

- `[timestamp]` - la marca de tiempo en formato `%Y-%m-%dT%H:%M:%S%z`.
- `ZBXTRAP` - encabezado que indica que un nuevo trap comienza en esta línea.
- `[address]` - dirección IP utilizada para encontrar el host de este trap.

Tenga en cuenta que `ZBXTRAP` y `[address]` se eliminarán del mensaje durante el procesamiento. Si el trap tiene otro formato, Zabbix podría analizar los traps de forma inesperada.

Ejemplo de trap:

```
2024-01-11T15:28:47+0200 .1.3.6.1.6.3.1.1.5.3 Normal "Status Events" localhost - ZBXTRAP 192.168.1.1 Link
```

Esto dará como resultado el siguiente trap para la interfaz SNMP con IP=192.168.1.1:

```
2024-01-11T15:28:47+0200 .1.3.6.1.6.3.1.1.5.3 Normal "Status Events"
localhost - Link down on interface 2. Admin state: 1. Operational state: 2
```

Requisitos del sistema

> **Note:**
> Se recomienda instalar archivos MIB para garantizar que los valores de las métricas se muestren en el formato correcto. Sin los archivos MIB, pueden producirse problemas de formato, como mostrar valores en HEX en lugar de UTF-8 o viceversa.

Compatibilidad con archivos grandes

Zabbix admite archivos grandes para archivos SNMP trapper. El tamaño máximo de archivo que Zabbix puede leer es 2^63 (8 EiB). Tenga en cuenta que el sistema de archivos puede imponer un límite inferior al tamaño del archivo.

Rotación de logs

Zabbix no proporciona ningún sistema de rotación de logs; eso debe ser gestionado por el usuario. La rotación de logs primero debe renombrar el archivo antiguo y solo después eliminarlo, para que no se pierdan traps:

1. Zabbix abre el archivo de traps en la última ubicación conocida y pasa al paso 3.
2. Zabbix comprueba si el archivo actualmente abierto ha sido rotado comparando el número de inode con el número de inode del archivo de traps definido. Si no hay ningún archivo abierto, Zabbix restablece la última ubicación y pasa al paso 1.
3. Zabbix lee los datos del archivo actualmente abierto y establece la nueva ubicación.
4. Los nuevos datos se analizan. Si este era el archivo rotado, el archivo se cierra y vuelve al paso 2.
5. Si no había nuevos datos, Zabbix espera 1 segundo y vuelve al paso 2.

Sistema de archivos

Debido a la implementación del archivo trap, Zabbix necesita que el sistema de archivos admita inodos para diferenciar archivos (la información se obtiene mediante una llamada `stat()`).

Ejemplos de configuración usando diferentes versiones del protocolo SNMP

Este ejemplo utiliza snmptrapd y un script receptor de Bash para pasar traps al server de Zabbix.

Configuración:

1. Configure Zabbix para iniciar el trapper SNMP y establezca el archivo de traps. Añada lo siguiente a `zabbix_server.conf`:

```
StartSNMPTrapper=1
SNMPTrapperFile=/var/lib/zabbix/snmptraps/snmptraps.log
```

2. Descargue el script de Bash en `/usr/sbin/zabbix_trap_handler.sh`:

```
curl -o /usr/sbin/zabbix_trap_handler.sh https://raw.githubusercontent.com/zabbix/zabbix-docker/trunk/temp
```

Si es necesario, ajuste la variable `ZABBIX_TRAPS_FILE` en el script. Para utilizar el valor predeterminado, cree primero el directorio principal:

```
mkdir -p /var/lib/zabbix/snmptraps
```

3. Añada lo siguiente a `snmtrapd.conf` (consulte el ejemplo funcional):

```
traphandle default /bin/bash /usr/sbin/zabbix_trap_handler.sh
```

> **Note:**
> Es posible que sea necesario reiniciar snmptrapd para que recoja los cambios realizados en su configuración.

4. Cree un item SNMP TEST (tenga en cuenta los requisitos de configuración iniciales):

   Tipo: *SNMP trap*
 Tipo de información: *Log*
 Interfaz del host: *SNMP 127.0.0.1*
 Clave: `snmptrap["linkUp"]`
 Formato de hora del log: `yyyyMMdd.hhmmss`

Tenga en cuenta que se utiliza el formato de fecha y hora ISO 8601.

5. A continuación, configuraremos `snmptrapd` para la versión del protocolo SNMP elegida y enviaremos traps de prueba mediante la utilidad `snmptrap`.

SNMPv1, SNMPv2

Los protocolos SNMPv1 y SNMPv2 utilizan la autenticación mediante una «cadena de comunidad». En el ejemplo siguiente usaremos «secret» como cadena de comunidad. Debe establecerse el mismo valor en los emisores de traps SNMP.

Tenga en cuenta que, aunque SNMPv2 todavía se utiliza ampliamente en entornos de producción, no ofrece cifrado ni autenticación real del emisor. Los datos se envían en texto plano y, por lo tanto, estas versiones del protocolo solo deben utilizarse en entornos seguros, como una red privada, y nunca deben utilizarse a través de una red pública o de terceros.

La versión 1 de SNMP ya no se utiliza realmente, puesto que no admite contadores de 64 bits y se considera un protocolo heredado.

Para habilitar la aceptación de traps SNMPv1 o SNMPv2, añada la siguiente línea a `snmptrapd.conf`. Reemplace `secret` por la cadena de comunidad SNMP configurada en los emisores de traps SNMP:

```
authCommunity log,execute,net secret
```

A continuación, podemos enviar un trap de prueba mediante `snmptrap`. En este ejemplo utilizaremos el OID común «link up»:

```
snmptrap -v 2c -c secret localhost '' linkUp.0
```

SNMPv3

SNMPv3 aborda los problemas de seguridad de SNMPv1/v2 y proporciona autenticación y cifrado. Puede utilizar los métodos de autenticación MD5 o varios métodos SHA, así como DES o varias variantes de AES como cifrado.

Configure `snmptrapd` para aceptar traps SNMPv3 añadiendo las siguientes líneas a `snmptrapd.conf`:

```
##### Cree un usuario USM SNMPv3 con autenticación y privacidad
createUser -e <engineID> <securityName> SHA <authKey> AES <privKey>
##### Permita al usuario registrar traps y ejecutar scripts de gestión de traps
authUser log,execute <securityName>
```

Por ejemplo:

```
createUser -e 0x8000000001020304 traptest SHA myAuthPassword AES myPrivPassword
authUser log,execute traptest
```

Attention:
Tenga en cuenta la palabra clave «execute», que permite ejecutar scripts para este modelo de seguridad de usuario.

Para probar la configuración de traps SNMPv3, envíe un trap de prueba mediante el siguiente comando:

```
snmptrap -v 3 -n "" -a SHA -A myAuthPassword -x AES -X myPrivPassword -l authPriv -u traptest -e 0x8000000
```

Warning:
Si desea utilizar métodos de cifrado seguros, como AES192 o AES256, utilice net-snmp a partir de la versión 5.8. Es posible que deba volver a compilarlo con la opción de `configure: --enable-blumenthal-aes`. Las versiones anteriores de net-snmp no admiten AES192/AES256. Consulte también: Strong Authentication or Encryption.

Verificación

En ambos ejemplos verá líneas similares en `/var/lib/zabbix/snmptraps/snmptraps.log`:

```
2024-01-30T10:04:23+0200 ZBXTRAP 127.0.0.1
UDP: [127.0.0.1]:56585->[127.0.0.1]:162
DISMAN-EVENT-MIB::sysUpTimeInstance = 2538834
SNMPv2-MIB::snmpTrapOID.0 = IF-MIB::linkUp.0
```

El valor del item en Zabbix será:

```
2024-01-30 10:04:23 2024-01-30 10:04:21

2024-01-30T10:04:21+0200 UDP: [127.0.0.1]:56585->[127.0.0.1]:162
DISMAN-EVENT-MIB::sysUpTimeInstance = 2538834
SNMPv2-MIB::snmpTrapOID.0 = IF-MIB::linkUp.0
```

Ejemplo con Perl:

```
2024-01-30T11:42:54+0200 ZBXTRAP 127.0.0.1
PDU INFO:
  receivedfrom                   UDP: [127.0.0.1]:58649->[127.0.0.1]:162
  notificationtype               TRAP
  version                        1
  community                      public
  errorstatus                    0
  transactionid                  1
  requestid                      2101882550
  messageid                      0
  errorindex                     0
VARBINDS:
  DISMAN-EVENT-MIB::sysUpTimeInstance type=67 value=Timeticks: (457671) 1:16:16.71
  SNMPv2-MIB::snmpTrapOID.0       type=6  value=OID: IF-MIB::linkUp.0
```

Véase también

- Artículo del blog de Zabbix sobre traps SNMP
- Configuración de snmptrapd (documentación oficial de net-snmp)
- Configuración de snmptrapd para recibir notificaciones SNMPv3 (documentación oficial de net-snmp)

Zabbix interno

Descripción general

Las comprobaciones internas permiten supervisar los procesos internos de Zabbix. En otras palabras, puede supervisar lo que ocurre con Zabbix server o Zabbix proxy.

Las comprobaciones internas se calculan:

- en Zabbix server, si el host está supervisado por el server
- en Zabbix proxy, si el host está supervisado por el proxy

Las comprobaciones internas son procesadas por el server o el proxy independientemente del estado de mantenimiento del host.

Para utilizar este item, seleccione el tipo de item **Zabbix internal**.

> **Note:**
> Las comprobaciones internas son procesadas por los sondeadores de Zabbix.

Rendimiento

El uso de algunos items internos puede afectar negativamente al rendimiento. Estos items son:

- `zabbix[host,,items]`
- `zabbix[host,,items_unsupported]`
- `zabbix[hosts]`
- `zabbix[items]`
- `zabbix[items_unsupported]`
- `zabbix[queue,,]`
- `zabbix[requiredperformance]`
- `zabbix[stats,,,queue,,]`
- `zabbix[triggers]`

Las secciones del frontend Información del sistema y Cola también se ven afectadas.

Comprobaciones compatibles

Las claves de los item se enumeran sin parámetros personalizables ni información adicional.
Haga clic en la clave del item para consultar todos los detalles.

Clave del item	Descripción
zabbix[apm,]	Las estadísticas de los datos de telemetría recibidos por el recopilador de APM.
zabbix[boottime]	La hora de inicio del proceso de Zabbix server o Zabbix proxy, en segundos.
zabbix[cluster,discovery,nodes]	Detecta los nodos del clúster de alta disponibilidad.
zabbix[connector_queue]	La cantidad de valores encolados en la cola del conector.
zabbix[discovery_queue]	La cantidad de comprobaciones de red encoladas en la cola de descubrimiento.
zabbix[host,,items]	La cantidad de item habilitados (compatibles y no compatibles) en el host.
zabbix[host,,items_unsupported]	La cantidad de item no compatibles habilitados en el host.
zabbix[host,maintenance]	El estado de mantenimiento actual del host.
zabbix[host,active_agent,available]	La disponibilidad de las comprobaciones de agent activo en el host.
zabbix[host,discovery,interfaces]	Los detalles de todas las interfaces configuradas del host en el frontend de Zabbix.
zabbix[host,available]	La disponibilidad de la interfaz principal para un tipo determinado de comprobaciones en el host.
zabbix[hosts]	La cantidad de hosts supervisados.
zabbix[items]	La cantidad de item habilitados (compatibles y no compatibles).
zabbix[items_unsupported]	La cantidad de item no compatibles.
zabbix[java,]	La información sobre Java gateway de Zabbix.
zabbix[lld_queue]	La cantidad de valores encolados en la cola de procesamiento de descubrimiento de bajo nivel.
zabbix[preprocessing]	Las estadísticas de los valores recibidos por el administrador de preprocesamiento.
zabbix[preprocessing_queue]	La cantidad de valores encolados en la cola de preprocesamiento.
zabbix[process,,]	El porcentaje de tiempo que un proceso específico de Zabbix o un grupo de procesos (identificado por <type> y <mode>) pasó en <state>.
zabbix[proxy,]	La información sobre el proxy de Zabbix.
zabbix[proxy,discovery]	La lista de proxy de Zabbix.
zabbix[proxy group,available]	La cantidad de proxy en línea en un grupo de proxy.
zabbix[proxy group,pavailable]	El porcentaje de proxy en línea en un grupo de proxy.
zabbix[proxy group,proxies]	La lista de proxy de Zabbix en un grupo de proxy.
zabbix[proxy group,state]	El estado de un grupo de proxy.

Clave del item	Descripción
zabbix[proxy group,discovery]	Devuelve una lista de grupos de proxy con datos de configuración y datos en tiempo real.
zabbix[proxy_buffer,buffer,]	Devuelve las estadísticas de uso del búfer de memoria del proxy.
zabbix[proxy_buffer,state,changes]	Devuelve la cantidad de cambios de estado entre los modos de búfer de disco y memoria desde el inicio.
zabbix[proxy_buffer,state,current]	Devuelve el estado de funcionamiento actual en el que se almacenan los datos nuevos.
zabbix[proxy_history]	La cantidad de valores en la tabla de historial del proxy que esperan ser enviados al server.
zabbix[queue,]	La cantidad de item supervisados en la cola que están retrasados al menos <from> segundos, pero menos de <to> segundos.
zabbix[rcache,]	Las estadísticas de disponibilidad de la caché de configuración de Zabbix.
zabbix[requiredperformance]	El rendimiento requerido de Zabbix server o Zabbix proxy, expresado como nuevos valores por segundo esperados.
zabbix[stats,]	Devuelve las métricas internas de un server o proxy de Zabbix. Si se especifican <ip> y <port>, las métricas se recuperan de la instancia remota; de lo contrario, de la instancia local.
zabbix[stats,,queue,]	Devuelve las métricas internas de la cola de un server o proxy de Zabbix. Si se especifican <ip> y <port>, las métricas se recuperan de la instancia remota; de lo contrario, de la instancia local.
zabbix[tcache,]	Las estadísticas de eficacia de la caché de la función de tendencias de Zabbix.
zabbix[triggers]	La cantidad de trigger habilitados en la base de datos de Zabbix, con todos los item habilitados en los hosts habilitados.
zabbix[uptime]	El tiempo de actividad del proceso de Zabbix server o proxy, en segundos.
zabbix[vcache,buffer,]	Las estadísticas de disponibilidad de la caché de valores de Zabbix.
zabbix[vcache,cache,]	Las estadísticas de eficacia de la caché de valores de Zabbix.
zabbix[version]	La versión de Zabbix server o proxy.
zabbix[vmware,buffer,]	Las estadísticas de disponibilidad de la caché de vmware de Zabbix.
zabbix[vps,written]	La cantidad total de valores del historial escritos en la base de datos.
zabbix[wcache,]	Las estadísticas y la disponibilidad de la caché de escritura de Zabbix.

Detalles de la clave del item

- Los parámetros sin corchetes angulares son obligatorios y deben usarse *tal cual* (por ejemplo, ”host” y ”available” en `zabbix[host,<type>,available]`).
- Los parámetros con corchetes angulares **< >** deben ser reemplazados por un valor válido. Si un parámetro tiene un valor por defecto, puede omitirse.
- Los valores de los items y parámetros de item etiquetados como ”no soportado en proxy” sólo pueden recuperarse si el host es monitorizado por el server. Por el contrario, los valores ”no soportados en server” sólo pueden recuperarse si el host es monitorizado por el proxy.

zabbix[apm,<type>]

 Las estadísticas de los datos de telemetría recibidos por el colector APM:

- *requests*: el número de solicitudes recibidas por el colector APM (contador):
  - *accepted*: el número de solicitudes aceptadas para su procesamiento;
  - *dropped*: el número de solicitudes descartadas;
- *rows*: el número de filas de datos de telemetría escritas en el origen de datos, por tipo de datos (contador).

Valor devuelto: *objeto JSON*.

Parámetros:

- **type**: *traces*, *metrics* o *logs*. Si no se especifica, se devuelven las estadísticas de todos los tipos.

Comentarios:

- Este item no es compatible con el server de Zabbix.

Ejemplo de valores devueltos para `zabbix[apm]`:

```
{
  "requests": {
    "accepted": 15230,
    "dropped": 12
  },
  "rows": {
    "traces": 48710,
    "metrics_gauge": 31250,
```

```
    "metrics_sum": 18400,
    "metrics_histogram": 5120,
    "metrics_exponential_histogram": 640,
    "metrics_summary": 210,
    "logs": 22870
  }
}
```

zabbix[boottime]

 El tiempo de inicio del proceso Zabbix server o Zabbix proxy en segundos.
 Valor de retorno: *Entero*.

zabbix[cep]

 Complex event processing statistics.
 Return value: *JSON*.

zabbix[cluster,discovery,nodes]

 Descubre los nodos del clúster de alta disponibilidad.
 Valor devuelto: *objeto JSON*.

Comentarios:

- Este item se puede utilizar en el descubrimiento de bajo nivel.
- Los campos devueltos coinciden principalmente con las propiedades del objeto de nodo de alta disponibilidad.
- Campos adicionales:
  - `db_timestamp` - hora actual del server (marca de tiempo Unix) incluida en la respuesta
  - `lastaccess_age` - segundos transcurridos desde el último heartbeat del nodo (db_timestamp - lastaccess)
  - `status` - estado del nodo:
    - 0 - en espera
    - 1 - detenido manualmente
    - 2 - no disponible
    - 3 - activo

Ejemplo de valores devueltos:

```
[
  {
    "id": "ckvupihk70001z8mkpw5cg0u3",
    "name": "zabbix-prod-01",
    "status": 3,
    "address": "10.0.4.12:10051",
    "port": 10051,
    "lastaccess": 1756115995,
    "db_timestamp": 1756116000,
    "lastaccess_age": 5
  },
  {
    "id": "ckvx2a9k70004b1nq2hz9d7f",
    "name": "zabbix-standby-02",
    "status": 0,
    "address": "10.0.4.13:10051",
    "port": 10051,
    "lastaccess": 1756115550,
    "db_timestamp": 1756116000,
    "lastaccess_age": 450
  },
  {
    "id": "ckw0bq3l70007y4r1a0m5kz8",
    "name": "zabbix-backup-eu1",
    "status": 1,
    "address": "backup.example.com:10051",
    "port": 10051,
    "lastaccess": 1756105080,
    "db_timestamp": 1756116000,
    "lastaccess_age": 10920
  }
]
```

zabbix[connector_queue]

 El número de valores encolados en la cola del conector.
 Valor de retorno: *Entero*.

zabbix[discovery_queue]

 El número de comprobaciones de red encoladas en la cola de descubrimiento.
 Valor de retorno: *Entero*.

zabbix[equipo„métricas]

 El número de métricas habilitadas (soportadas y no soportadas) en el equipo.
 Valor de retorno: *Entero*.

zabbix[equipo„items_unsupported]

 El número de métricas habilitadas no soportadas en el equipo.
 Valor de retorno: *Entero*.

zabbix[host„maintenance]

 El estado de mantenimiento actual del host.
 Valores devueltos: *0* - estado normal; *1* - mantenimiento con recolección de datos; *2* - mantenimiento sin recolección de datos.

Comentarios:

- Este item siempre es procesado por el Zabbix server independientemente de la ubicación del host (en el server o en el proxy). El proxy no recibirá este item con los datos de configuración.
- El segundo parámetro debe estar vacío y está reservado para uso futuro.

zabbix[equipo,active_agent,available]

 La disponibilidad de comprobaciones de agente activo en el equipo.
 Valores de retorno: *0* - desconocido; *1* - disponible; *2* - no disponible.

zabbix[equipo,descubrimiento,interfaces]

 Los detalles de todas las interfaces configuradas del equipo en el frontend de Zabbix.
 Valor de retorno: *objeto JSON*.

Comentarios:

- Esta métrica puede utilizarse en el descubrimiento de bajo nivel.
- Esta métrica no es compatible con el proxy de Zabbix.

zabbix[host,<type>,available]

 La disponibilidad de la interfaz principal de un tipo concreto de comprobaciones en el host.
 Valores devueltos: *0* - no disponible; *1* - disponible; *2* - desconocido.

Parámetros:

- **type** - *agent*, *snmp*, *ipmi* o *jmx*.

Comentarios:

- El valor del item se calcula de acuerdo con los parámetros de configuración relativos a la inaccesibilidad/no disponibilidad del host.

zabbixhosts

 El número de equipos monitorizados.
 Valor de retorno: *Entero*.

zabbixitems

 El número de métricas habilitadas (soportadas y no soportadas).
 Valor de retorno: *Entero*.

zabbix[items_unsupported]

 El número de métricas no soportadas.
 Valor de retorno: *Entero*.

zabbix[java„<param>]

 Información sobre Zabbix Java gateway.
 Valores devueltos: *1* - si `<param>` es *ping*; *Java gateway version* - si `<param>` es *version* (por ejemplo: ”8.0.0”).

Parámetros:

- **param** - *ping* o *version*.

Comentarios:

- Este item se puede usar para comprobar la disponibilidad de Java gateway mediante la función de trigger `nodata()`.
- El segundo parámetro debe estar vacío y está reservado para uso futuro.

zabbix[lld_queue]

 El número de valores encolados en la cola de procesamiento de bajo nivel de descubrimiento.
 Valor de retorno: *Entero*.

Comentarios:

- Esta métrica puede usarse para monitorizar la longitud de la cola de procesamiento de bajo nivel de descubrimiento.

zabbix[preprocessing]

 Estadísticas de los valores recibidos por el gestor de preprocesamiento:

- *queued* - el número y tamaño de los valores en cola que requieren preprocesamiento (contador)
- *direct* - el número y tamaño de los valores en cola que no requieren preprocesamiento (contador)
- *queue* - el número de valores encolados en la cola de preprocesamiento (igual que `zabbix[preprocessing_queue]`)

Valor devuelto: *JSON*.

Ejemplo de valores devueltos:

```
{"data":
    {
    "queued": {
        "count": 106,
        "size": 58620
    },
    "direct": {
        "count": 395,
        "size": 33843
    },
    "queue": 0
    }
}
```

zabbix[preprocessing_queue]

 El número de valores encolados en la cola de preprocesamiento.
 Valor devuelto: *Entero*.

Comentarios:

- Esta métrica puede usarse para monitorizar la longitud de la cola de preprocesamiento.

zabbix[process,<type>,<mode>,<state>]

 El porcentaje de tiempo que un proceso de Zabbix específico o un grupo de procesos (identificados mediante `<type>` y `<mode>`) pasó en `<state>`. Se calcula únicamente para el último minuto.
 Valor devuelto: *Float*.

Parámetros:

- **type**:
  - para procesos del server: *agent poller*, *alert manager*, *alert syncer*, *alerter*, *availability manager*, *browser poller*, *configuration syncer*, *configuration syncer worker*, *connector manager*, *connector worker*, *discovery manager*, *discovery worker*, *escalator*, *ha manager*, *history poller*, *history syncer*, *housekeeper*, *http agent poller*, *http poller*, *icmp pinger*, *internal poller ipmi manager*, *ipmi poller*, *java poller*, *lld manager*, *lld worker*, *odbc poller*, *poller*, *preprocessing manager*, *preprocessing worker*, *proxy group manager*, *proxy poller*, *self-monitoring*, *service manager*, *snmp poller*, *snmp trapper*, *task manager*, *timer*, *trapper*, *trigger housekeeper*, *unreachable poller*, *vmware collector*;
  - para procesos del proxy: *agent poller*, *apm manager*, *apm worker*, *availability manager*, *browser poller*, *configuration syncer*, *data sender*, *discovery manager*, *discovery worker*, *history syncer*, *housekeeper*, *http agent poller*, *http poller*, *icmp pinger*, *internal poller ipmi manager*, *ipmi poller*, *java poller*, *odbc poller*, *poller*, *preprocessing manager*, *preprocessing worker*, *self-monitoring*, *snmp poller*, *snmp trapper*, *task manager*, *trapper*, *unreachable poller*, *vmware collector*;
- **mode**:
  - *avg*: valor promedio de todos los procesos de un tipo determinado (predeterminado);
  - *count*: devuelve el número de forks de un tipo de proceso determinado; no se debe especificar `<state>`;
  - *max*: valor máximo;
  - *min*: valor mínimo;
  - *<process number>*: número del proceso (entre 1 y el número de instancias preforked; por ejemplo, si hay 4 trappers en ejecución, el valor está entre 1 y 4);
- **state**:
  - *busy*: el proceso está ocupado, por ejemplo, procesando una solicitud (predeterminado);
  - *idle*: el proceso está inactivo y no hace nada.

Comentarios:

- Si <mode> es un número de proceso de Zabbix que no está en ejecución (por ejemplo, si hay 5 pollers en ejecución y se especifica <mode> como 6), el item pasará a no soportado.
- El mínimo y el máximo se refieren al porcentaje de uso de un solo proceso. Por lo tanto, si en un grupo de 3 pollers los porcentajes de uso por proceso fueran 2, 18 y 66, *min* devolvería 2 y *max* devolvería 66.
- Los procesos informan de lo que están haciendo en la memoria compartida y el proceso de self-monitoring resume esos datos cada segundo. Los cambios de estado (busy/idle) se registran cuando se producen; por lo tanto, un proceso que pasa a estar ocupado se registra como tal y no cambia ni actualiza el estado hasta que pasa a estar inactivo. Esto garantiza que incluso los procesos completamente bloqueados se registren correctamente como ocupados al 100 %.
- Actualmente, ”busy” significa ”no estar durmiendo”, pero en el futuro podrían introducirse estados adicionales: espera de bloqueos, ejecución de consultas a la base de datos, etc. Tenga en cuenta que los pollers asíncronos se consideran ocupados si han alcanzado el límite establecido por el parámetro de configuración MaxConcurrentChecksPerPoller del server/proxy.
- En Linux y en la mayoría de los demás sistemas, la resolución es de 1/100 de segundo.

Ejemplos:

```
zabbix[process,poller,avg,busy] #el tiempo promedio que los procesos poller pasaron haciendo algo durante
zabbix[process,"icmp pinger",max,busy] #el tiempo máximo que cualquier proceso ICMP pinger pasó haciendo a
zabbix[process,"history syncer",2,busy] #el tiempo que el proceso history syncer número 2 pasó haciendo al
zabbix[process,trapper,count] #la cantidad de procesos trapper actualmente en ejecución
```

zabbix[proxy,<name>,<param>]

 Información sobre el proxy de Zabbix.
 Valor devuelto: *Integer*.

Parámetros:

- **name**: nombre del proxy;
- **param**:
  - *lastaccess*: marca de tiempo del último mensaje de latido recibido del proxy;
  - *delay*: tiempo durante el que los valores recopilados no se han enviado; se calcula como «retraso del proxy» + («hora actual del server» - «lastaccess del proxy»), donde «retraso del proxy» es la diferencia entre la hora actual del proxy y la marca de tiempo del valor no enviado más antiguo en el proxy.

Comentarios:

- Este item no es compatible con los hosts supervisados por un proxy o un grupo de proxies.
- La función `fuzzytime()` se puede utilizar para comprobar la disponibilidad del proxy.

Ejemplo:

```
zabbix[proxy,"Germany",lastaccess] #marca de tiempo del último mensaje de latido recibido del proxy "Germa
```

zabbix[proxy,discovery]

 La lista de proxies de Zabbix con el nombre, el modo, el cifrado, la compresión, la versión, la última vez que se detectó, el número de hosts, el número de items, los valores requeridos por segundo (vps), el estado de la versión (actual/obsoleta/no compatible), los tiempos de espera por tipo de item, el nombre del grupo de proxies (si el proxy pertenece a un grupo) y el estado (desconocido/sin conexión/en línea).
 Valor devuelto: *objeto JSON*.

Comentario:

- Este item no es compatible con hosts supervisados por un proxy o un grupo de proxies.

Ejemplo de valores devueltos:

```
[
  {
    "name":"Zabbix proxy",
    "passive":false,
    "unencrypted":true,
    "psk":false,
    "cert":false,
    "items":22,
    "compression":true,
    "version":"8.0.0beta2",
    "compatibility":1,
    "last_seen":1,
    "hosts":1,
    "requiredperformance":0.333333,
```

```
    "timeouts": {
        "zabbix_agent":3,
        "simple_check":3,
        "snmp_check":3,
        "external_check":3,
        "db_monitor":3,
        "http_agent":3,
        "ssh_agent":3,
        "telnet_agent":3,
        "script":3,
        "browser":60
    },
    "state":"online"
  }
]
```

zabbix[proxy group,<name>,available]

 El número de proxies en línea en un grupo de proxies.
 Valor devuelto: *Integer*.

Parámetros:

- **name** - el nombre del grupo de proxies.

Comentario:

- Este item no es compatible con los hosts supervisados por un proxy o un grupo de proxies.

zabbix[proxy group,<name>,pavailable]

 El porcentaje de proxies en línea en un grupo de proxies.
 Valor devuelto: *Float*.

Parámetros:

- **name**: el nombre del grupo de proxies.

Comentario:

- Este item no es compatible con los hosts supervisados por un proxy o un grupo de proxies.

zabbix[proxy group,<name>,proxies]

 La lista de proxies de Zabbix en un grupo de proxies con nombre, modo, cifrado, compresión, versión, última vez visto, cantidad de hosts, cantidad de items, valores requeridos por segundo (vps), estado de la versión (actual/desactualizada/no compatible), tiempos de espera, nombre del grupo de proxies y estado (desconocido/fuera de línea/en línea).
 Valor devuelto: *JSON*.

Parámetros:

- **name** - el nombre del grupo de proxies.

Comentario:

- Este item no es compatible con hosts supervisados por un proxy o un grupo de proxies.

zabbix[proxy group,<name>,state]

 El estado de un grupo de proxy.
 Valor devuelto: *0* - desconocido; *1* - sin conexión; *2* - en recuperación; *3* - en línea; *4* - degradándose.

Parámetros:

- **name** - el nombre del grupo de proxy.

Comentario:

- Este item no es compatible con los hosts supervisados por un proxy o un grupo de proxy.

zabbix[proxy group,discovery]

 Devuelve una lista de grupos de proxy con datos de configuración y datos en tiempo real. Los datos de configuración incluyen el nombre del grupo de proxy, el retraso de conmutación por error y el número mínimo de proxies en línea requerido. Los datos en tiempo real incluyen el estado del grupo de proxy (consulte los comentarios para obtener más información), el número de proxies en línea y el porcentaje de proxies en línea.
 Valor devuelto: *JSON*.

Comentarios:

- Este item no es compatible con los hosts supervisados por un proxy o un grupo de proxies.
- Este item no devuelve proxies que no pertenezcan a ningún grupo.
- Si hay un valor no válido para `failover_delay` o `min_online`, se informa del valor especial −1 para indicarlo. Los valores no válidos pueden producirse si se utilizan macros para la configuración y estas no se pueden expandir a un valor válido.
- El estado del grupo de proxy se informa como un entero: *0* - desconocido; *1* - sin conexión; *2* - en recuperación; *3* - en línea; *4* - degradándose.

Ejemplo de valores devueltos:

```
{
    "groups": [
       { "name": "Riga", "failover_delay": 60, "min_online": 1 },
       { "name": "Tokyo", "failover_delay": 60, "min_online": 2 },
       { "name": "Porto Alegre", "failover_delay": 60, "min_online": 3 }
    ],
    "details": {
        "Riga": { "state": 3, "available": 10, "pavailable": 20 },
        "Tokyo": { "state": 3, "available": 10, "pavailable": 20 },
        "Porto Alegre": { "state": 1, "available": 0, "pavailable": 0 }
    }
}
```

zabbix[proxy_buffer,buffer,<mode>]

 Las estadísticas de uso del búfer de memoria del proxy.
 Valores devueltos: *Integer* (para el tamaño); *Float* (para el porcentaje).

Parámetros:

- **mode**:
  - *total* - el tamaño total del búfer (se puede utilizar para comprobar si el búfer de memoria está habilitado);
  - *free* - el tamaño del búfer libre;
  - *pfree* - el porcentaje del búfer libre;
  - *used* - el tamaño del búfer utilizado;
  - *pused* - el porcentaje del búfer utilizado.

Comentarios:

- Devuelve un error «Proxy memory buffer is disabled» cuando el búfer de memoria está deshabilitado;
- Este item no es compatible con el server de Zabbix.

zabbix[proxy_buffer,state,changes]

 Devuelve el número de cambios de estado entre los modos de buffer de disco/memoria desde el inicio.
 Valores de retorno: *Entero*; *0* - el buffer de memoria está deshabilitado.

Comentarios:

- Los cambios de estado frecuentes indican que se debe aumentar el tamaño o la antigüedad del buffer de memoria.
- Si el estado del buffer de memoria se monitoriza con poca frecuencia (por ejemplo, una vez por minuto), entonces el buffer podría cambiar de estado sin que se registre.

zabbix[proxy_buffer,state,current]

 Devuelve el estado de funcionamiento actual donde se almacenan los datos nuevos.
 Valores devueltos: *0* - disco; *1* - memoria.

Comentario:

- También se devuelve 0 cuando el búfer de memoria está deshabilitado.

zabbix[proxy_history]

 El número de valores de la tabla de historial del proxy que están esperando para enviarse al server.
 Valores devueltos: *Integer*.

Comentario:

- Este item no es compatible con el server de Zabbix.

zabbix[queue,<from>,<to>]

 El número de items monitorizados en la cola que tienen un retraso de al menos `<from>` segundos, pero inferior a `<to>` segundos.
 Valor de retorno: *Integer*.

Parámetros:

- **from** - retrasado al menos por (el valor predeterminado es 6 segundos);
- **to** - retrasado como máximo por (el valor predeterminado es infinito).

Comentarios:

- Se admiten sufijos de tiempo (`s,m,h,d,w`) en los parámetros.

Attention:
El proxy de Zabbix no tiene conocimiento de los periodos de mantenimiento; consulte Cálculo de colas durante el mantenimiento para obtener más detalles.

zabbix[rcache,<cache>,<mode>]

 Las estadísticas de disponibilidad de la caché de configuración de Zabbix.
 Valores devueltos: *Integer* (para el tamaño); *Float* (para el porcentaje).

Parámetros:

- **cache** - *buffer*;
- **mode**:
  - *total* - el tamaño total del buffer;
  - *free* - el tamaño del buffer libre;
  - *pfree* - el porcentaje del buffer libre;
  - *used* - el tamaño del buffer usado;
  - *pused* - el porcentaje del buffer usado.

zabbix[requiredperformance]

 El rendimiento requerido del server o proxy de Zabbix, en nuevos valores por segundo esperados.
 Valor devuelto: *Float*.

Comentarios:

- Se correlaciona aproximadamente con "Required server performance, new values per second" en *Reports* > *System information*.

zabbix[stats,<ip>,<port>]

 Devuelve las métricas internas de un server o proxy de Zabbix. Si se especifican `<ip>` y `<port>`, las métricas se obtienen de la instancia remota; de lo contrario, de la instancia local.
 Valores devueltos: *objeto JSON*.

Parámetros:

- **ip** - la lista de IP/DNS/máscara de red de servers/proxies que se consultarán de forma remota (el valor predeterminado es 127.0.0.1);
- **port** - el puerto del server/proxy que se consultará de forma remota (el valor predeterminado es 10051).

Comentarios:

- La solicitud de estadísticas solo se aceptará desde las direcciones enumeradas en el parámetro `StatsAllowedIP` server/proxy de la instancia de destino.
- Este item devuelve un conjunto seleccionado de métricas internas. Para obtener más detalles, consulte Remote monitoring of Zabbix stats.

zabbix[stats,<ip>,<port>,queue,<from>,<to>]

 Devuelve las métricas internas de la cola (véase `zabbix[queue,<from>,<to>]`) de un server o proxy de Zabbix. Si se especifican `<ip>` y `<port>`, las métricas se obtienen de la instancia remota; de lo contrario, de la instancia local.
 Valores devueltos: *objeto JSON*.

Parámetros:

- **ip** - la lista de IP/DNS/máscara de red de servers/proxies que se consultarán de forma remota (el valor predeterminado es `127.0.0.1`);
- **port** - el puerto del server/proxy que se consultará de forma remota (el valor predeterminado es 10051);
- **from** - retrasado al menos (el valor predeterminado es 6 segundos);
- **to** - retrasado como máximo (el valor predeterminado es infinito).

Comentarios:

- La solicitud de estadísticas solo se aceptará desde las direcciones enumeradas en el parámetro `StatsAllowedIP` server/proxy de la instancia de destino.

- Este item devuelve un conjunto seleccionado de métricas internas. Para obtener detalles, consulte Remote monitoring of Zabbix stats.

Attention:
El proxy de Zabbix no tiene conocimiento de los periodos de mantenimiento; consulte Calculation of queues during maintenance para obtener detalles.

zabbix[tcache,cache,<parameter>]

 Las estadísticas de eficacia de la caché de la función de tendencias de Zabbix.
 Valores devueltos: *Integer* (para el tamaño); *Float* (para el porcentaje).

Parámetros:

- **parameter**:
  - *all* - solicitudes totales de la caché (predeterminado);
  - *hits* - aciertos de caché;
  - *phits* - porcentaje de aciertos de caché;
  - *misses* - fallos de caché;
  - *pmisses* - porcentaje de fallos de caché;
  - *items* - el número de elementos almacenados en caché;
  - *requests* - el número de solicitudes almacenadas en caché;
  - *pitems* - porcentaje de elementos almacenados en caché respecto de elementos almacenados en caché + solicitudes. Un porcentaje bajo probablemente significa que el tamaño de la caché se puede reducir.
  - *paccessed* - porcentaje de entradas accedidas en las últimas 24 horas.

Comentarios:

- Este item no es compatible con Zabbix proxy.

zabbixtriggers

 El número de disparadores habilitados en la base de datos de Zabbix, con todas las métricas habilitadas en los equipos habilitados.
 Valor de retorno: *Entero*.

Comentarios:

- Esta métrica no es compatible con el proxy de Zabbix.

zabbix[uptime]

 El tiempo de actividad del proceso del servidor o proxy de Zabbix en segundos.
 Valor devuelto: *Entero*.

zabbix[vcache,buffer,<mode>]

 Las estadísticas de disponibilidad de la caché de valores de Zabbix.
 Valores devueltos: *Integer* (para el tamaño); *Float* (para el porcentaje).

Parámetros:

- **mode**:
  - *total* - el tamaño total del buffer;
  - *free* - el tamaño del buffer libre;
  - *pfree* - el porcentaje del buffer libre;
  - *used* - el tamaño del buffer usado;
  - *pused* - el porcentaje del buffer usado.

Comentarios:

- Este item no es compatible con Zabbix proxy.

zabbix[vcache,cache,<parameter>]

 Las estadísticas de eficacia de la caché de valores de Zabbix.
 Valor devuelto: *Integer*. Si <parameter> se establece en *mode*, el valor devuelto es: *0* - modo normal; *1* - modo de memoria baja.

Parámetros:

- **parameter**:
  - *requests* - el número total de solicitudes;
  - *hits* - el número de aciertos de caché (valores históricos tomados de la caché);
  - *misses* - el número de fallos de caché (valores históricos tomados de la base de datos);
  - *mode* - el modo de funcionamiento de la caché de valores.

Comentarios:

- Una vez activado el modo de memoria baja, la caché de valores permanecerá en este estado durante 24 horas, incluso si el problema que activó este modo se resuelve antes.
- Puede usar esta clave con el paso de preprocesamiento *Change per second* para obtener estadísticas de valores por segundo.
- Este item no es compatible en Zabbix proxy.

zabbix[version]

 La versión de Zabbix server o proxy.
 Valor devuelto: *String*. Por ejemplo: `8.0.0`.

zabbix[vmware,buffer,<mode>]

 Las estadísticas de disponibilidad de la caché vmware de Zabbix.
 Valores devueltos: *Integer* (para el tamaño); *Float* (para el porcentaje).

Parámetros:

- **mode**:
  - *total* - el tamaño total del buffer;
  - *free* - el tamaño del buffer libre;
  - *pfree* - el porcentaje de buffer libre;
  - *used* - el tamaño del buffer usado;
  - *pused* - el porcentaje de buffer usado.

zabbix[vps,written]

 El número total de valores históricos escritos en la base de datos.
 Valor de retorno: *Entero*.

zabbix[wcache,<cache>,<mode>]

 Las estadísticas y la disponibilidad de la caché de escritura de Zabbix.
 Valores devueltos: *Integer* (para número/tamaño); *Float* (para porcentaje).

Parámetros:

- **cache** - *values*, *history*, *index* o *trend*;
- **mode**:
  - con el parámetro de caché *values*:
    - *all* - el número total de valores procesados por el server/proxy de Zabbix, excepto los items no soportados (contador) (predeterminado);
    - *float* - el número de valores float procesados (contador);
    - *uint* - el número de valores enteros sin signo procesados (contador);
    - *str* - el número de valores de caracteres/cadenas procesados (contador);
    - *log* - el número de valores de log procesados (contador);
    - *text* - el número de valores de texto procesados (contador);
    - *bin* - el número de valores binarios procesados (contador);
    - *json* - el número de valores JSON procesados (contador);
    - *not supported* - el número de veces que el procesamiento de un item dio como resultado que el item pasara a no soportado o mantuviera ese estado (contador);
  - con los parámetros de caché *history*, *index*, *trend*:
    - *pfree* - el porcentaje de búfer libre (predeterminado);
    - *total* - el tamaño total del búfer;
    - *free* - el tamaño del búfer libre;
    - *used* - el tamaño del búfer usado;
    - *pused* - el porcentaje de búfer usado.

Comentarios:

- El parámetro de caché `trend` no es compatible con Zabbix proxy.
- La caché de history se usa para almacenar los valores de los items. Un número bajo indica problemas de rendimiento en la base de datos.
- La caché de índice de history se usa para indexar los valores almacenados en la caché de history.
- Después de que la caché de history se llena y luego se vacía, la caché de índice de history seguirá conservando algunos datos. Este comportamiento es esperado y ayuda a que el sistema funcione de forma más eficiente al evitar el procesamiento adicional necesario para redimensionar constantemente la memoria.
- La caché de trend almacena el agregado de la hora actual para todos los items que reciben datos.
- Puede usar la clave `zabbix[wcache,values]` con el paso de preprocesamiento *Change per second* para obtener estadísticas de valores por segundo.

Zabbix trapper

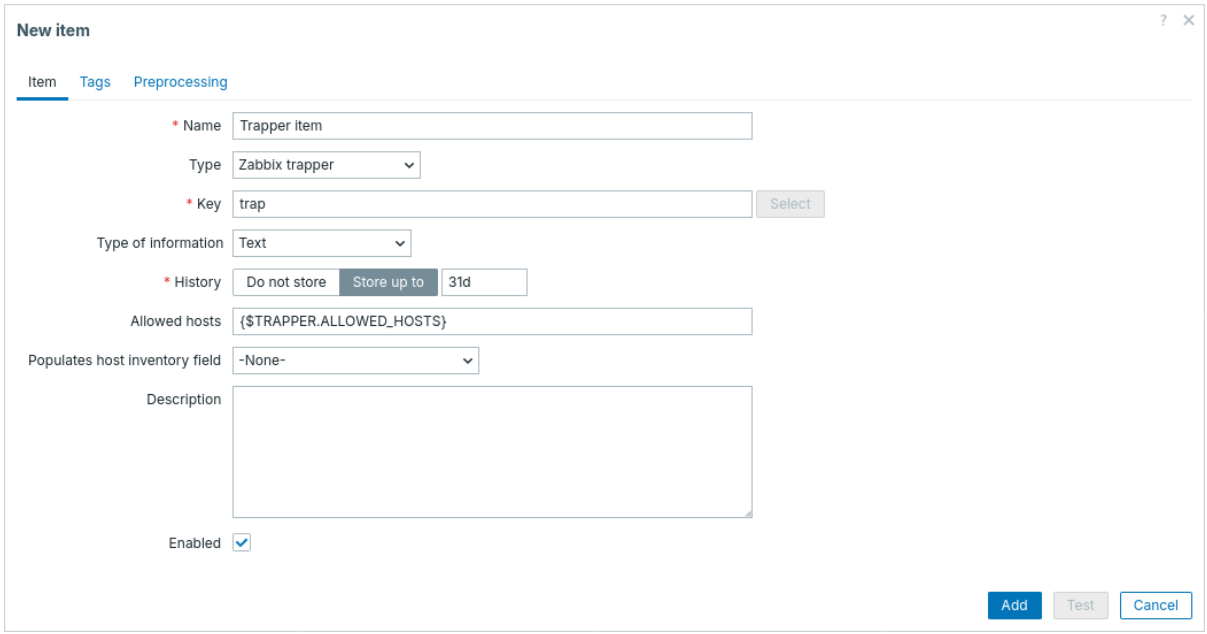
Resumen

Los items trapper aceptan datos entrantes en lugar de consultarlos. Esto es útil para cualquier dato que desee enviar a Zabbix.

Configuración

Para configurar un item trapper:

1. Vaya a *Data collection* > *Hosts*.
2. Haga clic en *Items* en la fila del host.
3. Haga clic en *Create item*.
4. Introduzca los parámetros del item en el formulario de configuración.



Todos los campos obligatorios están marcados con un asterisco rojo.

Los campos que requieren información específica para los items trapper son:

Type	Seleccione ”Zabbix trapper”.
Key	Introduzca una clave que se utilizará para reconocer el item al enviar datos a Zabbix server.
Type of information	Seleccione el tipo de información que corresponderá al formato de los datos que se enviarán.
Allowed hosts	Lista de direcciones IP separadas por comas (opcionalmente en notación CIDR) o nombres DNS. El valor predeterminado es la user macro `{$TRAPPER.ALLOWED_HOSTS}`. De forma predeterminada, `{$TRAPPER.ALLOWED_HOSTS}` es `127.0.0.1,::1`. Puede introducir las direcciones directamente en el campo o cambiar el valor de la macro. Si se deja vacío, se denegarán las conexiones entrantes desde todos los hosts. Si se especifica, solo se aceptarán conexiones entrantes desde los hosts enumerados aquí. Si la compatibilidad con IPv6 está habilitada, entonces ’127.0.0.1’, ’::127.0.0.1’, ’::ffff:127.0.0.1’ se tratan por igual y ’::/0’ permitirá cualquier dirección IPv4 o IPv6. Se puede usar ’0.0.0.0/0’ para permitir cualquier dirección IPv4. Tenga en cuenta que las ”direcciones IPv6 compatibles con IPv4” (prefijo 0000::/96) son compatibles, pero están obsoletas según RFC4291. Ejemplo: 127.0.0.1, 192.168.1.0/24, 192.168.3.1-255, 192.168.1-10.1-255, ::1,2001:db8::/32, mysqlserver1, zabbix.example.com, {HOST.HOST} Se admiten espacios, user macros y host macros {HOST.HOST}, {HOST.NAME}, {HOST.IP}, {HOST.DNS}, {HOST.CONN}.

Note:
Antes de enviar valores, es posible que tenga que esperar hasta 60 segundos después de guardar el item hasta que Zabbix server recoja los cambios de una actualización de la caché de configuración.

Envío de datos

El envío de datos al server o proxy de Zabbix es posible mediante la utilidad Zabbix sender o el protocolo de Zabbix sender. También es posible enviar datos al server de Zabbix mediante el método de API `history.push`.

Zabbix sender

Para enviar datos a Zabbix server o proxy mediante la utilidad Zabbix sender, puede ejecutar el siguiente comando para enviar el "test value":

```
zabbix_sender -z <server IP address> -p 10051 -s "New host" -k trap -o "test value"
```

Para enviar el "test value", se utilizan las siguientes opciones del comando:

- `-z` para especificar la dirección IP de Zabbix server
- `-p` para especificar el número de puerto de Zabbix server (10051 de forma predeterminada)
- `-s` para especificar el host (asegúrese de usar el nombre técnico en lugar del host name visible)
- `-k` para especificar la clave del item configurado en el item trapper
- `-o` para especificar el valor que se enviará

Attention:
El proceso trapper de Zabbix no expande los macros usados en la clave del item para comprobar la existencia de la clave del item correspondiente para el host de destino.

Para obtener más información sobre la comunicación entre Zabbix sender y Zabbix server o proxy, consulte *Zabbix sender protocol*.

history.push

Para enviar datos al servidor Zabbix utilizando el método API `history.push`, puede realizar la siguiente solicitud HTTP POST que contiene algunos valores de prueba:

```
curl --request POST \
  --url 'https://example.com/zabbix/api_jsonrpc.php' \
  --header 'Authorization: Bearer 0424bd59b807674191e7d77572075f33' \
  --header 'Content-Type: application/json-rpc' \
  --data '{"jsonrpc":"2.0","method":"history.push","params":[{"itemid":10600,"value":"valor de prueba 1"},
```

Si la solicitud es correcta, la respuesta devuelta por la API podría ser la siguiente:

```
{
    "jsonrpc": "2.0",
    "result": {
        "response": "success",
        "data": [
            {
                "itemid": "10600"
            },
            {
                "itemid": "10601",
                "error": "La métrica está deshabilitada."
            },
            {
                "error": "Sin permisos para el objeto referido o no existe."
            }
        ]
    },
    "id": 1
}
```

Los errores en los datos de respuesta indican que el envío de datos para métricas específicas no ha superado la validación por parte del servidor Zabbix. Esto puede ocurrir por las siguientes razones:

- el usuario que envía los datos no tiene permiso de *lectura* para el equipo de la métrica;
- el equipo está deshabilitado o en mantenimiento sin recolección de datos;

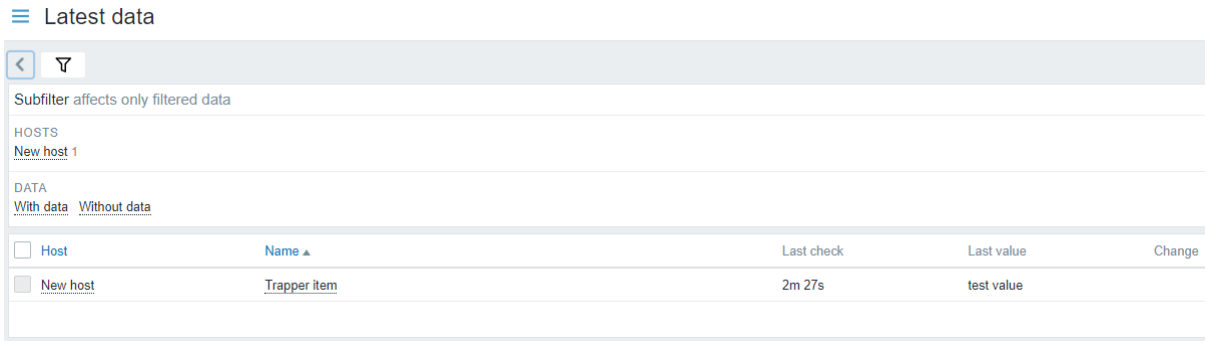
- la métrica no existe o aún no está incluida en la caché de configuración del servidor;
- la métrica está deshabilitada o su tipo es distinto de Zabbix trapper o HTTP agent (con trapping habilitado);
- la IP o DNS del usuario no está establecida en la lista de *Equipos permitidos* de la métrica;
- otra métrica tiene un valor con una marca de tiempo duplicada a nivel de nanosegundos.

La ausencia de errores indica que los valores enviados han sido aceptados para su procesamiento, lo que incluye preprocesamiento (si lo hay), procesamiento de disparadores y guardado en la base de datos. Tenga en cuenta que el procesamiento de un valor aceptado también puede fallar (por ejemplo, durante el preprocesamiento), lo que resulta en que el valor sea descartado.

Para obtener más información sobre cómo trabajar con la API de Zabbix, consulte API.

Mostrando datos

Una vez que se envían los datos, puede ir a *Monitoring* > *Latest data* para ver el resultado:



Note:
Si se envía un único valor numérico, el gráfico de datos mostrará una línea horizontal a la izquierda y a la derecha del punto temporal del valor.

Comprobación externa

Resumen

Una comprobación externa es una comprobación ejecutada por Zabbix server mediante la ejecución de un script de shell o un binario. Sin embargo, cuando los hosts son monitorizados por un Zabbix proxy, las comprobaciones externas son ejecutadas por el proxy.

Las comprobaciones externas no requieren que ningún agent se esté ejecutando en el host monitorizado.

La sintaxis de la clave del item es:

`script[<parameter1>,<parameter2>,...]`

Donde:

ARGUMENT	DEFINITION
script	Nombre de un script de shell o de un binario.
parameter(s)	Parámetros opcionales de la línea de comandos.

Si no desea pasar ningún parámetro al script, puede usar:

`script[]` o
`script`

Zabbix server o proxy buscará el directorio especificado para scripts externos y ejecutará el comando (consulte el parámetro `ExternalScripts` en el archivo de configuración de Zabbix server/proxy). El comando se ejecutará con el mismo usuario que Zabbix server/proxy, por lo que cualquier permiso de acceso o variable de entorno debe gestionarse en un script envoltorio, si es necesario. Los permisos sobre el comando también deben permitir que ese usuario lo ejecute. Solo los comandos del directorio especificado están disponibles para su ejecución.

Warning:
No abuse de las comprobaciones externas, ya que cada script requiere que Zabbix server/proxy inicie un proceso fork, y ejecutar muchos scripts puede reducir significativamente el rendimiento de Zabbix.

Ejemplo de uso

Ejecución del script `check_oracle.sh` con el primer parámetro `'-h'`. El segundo parámetro se reemplazará por la dirección IP o el nombre DNS, según la selección en las propiedades del host.

`check_oracle.sh["-h","{HOST.CONN}"]`

Suponiendo que el host está configurado para usar la dirección IP, el server/proxy de Zabbix ejecutará:

`check_oracle.sh '-h' '192.168.1.4'`

Resultado de la comprobación externa

El valor de retorno de una comprobación externa es una salida estándar junto con un error estándar producido por la comprobación.

> **Attention:**
> Un item que devuelve texto (carácter, log o tipo de información de texto) no se volverá no soportado en caso de una salida de error estándar.

El valor de retorno está limitado a 16MB (incluyendo los espacios en blanco finales que se truncan); también se aplican los límites de la base de datos.

Si no se encuentra el script solicitado o el servidor/proxy de Zabbix no tiene permisos para ejecutarlo, el item se volverá no soportado y se mostrará un mensaje de error correspondiente.

En caso de timeout, el item se volverá no soportado, se mostrará un mensaje de error correspondiente y se terminará el proceso bifurcado para el script.

Monitor de base de datos

Resumen

El tipo de item *Database monitor* en el frontend de Zabbix se utiliza para la supervisión mediante ODBC (comprobaciones ODBC).

ODBC es una API de middleware del lenguaje de programación C para acceder a sistemas de gestión de bases de datos (DBMS). El concepto de ODBC fue desarrollado por Microsoft y posteriormente adaptado a otras plataformas.

Zabbix puede consultar cualquier base de datos compatible con ODBC. Para ello, Zabbix no se conecta directamente a las bases de datos, sino que utiliza la interfaz ODBC y los controladores configurados en ODBC. Esto permite una supervisión más eficiente de distintas bases de datos para múltiples fines (por ejemplo, comprobar colas específicas de la base de datos, estadísticas de uso, etc.).

Zabbix admite unixODBC, que es una de las implementaciones de API ODBC de código abierto más utilizadas.

> **Attention:**
> Véase también: known issues para las comprobaciones ODBC.

Instalación de unixODBC

La forma recomendada de instalar unixODBC es utilizar los repositorios de paquetes predeterminados del sistema operativo Linux. En las distribuciones de Linux más populares, unixODBC está incluido de forma predeterminada en el repositorio de paquetes. Si los paquetes no están disponibles, los archivos de origen se pueden obtener en la página principal de unixODBC: http://www.unixodbc.org/download.html.

Para instalar unixODBC, use el gestor de paquetes del sistema de su elección:

```
##### Para sistemas Ubuntu/Debian:
apt install unixodbc unixodbc-dev

##### Para sistemas basados en RedHat/Fedora:
dnf install unixODBC unixODBC-devel

##### Para sistemas basados en SUSE:
zypper in unixODBC-devel
```

Attention:
El paquete `unixodbc-dev` o `unixODBC-devel` es necesario para compilar Zabbix con soporte para unixODBC. Para habilitar el soporte ODBC, Zabbix debe compilarse con la siguiente opción de configuración:

`--with-unixodbc[=ARG] # Use ODBC driver against unixODBC package.`

Instalación de los controladores unixODBC

El controlador de base de datos unixODBC debe instalarse para la base de datos que se va a monitorizar. Para obtener una lista de las bases de datos y controladores compatibles, consulte la página principal de unixODBC: http://www.unixodbc.org/drivers.html.

Note:
En algunas distribuciones de Linux, los controladores de base de datos están incluidos en los repositorios de paquetes.

MySQL

Para instalar el controlador de base de datos MySQL unixODBC, utilice el gestor de paquetes para el sistema de su elección:

```
##### Para sistemas Ubuntu/Debian:
apt install odbc-mariadb

##### Para sistemas basados en RedHat/Fedora:
dnf install mariadb-connector-odbc

##### Para sistemas basados en SUSE:
zypper install mariadb-connector-odbc
```

Para instalar el controlador de base de datos sin un gestor de paquetes, consulte la documentación de MySQL para `mysql-connector-odbc`, o la documentación de MariaDB para `mariadb-connector-odbc`.

PostgreSQL

Para instalar el controlador de base de datos unixODBC de PostgreSQL, utilice el gestor de paquetes para el sistema de su elección:

```
##### Para sistemas Ubuntu/Debian:
apt install odbc-postgresql

##### Para sistemas basados en RedHat/Fedora:
dnf install postgresql-odbc

##### Para sistemas basados en SUSE:
zypper install psqlODBC
```

Para instalar el controlador de base de datos sin un gestor de paquetes, consulte la documentación de PostgreSQL.

Oracle

Para instalar el controlador de base de datos unixODBC, consulte la documentación de Oracle.

MSSQL

Para instalar el controlador de base de datos MSSQL unixODBC, utilice el gestor de paquetes para el sistema de su elección:

```
##### Para sistemas Ubuntu/Debian:
apt install tdsodbc

##### Para sistemas basados en RedHat/Fedora (paquetes EPEL: https://docs.fedoraproject.org/en-US/epel/):
dnf install epel-release
dnf install freetds

##### Para sistemas basados en SUSE:
zypper install libtdsodbc0
```

Para instalar el controlador de base de datos sin un gestor de paquetes, consulte la guía de usuario de FreeTDS.

Configuración de unixODBC

Para configurar unixODBC, debe editar los archivos `odbcinst.ini` y `odbc.ini`. Puede verificar la ubicación de estos archivos ejecutando el siguiente comando:

```
odbcinst -j
```

El resultado del comando debería contener información similar a la siguiente:

```
unixODBC 2.3.9
DRIVERS............: /etc/odbcinst.ini
SYSTEM DATA SOURCES: /etc/odbc.ini
FILE DATA SOURCES..: /etc/ODBCDataSources
```

odbcinst.ini

El archivo `odbcinst.ini` enumera los controladores de base de datos ODBC instalados. Si falta `odbcinst.ini`, es necesario crearlo manualmente.

```
[TEST_MYSQL]
Description=ODBC para MySQL
Driver=/usr/lib/libmyodbc5.so
FileUsage=1
```

Parámetro	Descripción
TEST_MYSQL	Nombre del controlador de base de datos.
Description	Descripción del controlador de base de datos.
Driver	Ubicación de la biblioteca del controlador de base de datos.
FileUsage	Determina si el controlador de base de datos admite la conexión a un servidor de base de datos sin soporte para acceder a archivos locales (0); admite la lectura de datos desde archivos (1); admite la escritura de datos en archivos (2).
Threading	Nivel de serialización de hilos. Compatible con PostgreSQL. Desde la 1.6, si el gestor de controladores se compila con soporte para hilos, puede agregar otra entrada de controlador.

odbc.ini

El archivo `odbc.ini` se utiliza para configurar fuentes de datos. Tenga en cuenta que la lista de parámetros admitidos depende del controlador de base de datos (por ejemplo, las bases de datos Oracle pueden usar ServerName en lugar de Server, etc.).

```
[TEST_MYSQL]
Description=Base de datos de prueba MySQL
Driver=mysql
Server=127.0.0.1
User=root
Password=
Port=3306
Socket=
Database=zabbix
```

Parámetro	Descripción
TEST_MYSQL	Nombre de la fuente de datos (DSN).
Description	Descripción de la fuente de datos.
Driver	Nombre del controlador de base de datos (como se especifica en `odbcinst.ini`).
Server	IP/DNS del servidor de base de datos.
User	Usuario de la base de datos para la conexión.
Password	Contraseña del usuario de la base de datos.
Port	Puerto de conexión a la base de datos.
Socket	Socket de conexión a la base de datos.
Database	Nombre de la base de datos.

Para otras posibles opciones de parámetros de configuración, consulte la documentación de MySQL.

El archivo `odbc.ini` para PostgreSQL puede contener parámetros adicionales:

```
[TEST_PSQL]
Description=Base de datos de prueba de PostgreSQL
Driver=postgresql
Username=zbx_test
```

```
Password=zabbix
Servername=127.0.0.1
Database=zabbix
Port=5432
ReadOnly=No
Protocol=8.0+
ShowOidColumn=No
FakeOidIndex=No
RowVersioning=No
ShowSystemTables=No
Fetch=Yes
BoolsAsChar=Yes
SSLmode=Require
ConnSettings=
```

Parámetro	Descripción
ReadOnly	Especifica si la conexión a la base de datos permite solo operaciones de lectura (consultas SELECT) y restringe las modificaciones (sentencias INSERT, UPDATE y DELETE); útil para escenarios donde los datos deben permanecer sin cambios.
Protocol	Versión del protocolo backend de PostgreSQL (se ignora al usar conexiones SSL).
ShowOidColumn	Especifica si se debe incluir el Object ID (OID) en SQLColumns.
FakeOidIndex	Especifica si se debe crear un índice único falso en OID.
RowVersioning	Especifica si se debe permitir que las aplicaciones detecten si los datos han sido modificados por otros usuarios mientras intenta actualizar una fila. Tenga en cuenta que este parámetro puede acelerar el proceso de actualización, ya que, para actualizar una fila, no es necesario especificar cada columna individual en la cláusula WHERE.
ShowSystemTables	Especifica si el controlador de base de datos debe tratar las tablas del sistema como tablas normales en SQLTables; útil para la accesibilidad, permitiendo la visibilidad en las tablas del sistema.
Fetch	Especifica si el controlador debe usar automáticamente declare cursor/fetch para manejar sentencias SELECT y mantener una caché de 100 filas.
BoolsAsChar	Controla el mapeo de tipos Booleanos. Si se establece en "Yes", los Booleanos se mapean a SQL_CHAR; de lo contrario, se mapean a SQL_BIT.
SSLmode	Especifica el modo SSL para la conexión.
ConnSettings	Configuraciones adicionales enviadas al backend en la conexión.

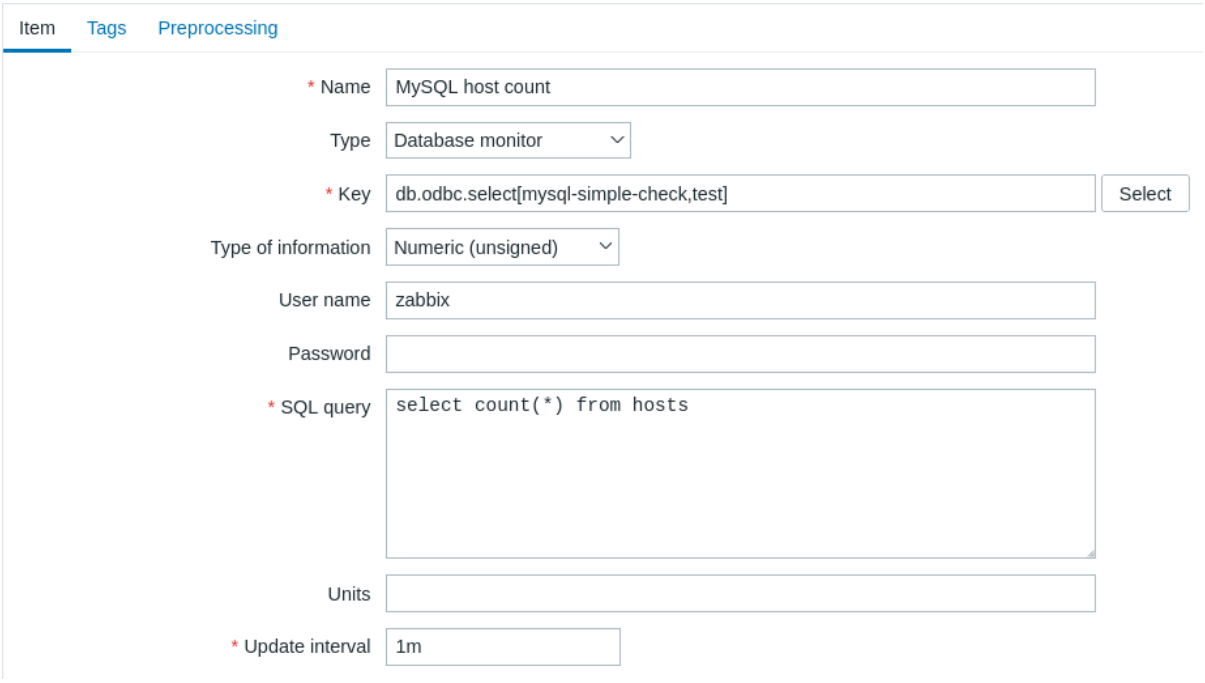
Probando la conexión ODBC

Para comprobar si la conexión ODBC funciona correctamente, puede utilizar la utilidad `isql` (incluida en el paquete `unixODBC`):

```
isql test
+---------------------------------------+
| ¡Conectado!                           |
|                                       |
| sentencia-sql                         |
| help [nombretabla]                    |
| quit                                  |
|                                       |
+---------------------------------------+
```

Configuración de item en Zabbix frontend

Configure un item de **monitorización de bases de datos**.



Todos los campos de entrada obligatorios están marcados con un asterisco rojo.

Para los items de monitorización de bases de datos, debe especificar:

Type	Seleccione ”Database monitor” aquí.
Key	Introduzca una de las claves de item compatibles: **db.odbc.select[]** - este item devuelve un solo valor (la primera columna de la primera fila del resultado de la consulta SQL); **db.odbc.get[]** - este item devuelve varias filas/columnas en formato JSON; **db.odbc.discovery[]** - este item devuelve datos de descubrimiento de bajo nivel.
User name	Introduzca el nombre de usuario de la base de datos (hasta 255 caracteres). Este parámetro es opcional si el nombre de usuario de la base de datos se especifica en el archivo `odbc.ini`. Si se usa una cadena de conexión y el campo *User name* no está vacío, se añade a la cadena de conexión como `UID=<user>`.
Password	Introduzca la contraseña del usuario de la base de datos (hasta 255 caracteres). Este parámetro es opcional si la contraseña se especifica en el archivo `odbc.ini`. Si se usa una cadena de conexión y el campo *Password* no está vacío, se añade a la cadena de conexión como `PWD=<password>`. Este campo admite caracteres especiales. Para todas las bases de datos excepto Oracle, las contraseñas que contengan `}` deben encerrarse entre llaves `{}`. La contraseña se añadirá a la cadena de conexión después del nombre de usuario, por ejemplo, `UID=<username>;PWD=P?;)*word`. Para probar la cadena resultante, puede ejecutar el siguiente comando: `isql -v -k 'Driver=libmaodbc.so;Database=zabbix;UID=zabbix;PWD=P?;)*word'`
SQL query	Introduzca la consulta SQL. Tenga en cuenta que con `db.odbc.select[]`, la consulta debe devolver solo un valor.
Type of information	Seleccione aquí el tipo de información que devolverá la consulta. Si el tipo de información se selecciona incorrectamente, el item pasará a no ser compatible.

Notas importantes

- Los items de monitoreo de base de datos dejarán de ser compatibles si no se inician procesos *odbc poller* en la configuración del server o del proxy. Para activar los ODBC pollers, establezca el parámetro `StartODBCPollers` en el archivo de configuración de Zabbix server o, para las comprobaciones realizadas por proxy, en el archivo de configuración de Zabbix proxy.
- El valor del parámetro *Timeout* en el formulario de configuración del item se utiliza como tiempo de espera de inicio de sesión ODBC y como tiempo de espera de ejecución de la consulta. Tenga en cuenta que estos ajustes de tiempo de espera podrían ignorarse si el controlador ODBC instalado no los admite.
- El comando SQL debe devolver un conjunto de resultados, como cualquier consulta que use la instrucción `select`. La sintaxis de la consulta dependerá del RDBMS que la procese. La sintaxis de la solicitud a un procedimiento almacenado debe comenzar con la palabra clave `call`.

Detalles de la clave de la métrica

Los parámetros sin corchetes angulares son obligatorios. Los parámetros marcados con corchetes angulares **< >** son opcionales.

db.odbc.select[<descripción corta única>,<dsn>,<cadena de conexión>]

 Devuelve un valor, es decir, la primera columna de la primera fila del resultado de la consulta SQL.
 Valor devuelto: Dependiendo de la consulta SQL.

Parámetros:

- **descripción corta única**: una descripción corta única para identificar la métrica (para usar en disparadores, etc.);
- **dsn**: el nombre de la fuente de datos (como se especifica en `odbc.ini`);
- **cadena de conexión**: la cadena de conexión (puede contener argumentos específicos del controlador).

Comentarios:

- Aunque `dsn` y `cadena de conexión` son parámetros opcionales, se requiere al menos uno de ellos; si ambos están definidos, se ignorará `dsn`.
- Si una consulta devuelve más de una columna, solo se lee la primera columna. Si una consulta devuelve más de una línea, solo se lee la primera línea.

db.odbc.get[<unique short description>,<dsn>,<connection string>]

 Transforma el resultado de la consulta SQL en un array JSON.
 Valor devuelto: *objeto JSON*.

Parámetros:

- **unique short description** - una breve descripción única para identificar el item (para usar en triggers, etc.);
- **dsn** - el nombre de la fuente de datos (según se especifica en `odbc.ini`);
- **connection string** - la cadena de conexión (puede contener argumentos específicos del controlador).

Comentarios:

- Aunque `dsn` y `connection string` son parámetros opcionales, se requiere al menos uno de ellos; si ambos están definidos, `dsn` se ignorará.
- Se pueden devolver varias filas/columnas en formato JSON. Este item puede usarse como un item maestro que recopila todos los datos en una sola llamada del sistema, mientras que el preprocesamiento JSONPath puede usarse en items dependientes para extraer valores individuales. Para obtener más información, consulte un ejemplo del formato devuelto, utilizado en la detección de bajo nivel.

Ejemplo:

```
# Connection for MySQL ODBC driver 5:
db.odbc.get[MySQL example,,"Driver=/usr/local/lib/libmyodbc5a.so;Database=master;Server=127.0.0.1;Port=330
```

db.odbc.discovery[<descripción corta única>,<dsn>,<cadena de conexión>]

 Transforma el resultado de la consulta SQL en un array JSON, utilizado para el descubrimiento de bajo nivel. Los nombres de las columnas del resultado de la consulta se convierten en nombres de macros de descubrimiento de bajo nivel emparejados con los valores de los campos descubiertos. Estas macros pueden utilizarse para crear prototipos de métricas, disparadores, etc.
 Valor de retorno: *objeto JSON*.

Parámetros:

- **descripción corta única**: una descripción corta única para identificar la métrica (para su uso en disparadores, etc.);
- **dsn**: el nombre de la fuente de datos (tal como se especifica en `odbc.ini`);
- **cadena de conexión**: la cadena de conexión (puede contener argumentos específicos del driver).

Comentarios:

- Aunque `dsn` y `cadena de conexión` son parámetros opcionales, se requiere al menos uno de ellos; si ambos están definidos, se ignorará `dsn`.

Mensajes de error

Los mensajes de error de ODBC están estructurados en campos para proporcionar información detallada. Por ejemplo, un mensaje de error podría verse así:

```
No se puede ejecutar la consulta ODBC: [SQL_ERROR]:[42601][7][ERROR: error de sintaxis en o cerca de ";";
```

- "`No se puede ejecutar la consulta ODBC`" - Mensaje de Zabbix
- "`[SQL_ERROR]`" - Código de retorno de ODBC
- "`[42601]`" - SQLState
- "`[7]`" - Código de error nativo

- "[ERROR: error de sintaxis en o cerca de ";"; Error al ejecutar la consulta]" - Mensaje de error nativo

Tenga en cuenta que la longitud del mensaje de error está limitada a 2048 bytes, por lo que el mensaje puede ser truncado. Si hay más de un registro de diagnóstico ODBC, Zabbix intenta concatenarlos (separados con |) hasta donde lo permita el límite de longitud.

agent HTTP

Overview

Este tipo de item permite la sondeo de datos mediante el protocolo HTTP/HTTPS. También es posible el trapping mediante la utilidad Zabbix sender o el protocolo de Zabbix sender (para enviar datos al server o proxy de Zabbix), o mediante el método de API `history.push` (para enviar datos al server de Zabbix).

Las comprobaciones de items HTTP son ejecutadas por el server de Zabbix. Sin embargo, cuando los hosts son monitorizados por un proxy de Zabbix, las comprobaciones de items HTTP son ejecutadas por el proxy.

Las comprobaciones de items HTTP no requieren que se ejecute ningún agent en el host monitorizado.

HTTP agent admite tanto HTTP como HTTPS. Zabbix seguirá opcionalmente las redirecciones (consulte la opción *Follow redirects* más abajo). El número máximo de redirecciones está codificado de forma fija en 10 (usando la opción de cURL `CURLOPT_MAXREDIRS`).

> **Attention:**
> El server/proxy de Zabbix debe estar configurado inicialmente con soporte para cURL (libcurl).

Las comprobaciones HTTP se ejecutan de forma asíncrona: no es necesario recibir la respuesta de una solicitud antes de que se inicien otras comprobaciones. La resolución DNS también es asíncrona.

La concurrencia máxima de las comprobaciones asíncronas es 1000 (definida por MaxConcurrentChecksPerPoller).

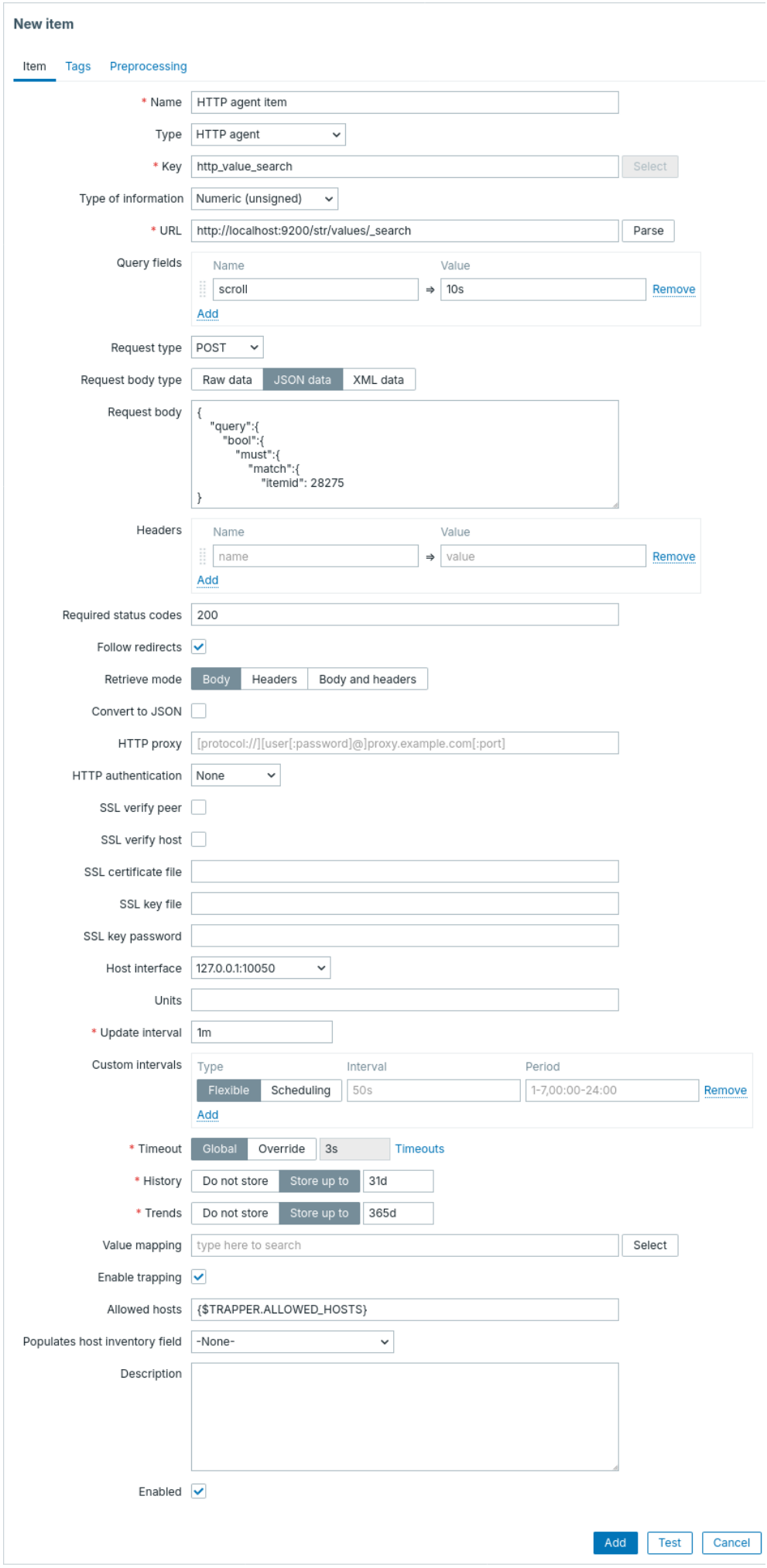
El número de pollers asíncronos de HTTP agent está definido por el parámetro StartHTTPAgentPollers.

La función de conexiones persistentes de cURL se ha añadido a las comprobaciones de HTTP agent desde Zabbix 7.0.

Configuración

Para configurar un item HTTP:

1. Vaya a *Recopilación de datos* > *Hosts*.
2. Haga clic en *Items* en la fila del host.
3. Haga clic en *Crear item*.
4. Introduzca los parámetros del item en el formulario.


New item
Item
Tags
Preprocessing
* Name
HTTP agent item
Type
HTTP agent
* Key
http_value_search
Select
Type of information
Numeric (unsigned)
* URL
http://localhost:9200/str/values/_search
Parse
Query fields
Name
Value
scroll
10s
Remove
Add
Request type
POST
Request body type
Raw data
JSON data
XML data
Request body
{
"query":{
"bool":{
"must":{
"match":{
"itemid": 28275
}
Headers
Name
Value
name
value
Remove
Add
Required status codes
200
Follow redirects
Retrieve mode
Body
Headers
Body and headers
Convert to JSON
HTTP proxy
[protocol://][user[:password]@]proxy.example.com[:port]
HTTP authentication
None
SSL verify peer
SSL verify host
SSL certificate file
SSL key file
SSL key password
Host interface
127.0.0.1:10050
Units
* Update interval
1m
Custom intervals
Type
Interval
Period
Flexible
Scheduling
50s
1-7,00:00-24:00
Remove
Add
* Timeout
Global
Override
3s
Timeouts
* History
Do not store
Store up to
31d
* Trends
Do not store
Store up to
365d
Value mapping
type here to search
Select
Enable trapping
Allowed hosts
{$TRAPPER.ALLOWED_HOSTS}
Populates host inventory field
-None-
Description
Enabled
Add
Test
Cancel

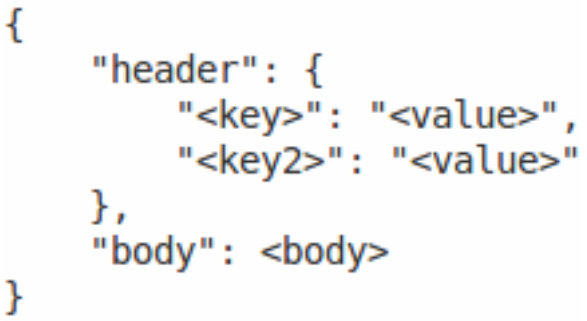
Todos los campos de entrada obligatorios están marcados con un asterisco rojo.

Los campos que requieren información específica para los items HTTP son:

Parámetro	Descripción
Tipo	Seleccione aquí **HTTP agent**.
Clave	Introduzca una clave de item única.

URL	URL a la que conectarse para recuperar datos. Por ejemplo:
	`https://www.example.com`
	`http://www.example.com/download`
	Los nombres de dominio pueden especificarse mediante caracteres Unicode. Se convierten automáticamente a punycode ASCII al ejecutar la comprobación HTTP.
	El botón *Analizar* puede utilizarse para separar los campos de consulta opcionales (como `?key=value&limit=10`) de la URL, trasladando los atributos y valores a *Campos de consulta* para la codificación automática de la URL.
	Por motivos de seguridad, no incluya nombres de usuario ni contraseñas reales en las URL. Utilice macros de usuario para evitar exponer las credenciales.
	Limitado a 2048 caracteres.
	Macros compatibles: {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {HOST.PORT}, {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG}, macros de usuario y macros de descubrimiento de bajo nivel.
	Esto establece la opción cURL CURLOPT_URL.

Campos de consulta	Variables para la URL (consulte lo anterior). Se especifican como pares de atributo y valor. Los valores se codifican automáticamente para la URL. Los valores de las macros se resuelven y, a continuación, se codifican automáticamente para la URL. Macros compatibles: {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {HOST.PORT}, {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG}, macros de usuario y macros de descubrimiento de bajo nivel. Esto establece la opción cURL CURLOPT_URL.
Tipo de solicitud	Seleccione el tipo de método de solicitud: *GET*, *POST*, *PUT* o *HEAD*.
Tipo de cuerpo de la solicitud	Seleccione el tipo de cuerpo de la solicitud: **Datos sin procesar**: cuerpo personalizado de la solicitud HTTP; las macros se sustituyen, pero no se realiza ninguna codificación. **Datos JSON**: cuerpo de la solicitud HTTP en formato JSON. Las macros pueden utilizarse como cadenas, números, true y false; las macros utilizadas como cadenas deben estar entre comillas dobles. Los valores de las macros se resuelven y se escapan automáticamente. Si no se especifica "Content-Type" en los encabezados, se utilizará de forma predeterminada "Content-Type: application/json". **Datos XML**: cuerpo de la solicitud HTTP en formato XML. Las macros pueden utilizarse como nodo de texto, atributo o sección CDATA. Los valores de las macros se resuelven y se escapan automáticamente en un nodo de texto y un atributo. Si no se especifica "Content-Type" en los encabezados, se utilizará de forma predeterminada "Content-Type: application/xml".
Cuerpo de la solicitud	Introduzca el cuerpo de la solicitud. Macros compatibles: {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {HOST.PORT}, {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG}, macros de usuario y macros de descubrimiento de bajo nivel.
Encabezados	Encabezados HTTP personalizados que se enviarán al realizar una solicitud. Se especifican como pares de atributo y valor. Macros compatibles: {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {HOST.PORT}, {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG}, macros de usuario y macros de descubrimiento de bajo nivel. Esto establece la opción cURL CURLOPT_HTTPHEADER.
Códigos de estado requeridos	Lista de códigos de estado HTTP esperados. Si Zabbix obtiene un código que no está en la lista, el item dejará de ser compatible. Si está vacía, no se realiza ninguna comprobación. Por ejemplo: 200,201,210-299. Macros compatibles en la lista: macros de usuario y macros de descubrimiento de bajo nivel. Esto utiliza la opción cURL CURLINFO_RESPONSE_CODE.
Seguir redirecciones	Marque la casilla para seguir las redirecciones HTTP. Esto establece la opción cURL CURLOPT_FOLLOWLOCATION.

Parámetro	Descripción
Modo de recuperación	Seleccione la parte de la respuesta que debe recuperarse: **Cuerpo**: solo el cuerpo **Encabezados**: solo los encabezados **Cuerpo y encabezados**: cuerpo y encabezados
Convertir a JSON	Los encabezados se guardan como pares de atributo y valor bajo la clave "header" si *Modo de recuperación* no está establecido en *Cuerpo*. Si se encuentra 'Content-Type: application/json', el cuerpo se guarda como un objeto; de lo contrario, se almacena como una cadena, por ejemplo: 
HTTP proxy	Puede especificar un HTTP proxy utilizando el formato `[protocol://][username[:password]@]proxy.example.com[:port]`. El prefijo opcional `protocol://` puede utilizarse para especificar protocolos de proxy alternativos (por ejemplo, `https`, `socks4`, `socks5`; consulte la documentación; la compatibilidad con el prefijo de protocolo se añadió en cURL 7.21.7). Si no se especifica ningún protocolo, el proxy se tratará como un HTTP proxy. Si especifica un protocolo incorrecto, la conexión fallará y el item dejará de ser compatible. De forma predeterminada, se utilizará el puerto 1080. Si se especifica, el proxy sobrescribirá las variables de entorno relacionadas con el proxy, como `http_proxy` y `HTTPS_PROXY`. Si no se especifica, el proxy no sobrescribirá las variables de entorno relacionadas con el proxy. El valor introducido se transmite "tal cual"; no se realiza ninguna comprobación de validez. Tenga en cuenta que con HTTP proxy solo se admite la autenticación simple. Macros compatibles: `{HOST.IP}`, `{HOST.CONN}`, `{HOST.DNS}`, `{HOST.HOST}`, `{HOST.NAME}`, `{HOST.PORT}`, `{ITEM.ID}`, `{ITEM.KEY}`, `{ITEM.KEY.ORIG}`, macros de usuario y macros de descubrimiento de bajo nivel. Esto establece la opción cURL `CURLOPT_PROXY`.
Autenticación HTTP	Seleccione la opción de autenticación: **Ninguna**: no se utiliza autenticación; **Basic**: se utiliza autenticación básica; **NTLM**: se utiliza autenticación NTLM (Windows NT LAN Manager); **Kerberos**: se utiliza autenticación Kerberos (consulte también: Configuración de Kerberos con Zabbix); **Digest**: se utiliza autenticación Digest. Esto establece la opción cURL `CURLOPT_HTTPAUTH`.
Nombre de usuario	Introduzca el nombre de usuario (hasta 255 caracteres). Este campo está disponible si *Autenticación HTTP* está establecida en *Basic*, *NTLM*, *Kerberos* o *Digest*. Se admiten macros de usuario y macros de descubrimiento de bajo nivel.
Contraseña	Introduzca la contraseña del usuario (hasta 255 caracteres). Este campo está disponible si *Autenticación HTTP* está establecida en *Basic*, *NTLM*, *Kerberos* o *Digest*. Se admiten macros de usuario y macros de descubrimiento de bajo nivel.
Verificar par SSL	Marque la casilla para verificar el certificado SSL del web server. El certificado del server se verifica con los certificados de la autoridad certificadora (CA) de confianza de la ubicación de CA de todo el sistema. Puede sobrescribir la ubicación de los archivos de CA mediante el parámetro de configuración `SSLCALocation` del server o proxy de Zabbix. Esto establece la opción cURL `CURLOPT_SSL_VERIFYPEER`.
Verificar host SSL	Marque la casilla para verificar que el campo *Common Name* o el campo *Subject Alternate Name* del certificado del web server coincida. Esto establece la opción cURL `CURLOPT_SSL_VERIFYHOST`.

Parámetro	Descripción
Archivo de certificado SSL	Nombre del archivo de certificado SSL utilizado para la autenticación del cliente. El archivo de certificado debe estar en formato PEM[1]. Si el archivo de certificado también contiene la clave privada, deje vacío el campo *Archivo de clave SSL*. Si la clave está cifrada, especifique la contraseña en el campo *Contraseña de la clave SSL*. El directorio que contiene este archivo se especifica mediante el parámetro de configuración `SSLCertLocation` del server o proxy de Zabbix. Macros compatibles: `{HOST.IP}`, `{HOST.CONN}`, `{HOST.DNS}`, `{HOST.HOST}`, `{HOST.NAME}`, `{HOST.PORT}`, `{ITEM.ID}`, `{ITEM.KEY}`, `{ITEM.KEY.ORIG}`, macros de usuario y macros de descubrimiento de bajo nivel. Esto establece la opción cURL `CURLOPT_SSLCERT`.
Archivo de clave SSL	Nombre del archivo de clave privada SSL utilizado para la autenticación del cliente. El archivo de clave privada debe estar en formato PEM[1]. El directorio que contiene este archivo se especifica mediante el parámetro de configuración `SSLKeyLocation` del server o proxy de Zabbix. Macros compatibles: `{HOST.IP}`, `{HOST.CONN}`, `{HOST.DNS}`, `{HOST.HOST}`, `{HOST.NAME}`, `{HOST.PORT}`, `{ITEM.ID}`, `{ITEM.KEY}`, `{ITEM.KEY.ORIG}`, macros de usuario y macros de descubrimiento de bajo nivel. Esto establece la opción cURL `CURLOPT_SSLKEY`.
Contraseña de la clave SSL	Contraseña del archivo de clave privada SSL. Macros compatibles: macros de usuario y macros de descubrimiento de bajo nivel. Esto establece la opción cURL `CURLOPT_KEYPASSWD`.
Tiempo de espera	Zabbix no dedicará más tiempo del establecido a procesar la URL (1-600 segundos). En realidad, este parámetro define el tiempo máximo para establecer una conexión con la URL y el tiempo máximo para realizar una solicitud HTTP. Por lo tanto, Zabbix no dedicará más de 2 x *Tiempo de espera* segundos a una comprobación. Esto establece la opción cURL `CURLOPT_TIMEOUT`. Para obtener más información sobre el parámetro *Tiempo de espera*, consulte los atributos generales de los items.
Habilitar trapping	Con esta casilla marcada, el item también funcionará como un item trapper y aceptará datos enviados al server o proxy de Zabbix mediante la utilidad Zabbix sender o el protocolo de Zabbix sender, o aceptará datos enviados al server de Zabbix mediante el método de API `history.push`. Para obtener más información sobre el envío de datos, consulte: Items trapper.
Hosts permitidos	Visible solo si la casilla *Habilitar trapping* está marcada. Lista de direcciones IP separadas por comas (opcionalmente en notación CIDR) o nombres DNS. El valor predeterminado es la macro de usuario `{$TRAPPER.ALLOWED_HOSTS}`. De forma predeterminada, `{$TRAPPER.ALLOWED_HOSTS}` es `127.0.0.1,::1`. Puede introducir las direcciones directamente en el campo o cambiar el valor de la macro. Si se deja vacío, se denegarán las conexiones entrantes de todos los hosts. Si se especifica, las conexiones entrantes solo se aceptarán de los hosts incluidos aquí. Si la compatibilidad con IPv6 está habilitada, `127.0.0.1`, `::127.0.0.1` y `::ffff:127.0.0.1` se tratarán de la misma forma, y `::/0` permitirá cualquier dirección IPv4 o IPv6. `0.0.0.0/0` puede utilizarse para permitir cualquier dirección IPv4. Tenga en cuenta que las "direcciones IPv6 compatibles con IPv4" (prefijo `0000::/96`) son compatibles, pero están obsoletas según RFC4291. Ejemplo: `127.0.0.1`, `192.168.1.0/24`, `192.168.3.1-255`, `192.168.1-10.1-255`, `::1,2001:db8::/32`, `mysqlserver1`, `zabbix.example.com`, `{HOST.HOST}` En este campo se permiten espacios y macros de usuario. Macros de host permitidas en este campo: `{HOST.HOST}`, `{HOST.NAME}`, `{HOST.IP}`, `{HOST.DNS}`, `{HOST.CONN}`.

Note:
Si el campo *HTTP proxy* se deja vacío, otra forma de utilizar un HTTP proxy consiste en establecer variables de entorno relacionadas con el proxy.
Para HTTP, establezca la variable de entorno `http_proxy` para el usuario del server de Zabbix. Por ejemplo:
 `http_proxy=http://proxy_ip:proxy_port`.
Para HTTPS, establezca la variable de entorno `HTTPS_PROXY`. Por ejemplo:
 `HTTPS_PROXY=http://proxy_ip:proxy_port`.
Hay más información disponible ejecutando un comando de shell: *# man curl*.

Attention:
[1] Zabbix admite únicamente archivos de certificado y de clave privada en formato PEM. Si los datos del certificado y de la clave privada están en un archivo con formato PKCS #12 (normalmente con la extensión `*.p12` o `*.pfx`), puede generar el archivo PEM a partir de él mediante los siguientes comandos:

```
openssl pkcs12 -in ssl-cert.p12 -clcerts -nokeys -out ssl-cert.pem
openssl pkcs12 -in ssl-cert.p12 -nocerts -nodes  -out ssl-cert.key
```

Ejemplos

Ejemplo 1

Envía solicitudes GET simples para recuperar datos de servicios como Elasticsearch:

- Crea un item GET con la URL: `localhost:9200/?pretty`
- Observa la respuesta:

```
{
  "name" : "YQ2VAY-",
  "cluster_name" : "elasticsearch",
  "cluster_uuid" : "kH4CYqh5QfqgeTsjh2F9zg",
  "version" : {
    "number" : "6.1.3",
    "build_hash" : "af51318",
    "build_date" : "2018-01-26T18:22:55.523Z",
    "build_snapshot" : false,
    "lucene_version" : "7.1.0",
    "minimum_wire_compatibility_version" : "5.6.0",
    "minimum_index_compatibility_version" : "5.0.0"
  },
  "tagline" : "Ya sabes, para buscar"
}
```

- Ahora extrae el número de versión usando un paso de preprocesamiento JSONPath: `$.version.number`

Ejemplo 2

Envía solicitudes POST simples para recuperar datos de servicios como Elasticsearch:

- Crea un item POST con la URL: `http://localhost:9200/str/values/_search?scroll=10s`
- Configura el siguiente cuerpo POST para obtener la carga del procesador (promedio de 1 min por núcleo)

```
{
    "query": {
        "bool": {
            "must": [{
                "match": {
                    "itemid": 28275
                }
            }],
            "filter": [{
                "range": {
                    "clock": {
                        "gt": 1517565836,
                        "lte": 1517566137
                    }
                }
            }]
        }
    }
}
```

- Recibido:

```
{
    "_scroll_id": "DnF1ZXJ5VGhlbkZldGNoBQAAAAAAAAAkFllRMlZBWS1UU1pxTmdEeGVwQjRBTFEAAAAAAAAAJRZZUTJWQVk
    "took": 18,
    "timed_out": false,
```

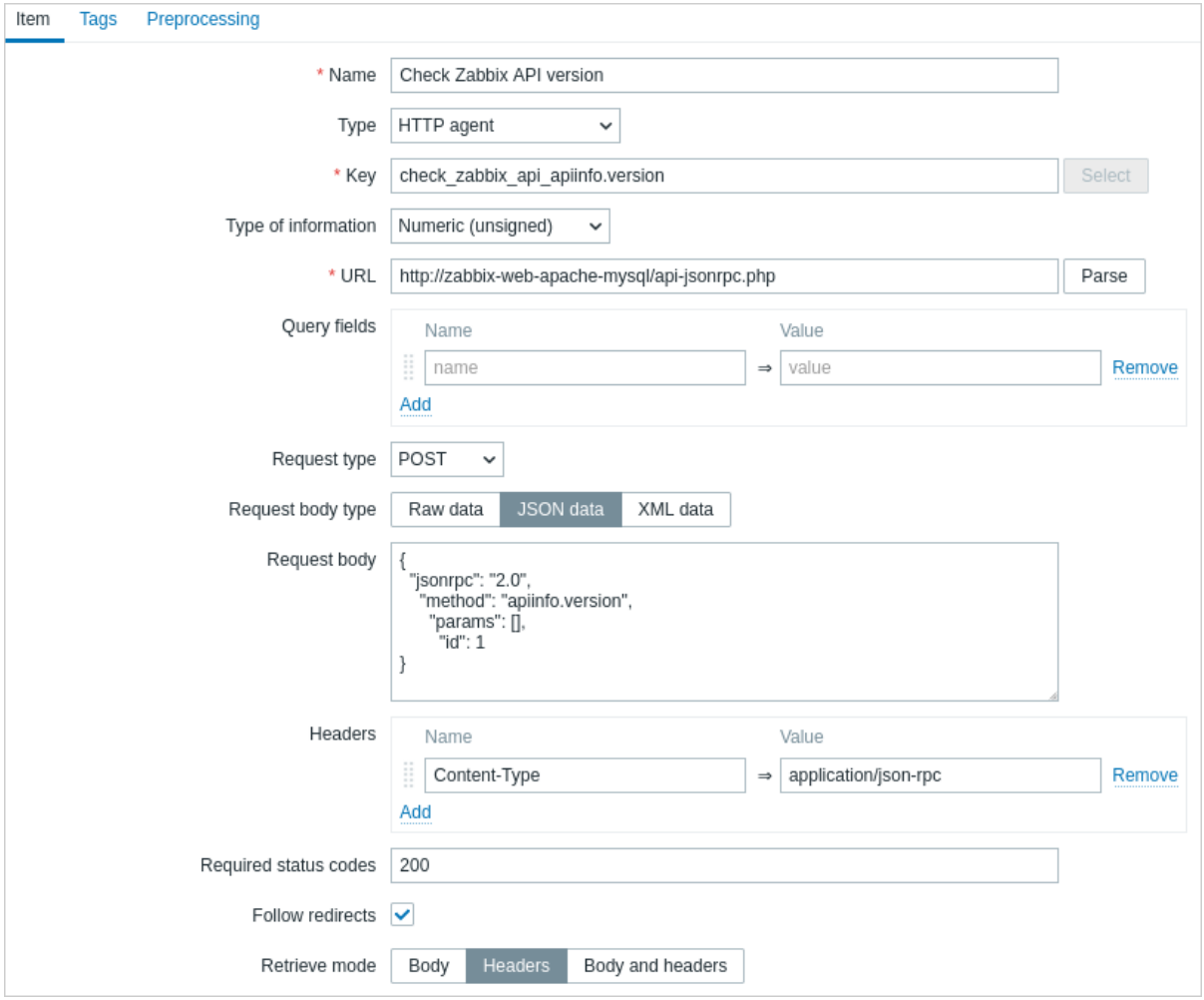
```
        "_shards": {
            "total": 5,
            "successful": 5,
            "skipped": 0,
            "failed": 0
        },
        "hits": {
            "total": 1,
            "max_score": 1.0,
            "hits": [{
                "_index": "dbl",
                "_type": "values",
                "_id": "dqX9VWEBV6sEKSMyk6sw",
                "_score": 1.0,
                "_source": {
                    "itemid": 28275,
                    "value": "0.138750",
                    "clock": 1517566136,
                    "ns": 25388713,
                    "ttl": 604800
                }
            }]
        }
    }
```


- Ahora usa un paso de preprocesamiento JSONPath para obtener el valor del item: `$.hits.hits[0]._source.value`

Ejemplo 3

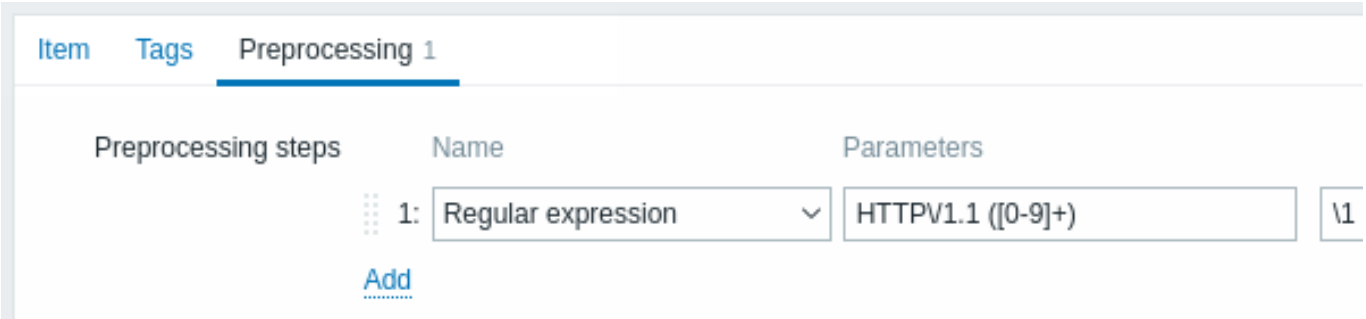
Comprobación de si la API de Zabbix está activa, usando apiinfo.version.

- Configuración del item:

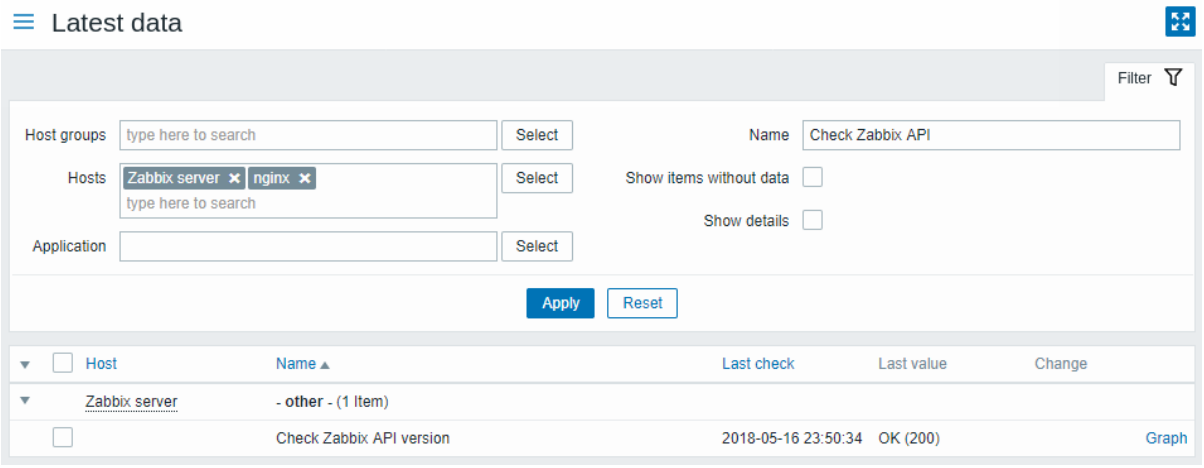


Observe el uso del método POST con datos JSON, estableciendo encabezados de solicitud y solicitando que se devuelvan solo los encabezados:

- Preprocesamiento del valor del item con expresión regular para obtener el código HTTP:



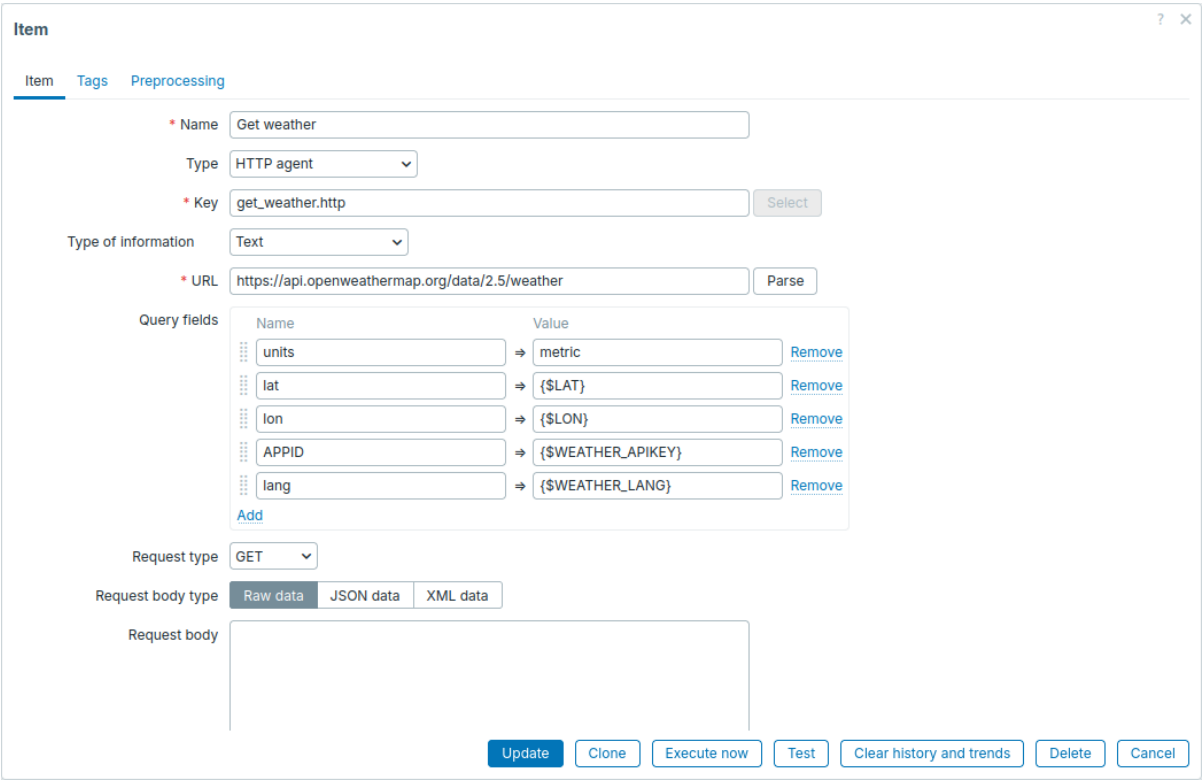
- Comprobación del resultado en *Latest data*:



Ejemplo 4

Obtención de información meteorológica mediante la conexión al servicio público de Openweathermap.

- Configure un item maestro para la recopilación masiva de datos en un único JSON:



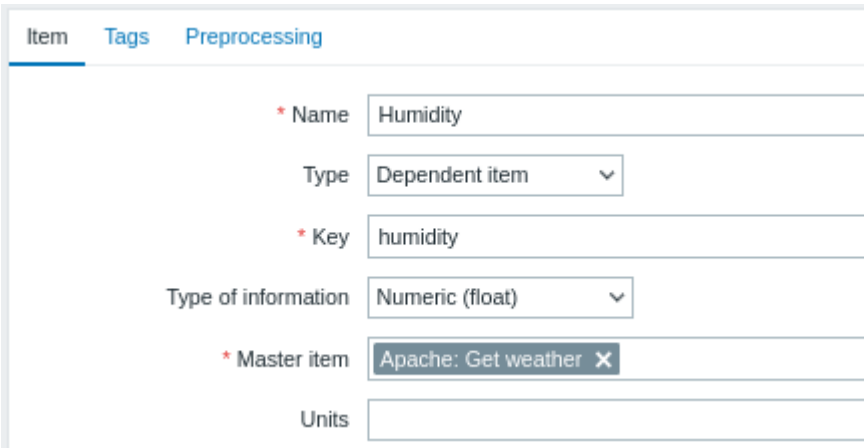
Observe el uso de macros en los campos de consulta. Consulte la API de Openweathermap para saber cómo completarlos.

Ejemplo de JSON devuelto en la respuesta al agent HTTP:

```
{
    "body": {
        "coord": {
            "lon": 40.01,
            "lat": 56.11
        },
        "weather": [{
            "id": 801,
            "main": "Clouds",
            "description": "few clouds",
            "icon": "02n"
        }],
        "base": "stations",
        "main": {
            "temp": 15.14,
            "pressure": 1012.6,
            "humidity": 66,
            "temp_min": 15.14,
            "temp_max": 15.14,
            "sea_level": 1030.91,
            "grnd_level": 1012.6
        },
        "wind": {
            "speed": 1.86,
            "deg": 246.001
        },
        "clouds": {
            "all": 20
        },
        "dt": 1526509427,
        "sys": {
            "message": 0.0035,
            "country": "RU",
            "sunrise": 1526432608,
            "sunset": 1526491828
        },
        "id": 487837,
        "name": "Stavrovo",
        "cod": 200
    }
}
```

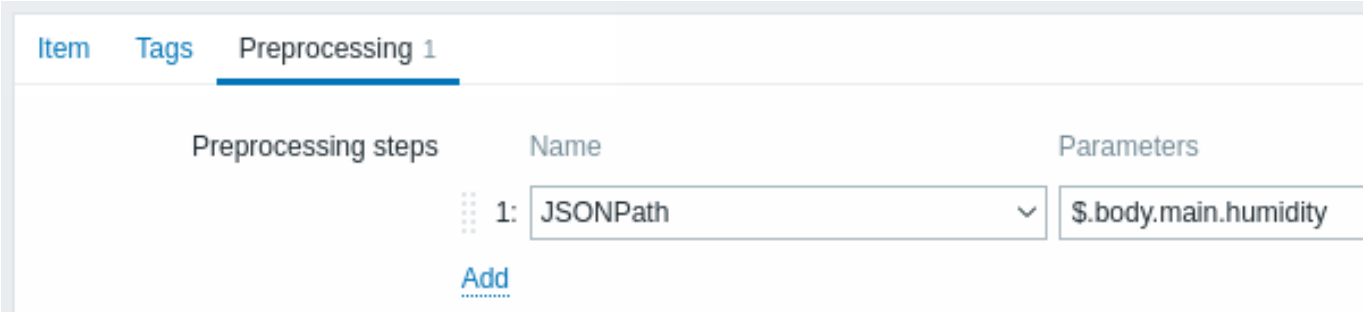

La siguiente tarea es configurar items dependientes que extraigan datos del JSON.

- Configure un item dependiente de ejemplo para la humedad:

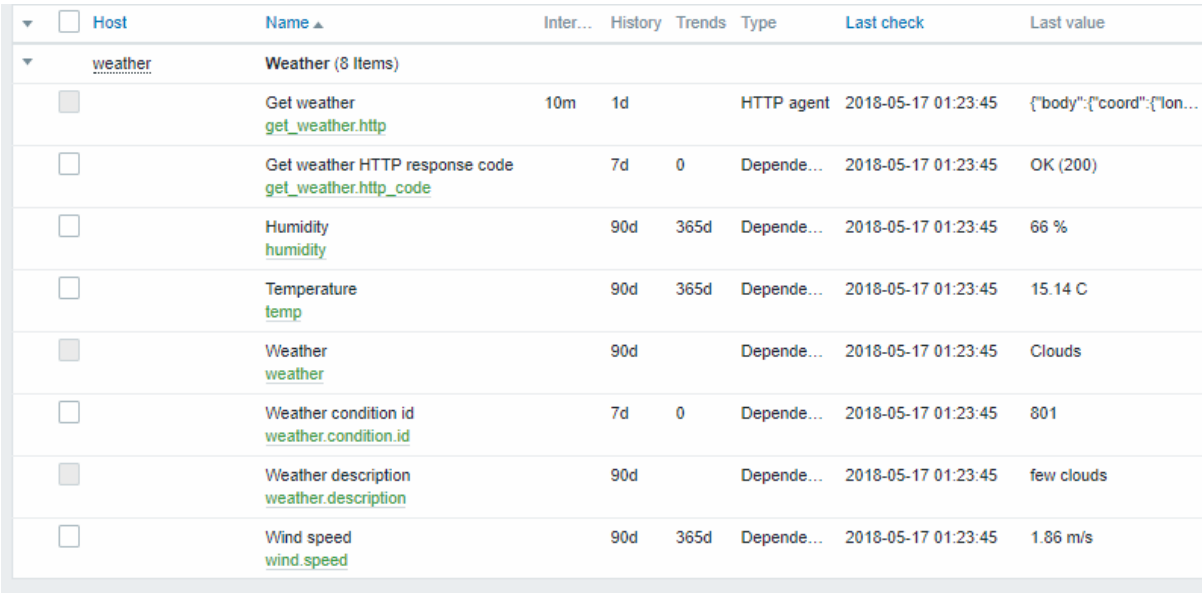


Otros métricos meteorológicos, como 'Temperature', se añaden de la misma manera.

- Ejemplo de preprocesamiento del valor de un item dependiente con JSONPath:



- Compruebe el resultado de los datos meteorológicos en *Latest data*:

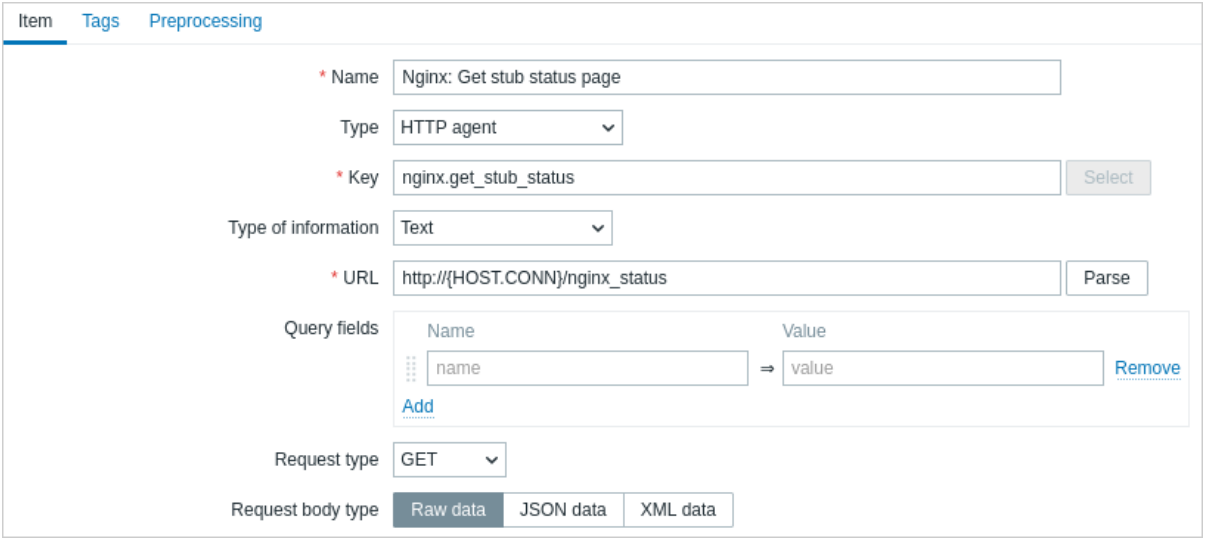


Host	Name	Inter...	History	Trends	Type	Last check	Last value
weather	**Weather** (8 Items)						
	Get weather get_weather.http	10m	1d		HTTP agent	2018-05-17 01:23:45	{"body":{"coord":{"lon...
	Get weather HTTP response code get_weather.http_code		7d	0	Depende...	2018-05-17 01:23:45	OK (200)
	Humidity humidity		90d	365d	Depende...	2018-05-17 01:23:45	66 %
	Temperature temp		90d	365d	Depende...	2018-05-17 01:23:45	15.14 C
	Weather weather		90d		Depende...	2018-05-17 01:23:45	Clouds
	Weather condition id weather.condition.id		7d	0	Depende...	2018-05-17 01:23:45	801
	Weather description weather.description		90d		Depende...	2018-05-17 01:23:45	few clouds
	Wind speed wind.speed		90d	365d	Depende...	2018-05-17 01:23:45	1.86 m/s

Ejemplo 5

Conexión a la página de estado de NGINX y obtención masiva de sus métricas.

- Configure NGINX siguiendo la guía oficial.
- Configure un item maestro para la recopilación masiva de datos:

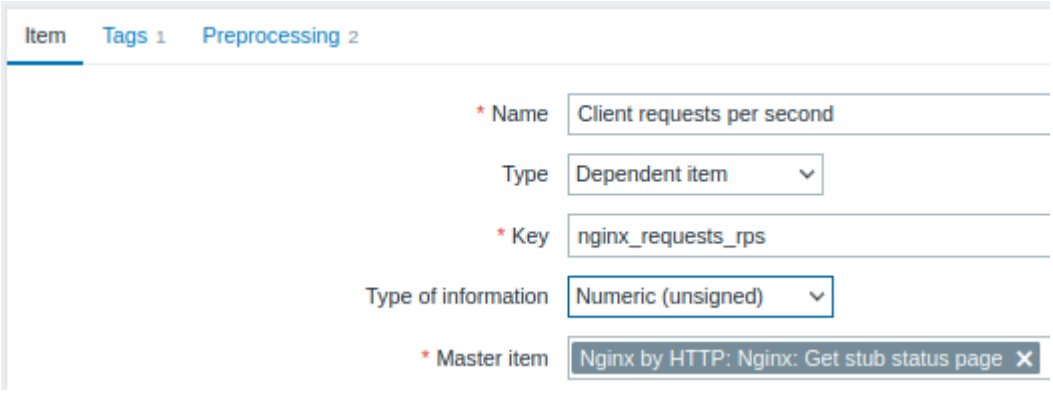


Salida de ejemplo del estado de NGINX:

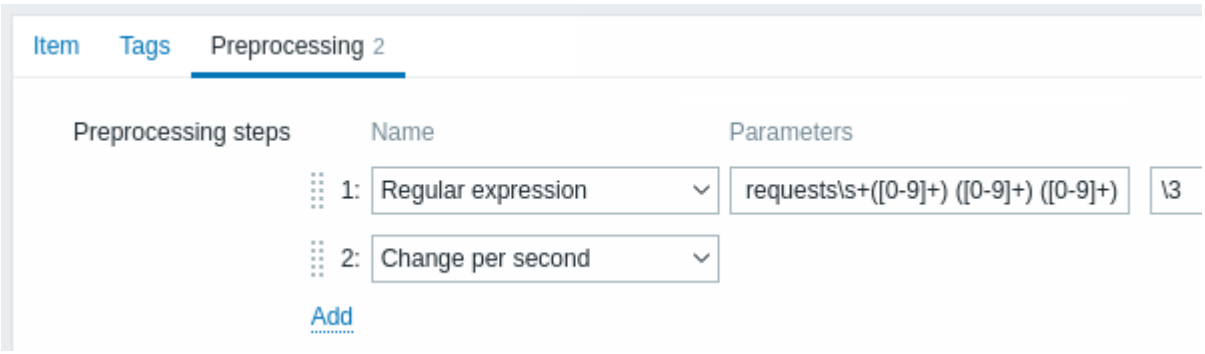
```
Active connections: 1 Active connections:
server accepts handled requests
 52 52 52
Reading: 0 Writing: 1 Waiting: 0
```

La siguiente tarea consiste en configurar los items dependientes que extraen los datos.

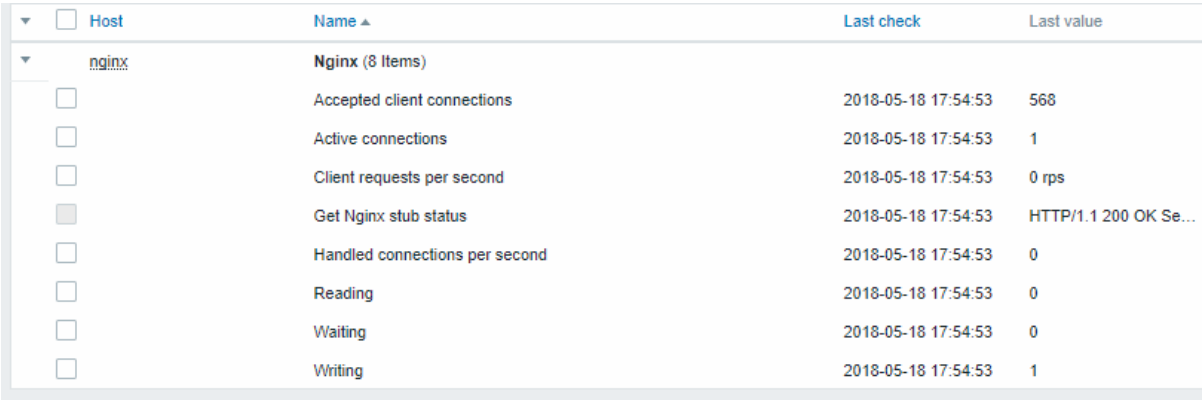
- Configure un item dependiente de ejemplo para las solicitudes por segundo:



- Preprocesamiento de ejemplo del valor de un item dependiente con la expresión regular `server accepts handled requests\s+([0-9]+) ([0-9]+) ([0-9]+)`:



- Compruebe el resultado completo del módulo stub en *Latest data*:



Comprobación de Prometheus

Descripción general

Zabbix puede consultar métricas expuestas en el formato de línea de Prometheus.

Se requieren dos pasos para comenzar a recopilar datos de Prometheus:

- un item maestro HTTP que apunte al endpoint de datos adecuado, por ejemplo `https://<prometheus host>/metrics`
- items dependientes que utilicen una opción de preprocesamiento de Prometheus para consultar los datos requeridos a partir de las métricas recopiladas por el item maestro

Hay dos opciones de preprocesamiento de datos de Prometheus:

- *Patrón de Prometheus* - se usa en items normales para consultar datos de Prometheus
- *Prometheus a JSON* - se usa en items normales y para el descubrimiento de bajo nivel. En este caso, los datos de Prometheus consultados se devuelven en formato JSON.

Procesamiento masivo

El procesamiento masivo es compatible con los items dependientes. Para habilitar el almacenamiento en caché y la indexación, el preprocesamiento de *Prometheus pattern* debe ser el **primer** paso de preprocesamiento. Cuando *Prometheus pattern* es el primer paso de preprocesamiento, los datos de Prometheus analizados se almacenan en caché y se indexan mediante la primera

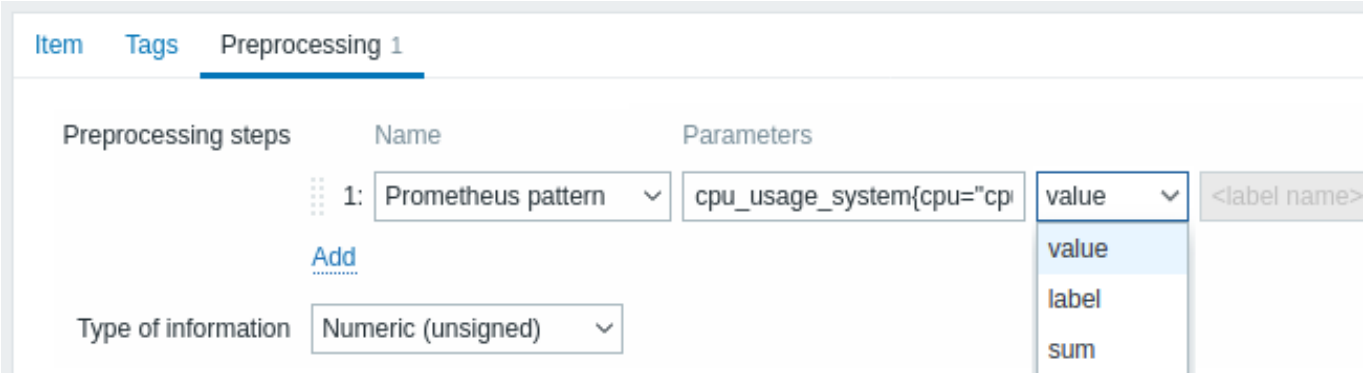
condición `<label>==<value>` en el paso de preprocesamiento de *Prometheus pattern*. Esta caché se reutiliza al procesar otros items dependientes en este lote. Para un rendimiento óptimo, la primera etiqueta debe ser la que tenga más valores diferentes.

Si hay otro preprocesamiento que deba realizarse antes del primer paso, debe trasladarse al item maestro o a un nuevo item dependiente que se usaría como item maestro para los items dependientes.

Configuración

Siempre que tenga configurado el item maestro HTTP, debe crear un item dependiente que utilice un paso de preprocesamiento de Prometheus:

1. Introduzca los parámetros generales del item dependiente en el formulario de configuración.
2. Vaya a la pestaña *Preprocessing*.
3. Seleccione una opción de preprocesamiento de Prometheus (*Prometheus pattern* o *Prometheus to JSON*).



Los siguientes parámetros son específicos de la opción de preprocesamiento *Prometheus pattern*:

Parámetro	Descripción	Ejemplos
Pattern	Para definir el patrón de datos requerido, puede utilizar un lenguaje de consulta similar al lenguaje de consulta de Prometheus (consulte la tabla comparativa), por ejemplo: <metric name> - seleccionar por nombre de métrica {__name__="<metric name>"} - seleccionar por nombre de métrica {__name__=~"<regex>"} - seleccionar por nombre de métrica que coincida con una expresión regular {<label name>="<label value>",...} - seleccionar por nombre de etiqueta {<label name>=~"<regex>",...} - seleccionar por nombre de etiqueta que coincida con una expresión regular {__name__=~".*"}==<value> - seleccionar por valor de métrica O una combinación de los elementos anteriores: <metric name>{<label1 name>="<label1 value>",<label2 name>=~"<regex>",...}==<value> El valor de la etiqueta puede ser cualquier secuencia de caracteres UTF-8, pero los caracteres de barra inversa, comillas dobles y salto de línea deben escaparse como \\, \" y \n, respectivamente; los demás caracteres no deben escaparse.	*wmi_os_physical_memory_free_bytes* *cpu_usage_system{cpu="cpu-total"}* *cpu_usage_system{cpu=".*"}* *cpu_usage_system{cpu="cpu-total",host="* *wmi_service_state{name="dhcp"}==1* *wmi_os_timezone{timezone=~".*"}==1*

Parámetro	Descripción	Ejemplos
Result processing	Especifique si desea devolver el valor, la etiqueta o aplicar la función correspondiente (si el patrón coincide con varias líneas y es necesario agregar el resultado): **value** - devolver el valor de la métrica (error si coinciden varias líneas) **label** - devolver el valor de la etiqueta especificada en el campo *Label* (error si coinciden varias métricas) **sum** - devolver la suma de los valores **min** - devolver el valor mínimo **max** - devolver el valor máximo **avg** - devolver el valor promedio **count** - devolver el número de valores Este campo solo está disponible para la opción *Prometheus pattern*.	Consulte también los ejemplos de uso de los parámetros a continuación.
Output	Defina el nombre de la etiqueta (opcional). En este caso, se devuelve el valor correspondiente al nombre de la etiqueta. Este campo solo está disponible para la opción *Prometheus pattern*, si se selecciona 'Label' en el campo *Result processing*.	

Ejemplos de uso de los parámetros

1. El caso de uso más común es devolver el **valor**. Para devolver el valor de `/var/db` de:

   `node_disk_usage_bytes{path="/var/cache"} 2.1766144e+09<br> node_disk_usage_bytes{path="/var/db"} 20480<br> node_disk_usage_bytes{path="/var/dpkg"} 8192<br> node_disk_usage_bytes{path="/var/empty"} 4096`

utilice los siguientes parámetros:

- *Pattern* - `node_disk_usage_bytes{path="/var/db"}`
- *Result processing* - seleccione 'value'

2. También puede interesarle el valor **promedio** de todos los parámetros `node_disk_usage_bytes`:

- *Pattern* - `node_disk_usage_bytes`
- *Result processing* - seleccione 'avg'

3. Aunque Prometheus solo admite datos numéricos, normalmente se utiliza una solución alternativa para devolver también la descripción textual correspondiente. Esto puede lograrse mediante un filtro y especificando la etiqueta. Por lo tanto, para devolver el valor de la etiqueta 'color' de:

   `elasticsearch_cluster_health_status{cluster="elasticsearch",color="green"} 1<br> elasticsearch_cluster 0<br> elasticsearch_cluster_health_status{cluster="elasticsearch",color="yellow"} 0`

utilice los siguientes parámetros:

- *Pattern* - `elasticsearch_cluster_health_status {cluster="elasticsearch"} == 1`
- *Result processing* - seleccione 'label'
- *Label* - especifique 'color'

El filtro (basado en el valor numérico '1') coincidirá con la fila correspondiente, mientras que la etiqueta devolverá la descripción del estado de salud (actualmente 'green', pero potencialmente también 'red' o 'yellow').

Prometheus a JSON

Los datos de Prometheus se pueden usar para el descubrimiento de bajo nivel. En este caso se necesitan datos en formato JSON, y la opción de preprocesamiento *Prometheus to JSON* devolverá exactamente eso.

Para más detalles, consulte Discovery using Prometheus data.

Comparación de lenguajes de consulta

La siguiente tabla enumera las diferencias y similitudes entre PromQL y el lenguaje de consulta de preprocesamiento de Zabbix para Prometheus.

Selector de vector instantáneo de PromQL		Lenguaje de preprocesamiento de Zabbix para Prometheus
Diferencias		
Destino de la consulta	server de Prometheus	Texto plano en formato de exposición de Prometheus
Devuelve	Vector instantáneo	Valor de métrica o etiqueta (patrón de Prometheus) Array de métricas para un único valor en JSON (Prometheus a JSON)
Operadores de coincidencia de etiquetas	**=**, **!=**, **=~**, **!~**	**=**, **!=**, **=~**, **!~**
Expresión regular usada en la coincidencia de nombre de etiqueta o métrica	RE2	PCRE
Operadores de comparación	Ver lista	Solo se admite **==** (igual) para el filtrado de valores
Similitudes		
Selección por nombre de métrica que coincide exactamente con una cadena	<nombre de métrica> o {__name__="<nombre de métrica>"}	<nombre de métrica> o {__name__="<nombre de métrica>"}
Selección por nombre de métrica que coincide con una expresión regular	{__name__=~"<regex>"}	{__name__=~"<regex>"}
Selección por valor de <nombre de etiqueta> que coincide exactamente con una cadena	{<nombre de etiqueta>="<valor de etiqueta>",...}	{<nombre de etiqueta>="<valor de etiqueta>",...}
Selección por valor de <nombre de etiqueta> que coincide con una expresión regular	{<nombre de etiqueta>=~"<regex>",...}	{<nombre de etiqueta>=~"<regex>",...}
Selección por valor que coincide exactamente con una cadena	{__name__=~".*"} == <valor>	{__name__=~".*"} == <valor>

IPMI agent

Descripción general

Puede supervisar el estado y la disponibilidad de los dispositivos con Intelligent Platform Management Interface (IPMI) en Zabbix. Para realizar comprobaciones de IPMI, el server de Zabbix debe estar inicialmente configurado con compatibilidad con IPMI.

IPMI es una interfaz estandarizada para la administración remota de sistemas informáticos "lights-out" o "out-of-band". Permite supervisar directamente el estado del hardware desde las denominadas tarjetas de administración "out-of-band", independientemente del sistema operativo o de si la máquina está encendida.

La supervisión de IPMI de Zabbix solo funciona con dispositivos compatibles con IPMI (HP iLO, DELL DRAC, IBM RSA, Sun SSP, etc.).

Un proceso de administración de IPMI programa las comprobaciones de IPMI mediante los sondeos de IPMI. Un host siempre es sondeado por un único sondeo de IPMI a la vez, lo que reduce el número de conexiones abiertas a los controladores BMC. Por lo tanto, es seguro aumentar el número de sondeos de IPMI sin preocuparse por sobrecargar el controlador BMC. El proceso de administración de IPMI se inicia automáticamente cuando se inicia al menos un sondeo de IPMI.

Consulte también los problemas conocidos relacionados con las comprobaciones de IPMI.

Configuración

Configuración de equipos

Se debe configurar un equipo para procesar comprobaciones IPMI. Se debe añadir una interfaz IPMI, con los respectivos números de IP y puerto, y se deben definir los parámetros de autenticación IPMI.

Consulte la configuración de equipos para más detalles.

Configuración del server

De forma predeterminada, el server de Zabbix no está configurado para iniciar ningún poller de IPMI, por lo que cualquier item de IPMI agregado no funcionará. Para cambiar esto, abra el archivo de configuración del server de Zabbix (`zabbix_server.conf`) como root y busque la siguiente línea:

```
# StartIPMIPollers=0
```

Quite el comentario y establezca el número de pollers en, por ejemplo, 3, de modo que quede así:

```
StartIPMIPollers=3
```

Guarde el archivo y reinicie zabbix_server después.

Configuración de la métrica

Al configurar una métrica a nivel de equipo:

- Seleccione 'IPMI agent' como el *Tipo*
- Introduzca una clave de métrica que sea única dentro del equipo (por ejemplo, ipmi.fan.rpm)
- Para *Interfaz de equipo* seleccione la interfaz IPMI relevante (IP y puerto). Tenga en cuenta que debe existir una interfaz IPMI en el equipo.
- Especifique el *Sensor IPMI* (por ejemplo, 'FAN MOD 1A RPM' en Dell Poweredge) del que se obtendrá la métrica. Por defecto, se debe especificar el ID del sensor. También es posible utilizar prefijos antes del valor:
  - `id:` - para especificar el ID del sensor;
  - `name:` - para especificar el nombre completo del sensor. Esto puede ser útil en situaciones en las que los sensores solo pueden distinguirse especificando el nombre completo.
- Seleccione el tipo de información correspondiente ('Numérico (coma flotante)' en este caso; para sensores discretos - 'Numérico (sin signo)'), las unidades (probablemente 'rpm') y cualquier otro atributo requerido de la métrica

Comprobaciones soportadas

El agente IPMI soporta el item incorporado **ipmi.get**, que devuelve información relacionada con el sensor IPMI y puede utilizarse para el descubrimiento de sensores IPMI.
 Valor devuelto: *objeto JSON*

Tiempo de espera y finalización de sesión

Los tiempos de espera de mensajes IPMI y los recuentos de reintentos se definen en la biblioteca OpenIPMI. Debido al diseño actual de OpenIPMI, no es posible hacer que estos valores sean configurables en Zabbix, ni a nivel de interfaz ni de métrica.

El tiempo de espera por inactividad de la sesión IPMI para LAN es de 60 +/-3 segundos. Actualmente no es posible implementar el envío periódico del comando Activate Session con OpenIPMI. Si no hay comprobaciones de métricas IPMI desde Zabbix a un BMC en particular durante más tiempo que el tiempo de espera de sesión configurado en el BMC, entonces la siguiente comprobación IPMI después de que expire el tiempo de espera fallará debido a tiempos de espera individuales de mensajes, reintentos o error de recepción. Después de eso, se abre una nueva sesión y se inicia un nuevo escaneo completo del BMC. Puede abrirse un nuevo puerto UDP para gestionar la nueva sesión. La inactividad se define por la ausencia tanto de solicitudes salientes como de respuestas entrantes. Si desea evitar escaneos innecesarios del BMC, se recomienda establecer el intervalo de sondeo de la métrica IPMI por debajo del tiempo de espera por inactividad de la sesión IPMI configurado en el BMC.

Notas sobre sensores discretos IPMI

Para encontrar sensores en un host, inicie el servidor Zabbix con **DebugLevel=4** habilitado. Espere unos minutos y busque los registros de descubrimiento de sensores en el archivo de registro del servidor Zabbix:

```
$ grep 'Added sensor' zabbix_server.log
8358:20130318:111122.170 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:7 id:'CATERR' reading_type:
8358:20130318:111122.170 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:15 id:'CPU Therm Trip' read
8358:20130318:111122.171 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:17 id:'System Event Log' re
8358:20130318:111122.171 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:17 id:'PhysicalSecurity' re
8358:20130318:111122.171 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:14 id:'IPMI Watchdog' readi
8358:20130318:111122.171 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:16 id:'Power Unit Stat' rea
8358:20130318:111122.171 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:16 id:'P1 Therm Ctrl %' rea
8358:20130318:111122.172 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:16 id:'P1 Therm Margin' rea
8358:20130318:111122.172 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:13 id:'System Fan 2' readin
8358:20130318:111122.172 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:13 id:'System Fan 3' readin
8358:20130318:111122.172 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:14 id:'P1 Mem Margin' readi
8358:20130318:111122.172 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:17 id:'Front Panel Temp' re
8358:20130318:111122.173 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:15 id:'Baseboard Temp' read
8358:20130318:111122.173 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:9 id:'BB +5.0V' reading_typ
8358:20130318:111122.173 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:14 id:'BB +3.3V STBY' readi
8358:20130318:111122.173 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:9 id:'BB +3.3V' reading_typ
8358:20130318:111122.173 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:17 id:'BB +1.5V P1 DDR3' re
8358:20130318:111122.173 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:17 id:'BB +1.1V P1 Vccp' re
8358:20130318:111122.174 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:14 id:'BB +1.05V PCH' readi
```

Para descifrar los tipos y estados de los sensores IPMI, está disponible una copia de las especificaciones IPMI 2.0 (tenga en cuenta que no se planean más actualizaciones de la especificación IPMI).

El primer parámetro con el que comenzar es "reading_type". Utilice la "Tabla 42-1, Rangos de códigos de tipo de evento/lectura" de las especificaciones para descifrar el código "reading_type". La mayoría de los sensores en nuestro ejemplo tienen "read-

ing_type:0x1", lo que significa sensor de "umbral". La "Tabla 42-3, Códigos de tipo de sensor" muestra que "type:0x1" significa sensor de temperatura, "type:0x2" - sensor de voltaje, "type:0x4" - ventilador, etc. Los sensores de umbral a veces se denominan sensores "analógicos" ya que miden parámetros continuos como temperatura, voltaje, revoluciones por minuto.

Otro ejemplo: un sensor con "reading_type:0x3". La "Tabla 42-1, Rangos de códigos de tipo de evento/lectura" dice que los códigos de tipo de lectura 02h-0Ch significan sensor "Genérico Discreto". Los sensores discretos tienen hasta 15 estados posibles (en otras palabras, hasta 15 bits significativos). Por ejemplo, para el sensor 'CATERR' con "type:0x7", la "Tabla 42-3, Códigos de tipo de sensor" muestra que este tipo significa "Procesador" y el significado de los bits individuales es: 00h (el bit menos significativo) - IERR, 01h - Thermal Trip, etc.

Hay algunos sensores con "reading_type:0x6f" en nuestro ejemplo. Para estos sensores, la "Tabla 42-1, Rangos de códigos de tipo de evento/lectura" aconseja utilizar la "Tabla 42-3, Códigos de tipo de sensor" para descifrar el significado de los bits. Por ejemplo, el sensor 'Power Unit Stat' tiene el tipo "type:0x9", que significa "Unidad de energía". El desplazamiento 00h significa "PowerOff/Power Down". En otras palabras, si el bit menos significativo es 1, entonces el servidor está apagado. Para probar este bit, se puede utilizar la función `bitand` con la máscara '1'. La expresión del trigger podría ser como

```
bitand(last(/www.example.com/Power Unit Stat,#1),1)=1
```

para advertir sobre un apagado del servidor.

Notas sobre los nombres de sensores discretos en OpenIPMI-2.0.16, 2.0.17, 2.0.18 y 2.0.19

Los nombres de los sensores discretos en OpenIPMI-2.0.16, 2.0.17 y 2.0.18 a menudo tienen un "0" adicional (u otro dígito o letra) añadido al final. Por ejemplo, mientras que `ipmitool` y OpenIPMI-2.0.19 muestran los nombres de los sensores como "`PhysicalSecurity`" o "`CATERR`", en OpenIPMI-2.0.16, 2.0.17 y 2.0.18 los nombres son "`PhysicalSecurity0`" o "`CATERR0`", respectivamente.

Al configurar una métrica IPMI con el servidor Zabbix usando OpenIPMI-2.0.16, 2.0.17 y 2.0.18, utilice estos nombres terminados en "0" en el campo *Sensor IPMI* de las métricas del agente IPMI. Cuando su servidor Zabbix se actualice a una nueva distribución de Linux, que utilice OpenIPMI-2.0.19 (o posterior), las métricas con estos sensores discretos IPMI pasarán a estar "NO SOPORTADO". Debe cambiar sus nombres de *Sensor IPMI* (eliminar el '0' al final) y esperar un tiempo antes de que vuelvan a estar "Habilitado".

Notas sobre la disponibilidad simultánea de sensores de umbral y discretos

Algunos agentes IPMI proporcionan tanto un sensor de umbral como un sensor discreto bajo el mismo nombre. Siempre se da preferencia al sensor de umbral.

Notas sobre la terminación de la conexión

Si no se realizan comprobaciones IPMI (por cualquier motivo: todas las métricas IPMI del equipo deshabilitadas/no soportadas, equipo deshabilitado/eliminado, equipo en mantenimiento, etc.) la conexión IPMI será terminada por el servidor o proxy de Zabbix en 3 a 4 horas, dependiendo del momento en que se inició el servidor/proxy de Zabbix.

SSH agent

Resumen

Las comprobaciones SSH se realizan como supervisión sin agent. No se necesita el agent de Zabbix para las comprobaciones SSH.

Para realizar comprobaciones SSH, el server de Zabbix debe estar inicialmente configurado con compatibilidad SSH2 (libssh o libssh2). Consulte también: Requisitos.

> **Attention:**
> A partir de RHEL 8, solo se admite libssh. Para otras distribuciones, se recomienda libssh en lugar de libssh2.

Configuración

Autenticación por frase de contraseña

Las comprobaciones SSH proporcionan dos métodos de autenticación: un par usuario/contraseña y basado en archivo de clave.

Si no tiene intención de usar claves, no se requiere configuración adicional, además de vincular libssh o libssh2 a Zabbix, si está compilando desde el código fuente.

Autenticación con archivo de clave

Para usar autenticación basada en clave para los items SSH, se requieren ciertos cambios en la configuración del server.

Abra el archivo de configuración del server de Zabbix (`zabbix_server.conf`) como `root` y busque la siguiente línea:

```
##### SSHKeyLocation=
```

Descoméntela y establezca la ruta completa a la carpeta donde se ubicarán las claves pública y privada:

```
SSHKeyLocation=/home/zabbix/.ssh
```

Guarde el archivo y reinicie el server de Zabbix después.

La ruta `/home/zabbix` aquí es el directorio personal de la cuenta de usuario *zabbix*, y `.ssh` es un directorio donde, de forma predeterminada, se generarán las claves pública y privada mediante un comando ssh-keygen dentro del directorio personal.

Por lo general, los paquetes de instalación del server de Zabbix de distintas distribuciones de SO crean la cuenta de usuario *zabbix* con un directorio personal en otra ubicación, por ejemplo, `/var/lib/zabbix` (como ocurre con las cuentas del sistema).

Antes de generar las claves, podría reasignar el directorio personal a `/home/zabbix`, de modo que coincida con el parámetro de configuración `SSHKeyLocation` del server de Zabbix mencionado anteriormente.

> **Note:**
> Los siguientes pasos se pueden omitir si la cuenta *zabbix* se ha agregado manualmente según la sección de instalación. En tal caso, lo más probable es que el directorio personal de la cuenta *zabbix* ya sea `/home/zabbix`.

Para cambiar el directorio personal de la cuenta de usuario *zabbix*, deben detenerse todos los procesos en ejecución que lo estén utilizando:

```
systemctl stop zabbix-agent
systemctl stop zabbix-server
```

Para cambiar la ubicación del directorio personal con un intento de moverlo (si existe), debe ejecutarse el siguiente comando:

```
usermod -m -d /home/zabbix zabbix
```

También es posible que un directorio personal no existiera en la ubicación anterior, por lo que debe crearse en la nueva ubicación. Una forma segura de hacerlo es:

```
test -d /home/zabbix || mkdir /home/zabbix
```

Para asegurarse de que todo sea seguro, se pueden ejecutar comandos adicionales para establecer permisos en el directorio personal:

```
chown zabbix:zabbix /home/zabbix
chmod 700 /home/zabbix
```

Los procesos detenidos anteriormente ya pueden iniciarse de nuevo:

```
systemctl start zabbix-agent
systemctl start zabbix-server
```

Ahora, los pasos para generar las claves pública y privada pueden realizarse con los siguientes comandos (para mayor claridad, los indicadores del comando se han comentado):

```
sudo -u zabbix ssh-keygen -t rsa
##### Generating public/private rsa key pair.
##### Enter file in which to save the key (/home/zabbix/.ssh/id_rsa):
/home/zabbix/.ssh/id_rsa
##### Enter passphrase (empty for no passphrase):
<Leave empty>
##### Enter same passphrase again:
<Leave empty>
##### Your identification has been saved in /home/zabbix/.ssh/id_rsa.
##### Your public key has been saved in /home/zabbix/.ssh/id_rsa.pub.
##### The key fingerprint is:
##### 90:af:e4:c7:e3:f0:2e:5a:8d:ab:48:a2:0c:92:30:b9 zabbix@it0
##### The key's randomart image is:
##### +--[ RSA 2048]----+
##### |                 |
##### |       .         |
##### |      o          |
##### | .     o         |
##### |+     . S        |
##### |.+   o =         |
##### |E .   * =        |
```

```
##### |=o . ..* .        |
##### |... oo.o+         |
##### +-----------------+
```

Note:
Las claves pública y privada (`id_rsa.pub` e `id_rsa`) se han generado de forma predeterminada en el directorio `/home/zabbix/.ssh`, que corresponde al parámetro de configuración `SSHKeyLocation` del server de Zabbix.

Attention:
Los tipos de clave distintos de "rsa" pueden ser compatibles con la herramienta ssh-keygen y con los servidores SSH, pero es posible que no sean compatibles con libssh2, que es la biblioteca utilizada por Zabbix.

Formulario de configuración de shell

Este paso debe realizarse solo una vez para cada host que vaya a ser monitorizado mediante comprobaciones SSH.

Usando los siguientes comandos, el archivo de clave **pública** puede instalarse en un host remoto *10.10.10.10*, de modo que las comprobaciones SSH puedan realizarse con una cuenta *root* (para una mejor legibilidad, los prompts de comando se han comentado):

```
sudo -u zabbix ssh-copy-id root@10.10.10.10
##### The authenticity of host '10.10.10.10 (10.10.10.10)' can't be established.
##### RSA key fingerprint is 38:ba:f2:a4:b5:d9:8f:52:00:09:f7:1f:75:cc:0b:46.
##### Are you sure you want to continue connecting (yes/no)?
yes
##### Warning: Permanently added '10.10.10.10' (RSA) to the list of known hosts.
##### root@10.10.10.10's password:
<Enter root password>
##### Now try logging into the machine, with "ssh 'root@10.10.10.10'",
##### and check to make sure that only the key(s) you wanted were added.
```

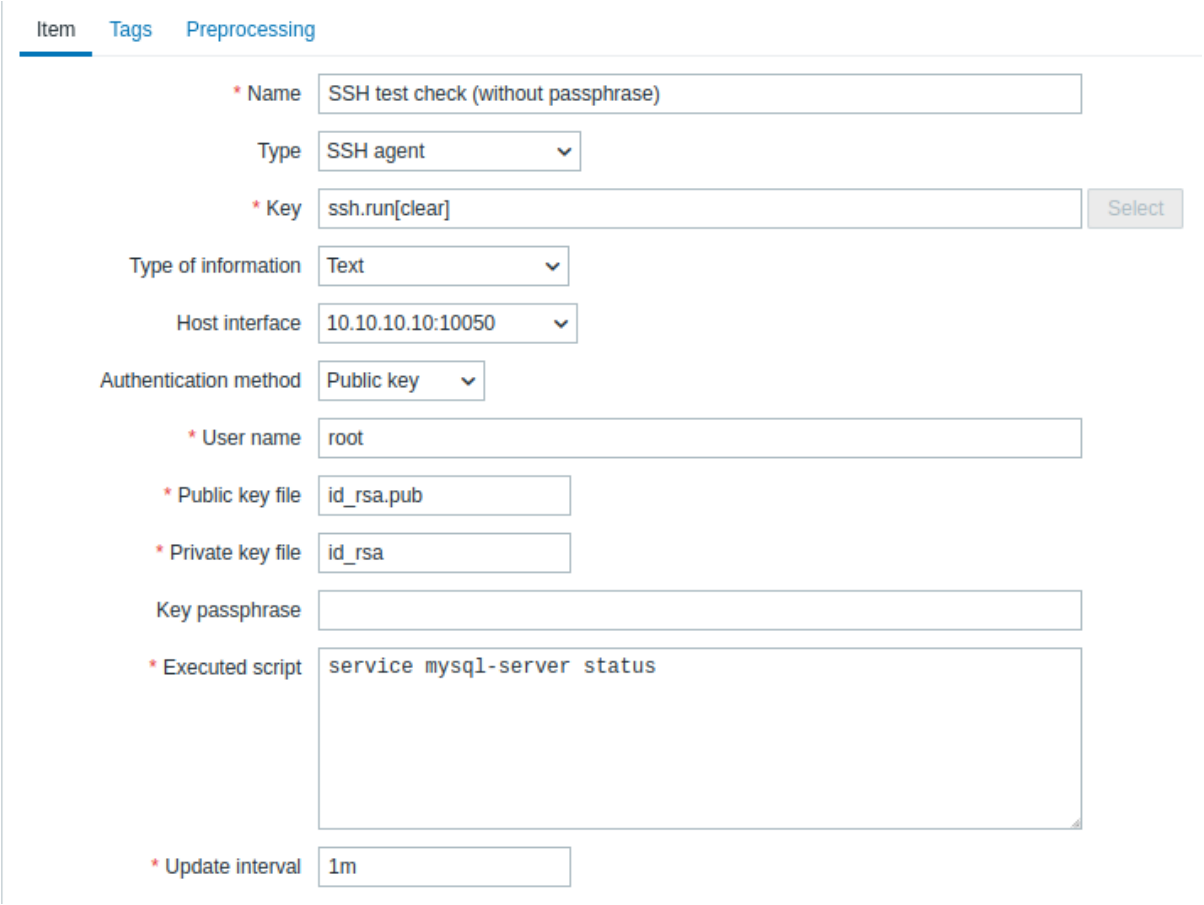
Ahora es posible comprobar el inicio de sesión SSH usando la clave privada predeterminada (`/home/zabbix/.ssh/id_rsa`) para la cuenta de usuario *zabbix*:

```
sudo -u zabbix ssh root@10.10.10.10
```

Si el inicio de sesión tiene éxito, entonces la parte de configuración en la shell ha finalizado y la sesión SSH remota puede cerrarse.

Configuración del item

Los comandos que se ejecutarán realmente deben colocarse en el campo *Executed script* de la configuración del item. Se pueden ejecutar varios comandos uno tras otro colocándolos en una nueva línea. En este caso, los valores devueltos también se formatearán en varias líneas.



Todos los campos de entrada obligatorios están marcados con un asterisco rojo.

Los campos que requieren información específica para los items SSH son:

Parámetro	Descripción	Comentarios
Type	Seleccione **SSH agent** aquí.	

Parámetro	Descripción	Comentarios

Key	Clave única (por host) del item con el formato `ssh.run[unique short description,<ip>,<port>,<encoding>,<ssh options>,<subsystem>]`	**unique short description** es obligatoria y debe ser única para cada item SSH por host.
		
		El puerto predeterminado es 22, no el puerto especificado en la interfaz a la que está asignado este item.
		
		ssh options permite pasar opciones SSH adicionales con el formato `key1=value1;key2=value2,value3`. Se pueden pasar varios valores para una clave separados por comas (en este caso, el parámetro debe estar entre comillas); se pueden pasar varias claves de opciones separadas por punto y coma.
		
		Se admiten las siguientes claves de opciones: `KexAlgorithms`, `HostkeyAlgorithms`, `Ciphers`, `MACs`, `PubkeyAcceptedKeyTypes`. La compatibilidad con las claves y los valores de las opciones depende de la biblioteca SSH (por ejemplo, `PubkeyAcceptedKeyTypes` solo es compatible con libssh); si una opción no es compatible, se devolverá un error y el item pasará a no compatible.
		
		Tenga en cuenta que no se admiten el signo "+" para añadir configuraciones de cifrado ni "!" para deshabilitar configuraciones de cifrado específicas (como en GnuTLS y OpenSSL).
		
		Ejemplos:
		=>
		`ssh.run[KexAlgorithms,127.0.0.1,,,Ciphers=aes128`
		=>
		`ssh.run[KexAlgorithms,,,,"KexAlgorithms=diffie-h`
		=>
		`ssh.run[PubkeyAcceptedKeyTypes,127.0.0.1,,,Pubke`
		
		subsystem permite pasar un subsistema SSH, limitando la conexión SSH a las operaciones específicas permitidas por el subsistema (por ejemplo, transferencias de archivos mediante SFTP o administración de dispositivos de red mediante NETCONF). Tenga en cuenta que el uso de un subsistema también puede requerir una sintaxis de script específica en el parámetro *Executed script*.
		
		Ejemplos:
		=>
		`ssh.run[SFTPBackup,192.0.2.18,,,,sftp]`
		=>
		`ssh.run[Cisco1234,192.0.2.18,,,,netconf]`

Authentication method	Uno de los métodos "Password" o "Public key".	
User name	Nombre de usuario (hasta 255 caracteres) para autenticarse en el host remoto. Obligatorio.	
Public key file	Nombre del archivo de clave pública si *Authentication method* es "Public key". Obligatorio.	Ejemplo: *id_rsa.pub*: nombre de archivo de clave pública predeterminado generado por el comando ssh-keygen.

Parámetro	Descripción	Comentarios
Private key file	Nombre del archivo de clave privada si *Authentication method* es "Public key". Obligatorio.	Ejemplo: *id_rsa*: nombre de archivo de clave privada predeterminado.
Password o *Key passphrase*	Contraseña (hasta 255 caracteres) para autenticarse o frase de contraseña **si** se utilizó para la clave privada.	Deje vacío el campo *Key passphrase* si no se utilizó ninguna frase de contraseña. Consulte también los problemas conocidos relacionados con el uso de frases de contraseña.

Executed script	Comando(s) de shell ejecutado(s) mediante una sesión SSH remota.	El valor devuelto por los comandos de shell ejecutados está limitado a 16 MB (incluidos los espacios en blanco finales, que se truncan); también se aplican los límites de la base de datos.
		
		Tenga en cuenta que la biblioteca libssh2 puede truncar los scripts ejecutables a aproximadamente 32 kB.
		
		Ejemplos:
		`date +%s`
		`systemctl status mysql-server`
		`ps auxww | grep httpd | wc -l`
		
		Ejemplo (para el subsistema NETCONF):
		`<rpc>`
		`<get-software-information/>`
		`</rpc>`
		`]]>]]>`
		`<rpc>`
		`<close-session/>`
		`</rpc>`
		`]]>]]>`

Agente Telnet

Descripción general

Las comprobaciones Telnet se realizan como monitorización sin agente. No se necesita el agente Zabbix para las comprobaciones Telnet.

Campos configurables

El/los comando(s) que se ejecutarán deben colocarse en el campo **Script ejecutado** en la configuración del item.
 Se pueden ejecutar varios comandos uno tras otro colocándolos en una nueva línea. En este caso, el valor devuelto también se formateará como multilínea.

Caracteres soportados con los que puede terminar el prompt de la shell:

- $
- #
- >
- %

Note:
Una línea de prompt de telnet que termine con uno de estos caracteres se eliminará del valor devuelto, pero solo para el primer comando en la lista de comandos, es decir, solo al inicio de la sesión telnet.

Clave	Descripción
telnet.run[<descripción corta única>,<ip>,<puerto>,<codificación>]	Ejecuta un comando en un dispositivo remoto usando una conexión telnet

Attention:
Si una comprobación telnet devuelve un valor con caracteres no ASCII y en una codificación que no sea UTF8, entonces el parámetro *<codificación>* de la clave debe especificarse correctamente. Consulte la página sobre la codificación de los valores devueltos para más detalles.

agente JMX

Descripción general

La monitorización JMX puede utilizarse para monitorizar contadores JMX de una aplicación Java.

La monitorización JMX tiene soporte nativo en Zabbix en forma de un demonio de Zabbix llamado "Java gateway de Zabbix".

Para recuperar el valor de un contador JMX particular en un equipo, el servidor Zabbix consulta al **Java gateway de Zabbix**, que a su vez utiliza la API de gestión JMX para consultar la aplicación de interés de forma remota.

Para más detalles y configuración, consulte la sección Java gateway de Zabbix.

Warning:
La comunicación entre el Java gateway y la aplicación JMX monitorizada no debe estar bloqueada por el firewall.

Habilitar la monitorización remota JMX para una aplicación Java

Una aplicación Java no necesita ningún software adicional instalado, pero debe iniciarse con las opciones de línea de comandos especificadas a continuación para tener soporte para la monitorización remota JMX.

Como mínimo, si solo desea comenzar monitorizando una aplicación Java simple en un equipo local sin seguridad aplicada, iníciela con estas opciones:

```
java \
-Dcom.sun.management.jmxremote \
-Dcom.sun.management.jmxremote.port=12345 \
-Dcom.sun.management.jmxremote.authenticate=false \
-Dcom.sun.management.jmxremote.ssl=false \
-Dcom.sun.management.jmxremote.registry.ssl=false \
-jar /ruta/a/su/aplicacion.jar
```

Esto hace que Java escuche conexiones JMX entrantes en el puerto 12345, solo desde el equipo local, y le indica que no requiera autenticación ni SSL.

Si desea permitir conexiones en otra interfaz, establezca el parámetro -Djava.rmi.server.hostname en la IP de esa interfaz.

Si desea ser más estricto con la seguridad, hay muchas otras opciones de Java disponibles para usted. Por ejemplo, el siguiente ejemplo inicia la aplicación con un conjunto de opciones más versátil y la abre a una red más amplia, no solo al equipo local.

```
java \
-Djava.rmi.server.hostname=192.168.3.14 \
-Dcom.sun.management.jmxremote \
-Dcom.sun.management.jmxremote.port=12345 \
-Dcom.sun.management.jmxremote.authenticate=true \
-Dcom.sun.management.jmxremote.password.file=/etc/java-6-openjdk/management/jmxremote.password \
-Dcom.sun.management.jmxremote.access.file=/etc/java-6-openjdk/management/jmxremote.access \
-Dcom.sun.management.jmxremote.ssl=true \
-Dcom.sun.management.jmxremote.registry.ssl=true \
-Djavax.net.ssl.keyStore=$YOUR_KEY_STORE \
-Djavax.net.ssl.keyStorePassword=$YOUR_KEY_STORE_PASSWORD \
-Djavax.net.ssl.trustStore=$YOUR_TRUST_STORE \
-Djavax.net.ssl.trustStorePassword=$YOUR_TRUST_STORE_PASSWORD \
-Dcom.sun.management.jmxremote.ssl.need.client.auth=true \
-jar /ruta/a/su/aplicacion.jar
```

La mayoría (si no todas) de estas configuraciones pueden especificarse en `$JRE/lib/management/management.properties` (o dondequiera que se encuentre ese archivo en su sistema).

Tenga en cuenta que si desea usar SSL, debe modificar el script startup.sh agregando las opciones `-Djavax.net.ssl.*` al Java gateway, para que sepa dónde encontrar los almacenes de claves y de confianza.

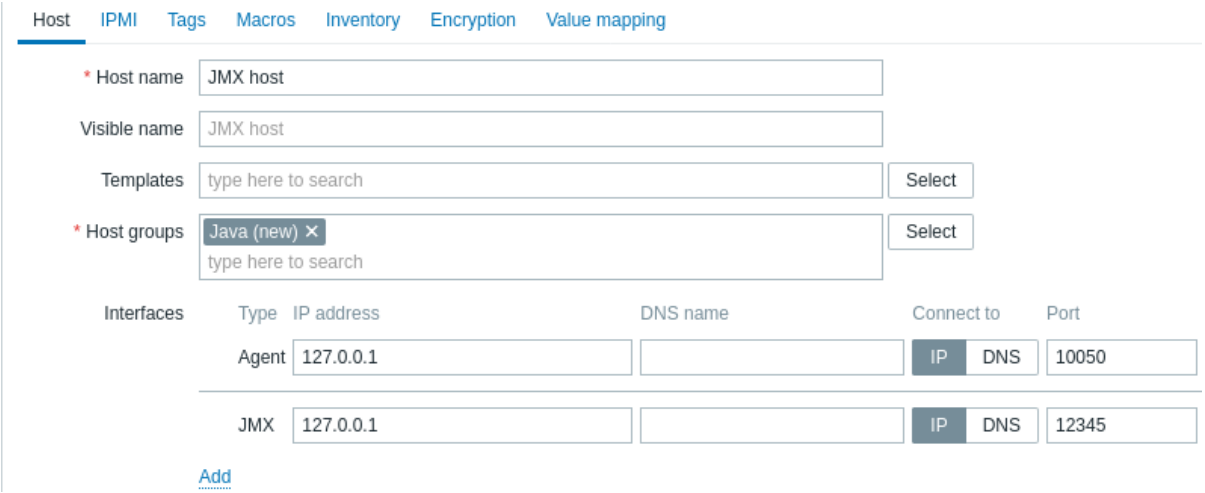
Consulte Monitorización y gestión usando JMX para una descripción detallada.

Configuración de interfaces y métricas JMX en el frontend de Zabbix

Con el Java gateway en ejecución, el servidor sabiendo dónde encontrarlo y una aplicación Java iniciada con soporte para la monitorización remota de JMX, es momento de configurar las interfaces y métricas en la interfaz gráfica de Zabbix.

Configuración de la interfaz JMX

Comience creando una interfaz de tipo JMX en el host de interés.

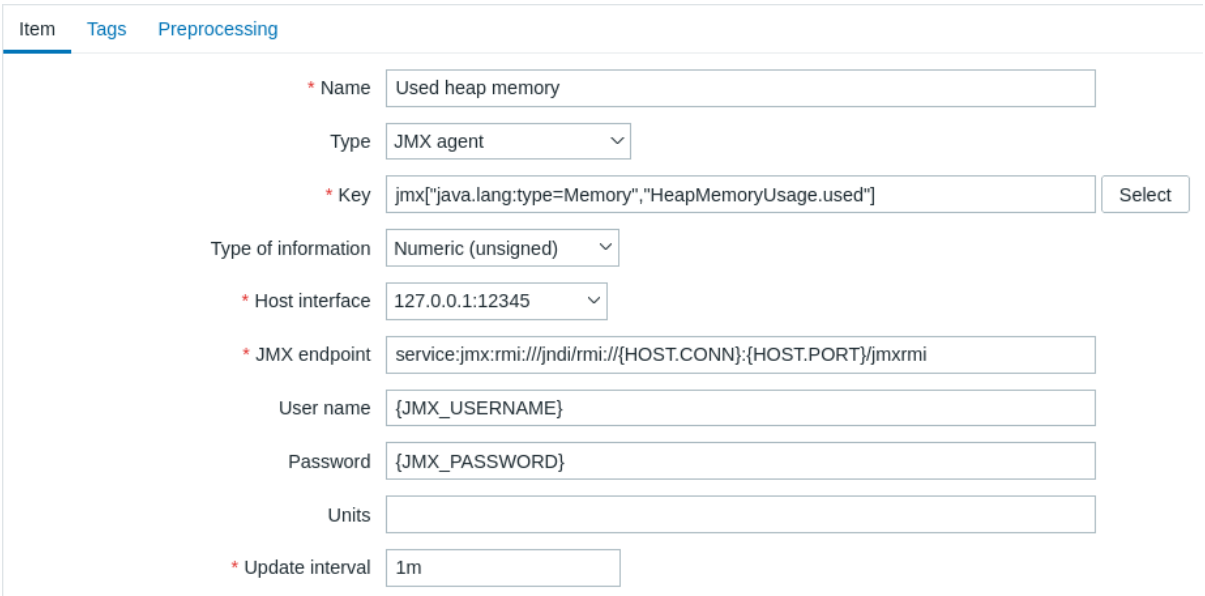


Todos los campos obligatorios están marcados con un asterisco rojo.

Añadir item de agent JMX

Para cada contador JMX que le interese, añada un item de **JMX agent** vinculado a esa interfaz.

La clave en la captura de pantalla siguiente es `jmx["java.lang:type=Memory","HeapMemoryUsage.used"]`.



Todos los campos de entrada obligatorios están marcados con un asterisco rojo.

Los campos que requieren información específica para los items JMX son:

Type	Establezca aquí **JMX agent**.
Key	La clave del item `jmx[]` contiene tres parámetros: **object name** - el nombre de objeto de un MBean **attribute name** - un nombre de atributo de MBean con nombres de campos de datos compuestos opcionales separados por puntos **unique short description** - una descripción única que permite varios items JMX con el mismo nombre de objeto y nombre de atributo en el host (opcional) Consulte más abajo para obtener más detalles sobre las claves de item JMX. Puede descubrir MBeans y atributos de MBean mediante un item `jmx.discovery[]` de low-level discovery.

JMX endpoint	Puede especificar un JMX endpoint personalizado. Asegúrese de que los parámetros de conexión del JMX endpoint coincidan con la interfaz JMX. Esto se puede lograr usando macros `{HOST.*}` como en el JMX endpoint predeterminado. Se admiten las macros `{HOST.*}` y las macros de usuario.
User name	Especifique el nombre de usuario (hasta 255 caracteres), si ha configurado autenticación en su aplicación Java. Se admiten macros de usuario.
Password	Especifique la contraseña (hasta 255 caracteres), si ha configurado autenticación en su aplicación Java. Se admiten macros de usuario.

Si desea supervisar un contador Boolean que sea ”true” o ”false”, especifique el tipo de información como ”Numeric (unsigned)” y seleccione el paso de preprocesamiento ”Boolean to decimal” en la pestaña Preprocessing. Server almacenará los valores Boolean como 1 o 0, respectivamente.

Claves de métricas JMX en detalle

Atributos simples

Un nombre de objeto MBean no es más que una cadena que usted define en su aplicación Java. Un nombre de atributo, por otro lado, puede ser más complejo. En caso de que un atributo devuelva un tipo de dato primitivo (un entero, una cadena, etc.), no hay de qué preocuparse, la clave se verá así:

```
jmx[com.example:Type=Hello,weight]
```

En este ejemplo, el nombre del objeto es ”com.example:Type=Hello”, el nombre del atributo es ”weight”, y el tipo de valor devuelto probablemente debería ser ”Numérico (flotante)”.

Atributos que devuelven datos compuestos

Se vuelve más complicado cuando su atributo devuelve datos compuestos. Por ejemplo: el nombre de su atributo es ”apple” y devuelve un hash que representa sus parámetros, como ”weight”, ”color”, etc. Su clave podría verse así:

```
jmx[com.example:Type=Hello,apple.weight]
```

Así es como se separan el nombre del atributo y la clave del hash, utilizando un punto. De la misma manera, si un atributo devuelve datos compuestos anidados, las partes se separan por un punto:

```
jmx[com.example:Type=Hello,fruits.apple.weight]
```

Atributos que devuelven datos tabulares

Los atributos de datos tabulares consisten en uno o varios atributos compuestos. Si dicho atributo se especifica en el parámetro de nombre de atributo, entonces el valor de esta métrica devolverá la estructura completa del atributo en formato JSON. Los valores de los elementos individuales dentro del atributo de datos tabulares pueden recuperarse utilizando preprocesamiento.

Ejemplo de atributo de datos tabulares:

```
jmx[com.example:type=Hello,foodinfo]
```

Valor de la métrica:

```
[
  {
    "a": "manzana",
    "b": "plátano",
    "c": "cereza"
  },
  {
    "a": "patata",
    "b": "lechuga",
    "c": "cebolla"
  }
]
```

Problema con los puntos

Hasta aquí todo bien. Pero, ¿qué pasa si un nombre de atributo o una clave de hash contiene el símbolo de punto? Aquí tienes un ejemplo:

```
jmx[com.example:Type=Hello,all.fruits.apple.weight]
```

Eso es un problema. ¿Cómo decirle a Zabbix que el nombre del atributo es "all.fruits", y no solo "all"? ¿Cómo distinguir un punto que forma parte del nombre de un punto que separa un nombre de atributo y claves de hash?

Esto es posible, todo lo que necesitas hacer es escapar los puntos que forman parte del nombre con una barra invertida:

```
jmx[com.example:Type=Hello,all\.fruits.apple.weight]
```

De la misma manera, si tu clave de hash contiene un punto, escápalo:

```
jmx[com.example:Type=Hello,all\.fruits.apple.total\.weight]
```

Otros problemas

Un carácter de barra invertida en un nombre de atributo debe escaparse:

```
jmx[com.example:type=Hello,c:\\documents]
```

Para manejar cualquier otro carácter especial en la clave del item JMX, consulte la sección sobre el formato de la clave del item section.

En realidad, eso es todo. ¡Feliz monitorización JMX!

Tipos de datos no primitivos

Es posible trabajar con MBeans personalizados que devuelven tipos de datos no primitivos, que sobrescriben el método **toString()**.

Uso de un endpoint personalizado con JBoss EAP 6.4

Los endpoints personalizados permiten trabajar con distintos protocolos de transporte además del RMI predeterminado.

Para ilustrar esta posibilidad, intentemos configurar la monitorización de JBoss EAP 6.4 como ejemplo. Primero, hagamos algunas suposiciones:

- Ya ha instalado Zabbix Java gateway. Si no es así, puede hacerlo de acuerdo con la documentación.
- Zabbix server y Java gateway están instalados con el prefijo `/usr/local/`
- JBoss ya está instalado en `/opt/jboss-eap-6.4/` y se está ejecutando en modo standalone
- Supondremos que todos estos componentes funcionan en el mismo host
- El firewall y SELinux están deshabilitados (o configurados en consecuencia)

Hagamos algunos ajustes simples en `zabbix_server.conf`:

```
JavaGateway=127.0.0.1
StartJavaPollers=5
```

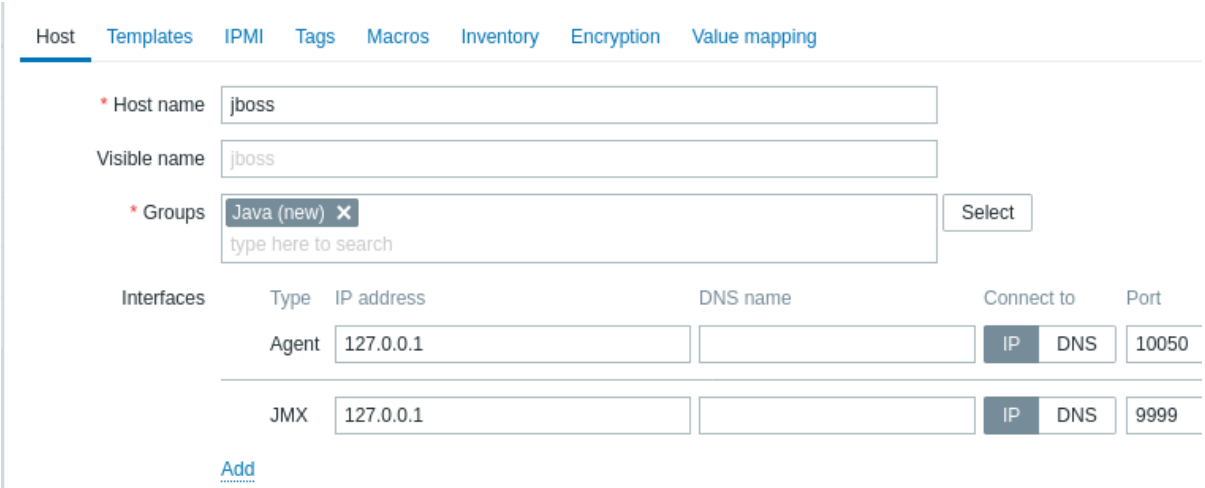
Y en el archivo de configuración `zabbix_java/settings.sh` (o `zabbix_java_gateway.conf`):

```
START_POLLERS=5
```

Compruebe que JBoss escucha en su puerto de administración estándar:

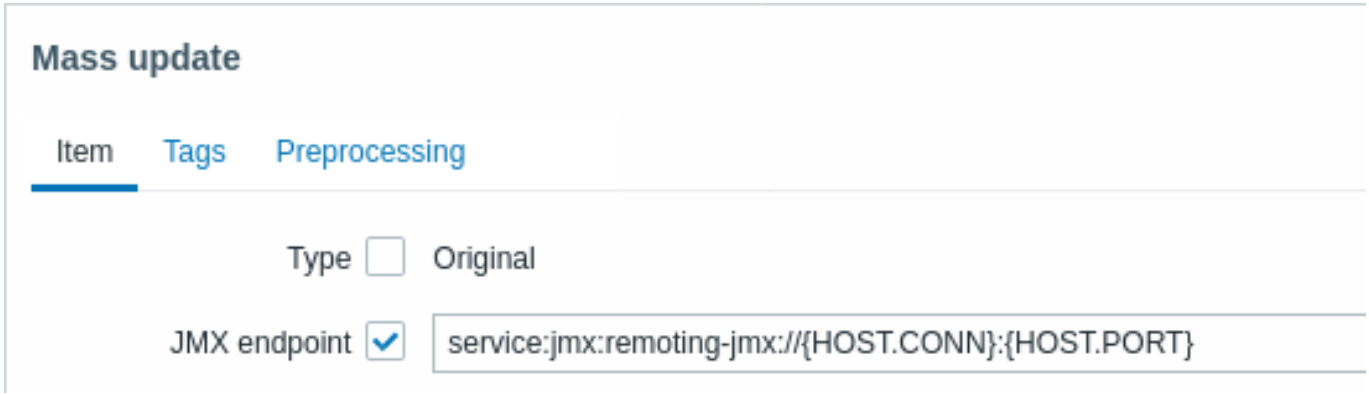
```
$ netstat -natp | grep 9999
tcp        0      0 127.0.0.1:9999          0.0.0.0:*               LISTEN      10148/java
```

Ahora vamos a crear un host con interfaz JMX 127.0.0.1:9999 en Zabbix.



Como sabemos que esta versión de JBoss usa el protocolo JBoss Remoting en lugar de RMI, podemos actualizar masivamente el parámetro del endpoint JMX para los items de nuestra template JMX en consecuencia:

```
service:jmx:remoting-jmx://{HOST.CONN}:{HOST.PORT}
```



Actualicemos la caché de configuración:

```
/usr/local/sbin/zabbix_server -R config_cache_reload
```

Tenga en cuenta que primero puede encontrarse con un error.

```
3. mc [root@centos7-dev]:/home/vagrant/zabbix-3.2.6/src/zabbix_java (ssh)
com.zabbix.gateway.ZabbixException: java.net.MalformedURLException: Unsupported protocol: remoting-jmx
        at com.zabbix.gateway.JMXItemChecker.getValues(JMXItemChecker.java:97) ~[zabbix-java-gateway-3.4.2.jar:na]
        at com.zabbix.gateway.SocketProcessor.run(SocketProcessor.java:63) ~[zabbix-java-gateway-3.4.2.jar:na]
        at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1149) [na:1.8.0_144]
        at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:624) [na:1.8.0_144]
        at java.lang.Thread.run(Thread.java:748) [na:1.8.0_144]
Caused by: java.net.MalformedURLException: Unsupported protocol: remoting-jmx
        at javax.management.remote.JMXConnectorFactory.newJMXConnector(JMXConnectorFactory.java:359) ~[na:1.8.0_144]
        at javax.management.remote.JMXConnectorFactory.connect(JMXConnectorFactory.java:269) ~[na:1.8.0_144]
        at com.zabbix.gateway.ZabbixJMXConnectorFactory$1.run(ZabbixJMXConnectorFactory.java:76) ~[zabbix-java-gatewa
-3.4.2.jar:na]
        at java.util.concurrent.Executors$RunnableAdapter.call(Executors.java:511) ~[na:1.8.0_144]
        at java.util.concurrent.FutureTask.run(FutureTask.java:266) ~[na:1.8.0_144]
        ... 3 common frames omitted
2017-11-07 13:52:12.644 [pool-1-thread-1] WARN  com.zabbix.gateway.SocketProcessor - error processing request
com.zabbix.gateway.ZabbixException: java.net.MalformedURLException: Unsupported protocol: remoting-jmx
        at com.zabbix.gateway.JMXItemChecker.getValues(JMXItemChecker.java:97) ~[zabbix-java-gateway-3.4.2.jar:na]
        at com.zabbix.gateway.SocketProcessor.run(SocketProcessor.java:63) ~[zabbix-java-gateway-3.4.2.jar:na]
        at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1149) [na:1.8.0_144]
        at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:624) [na:1.8.0_144]
        at java.lang.Thread.run(Thread.java:748) [na:1.8.0_144]
Caused by: java.net.MalformedURLException: Unsupported protocol: remoting-jmx
        at javax.management.remote.JMXConnectorFactory.newJMXConnector(JMXConnectorFactory.java:359) ~[na:1.8.0_144]
        at javax.management.remote.JMXConnectorFactory.connect(JMXConnectorFactory.java:269) ~[na:1.8.0_144]
        at com.zabbix.gateway.ZabbixJMXConnectorFactory$1.run(ZabbixJMXConnectorFactory.java:76) ~[zabbix-java-gatewa
-3.4.2.jar:na]
        at java.util.concurrent.Executors$RunnableAdapter.call(Executors.java:511) ~[na:1.8.0_144]
        at java.util.concurrent.FutureTask.run(FutureTask.java:266) ~[na:1.8.0_144]
        ... 3 common frames omitted
2017-11-07 13:52:14.889 [Thread-0] INFO  com.zabbix.gateway.JavaGateway - Zabbix Java Gateway 3.4.2 (revision 72885)
as stopped
2017-11-07 13:52:26.167 [main] INFO  com.zabbix.gateway.JavaGateway - Zabbix Java Gateway 3.4.2 (revision 72885) has
tarted
```

"Unsupported protocol: remoting-jmx" significa que Java gateway no sabe cómo trabajar con el protocolo especificado. Esto se puede corregir creando un archivo `~/needed_modules.txt` con el siguiente contenido:

```
jboss-as-remoting
jboss-logging
jboss-logmanager
jboss-marshalling
jboss-remoting
jboss-sasl
jcl-over-slf4j
jul-to-slf4j-stub
log4j-jboss-logmanager
remoting-jmx
slf4j-api
xnio-api
xnio-nio
```

y luego ejecutando el comando:

```
for i in $(cat ~/needed_modules.txt); do find /opt/jboss-eap-6.4 -iname "${i}*.jar" -exec cp '{}' /usr/loc
```

De este modo, Java gateway tendrá todos los módulos necesarios para trabajar con jmx-remoting. Solo queda reiniciar Java gateway, esperar un poco y, si ha hecho todo correctamente, verá que los datos de monitorización JMX comienzan a llegar a Zabbix (consulte también: Latest data).

Elemento calculado

Descripción general

Este tipo de item permite realizar cálculos sobre las métricas que ya existen. Un item calculado no recopila datos por sí mismo.

Por ejemplo, puede querer:

- Contar dos valores.
- Calcular el promedio de varios valores de item.
- Calcular el valor total de un grupo de items.

El valor resultante se almacena en la base de datos de Zabbix como cualquier otro item; se almacenan tanto los valores del historial como los de las tendencias, y se pueden generar gráficos.

El cálculo se basa en una fórmula. La sintaxis de la fórmula es la misma que la de las expresiones de trigger.

De hecho, es posible mantener los triggers simples colocando la lógica compleja de transformación de datos en un item calculado y, a continuación, hacer referencia al item calculado en un trigger mediante la clave del item.

Existen dos posibilidades para especificar el origen de datos de los cálculos:

1. Especificar los items en la fórmula.
2. Realizar una agregación de datos basada en un filtro que seleccione los items.

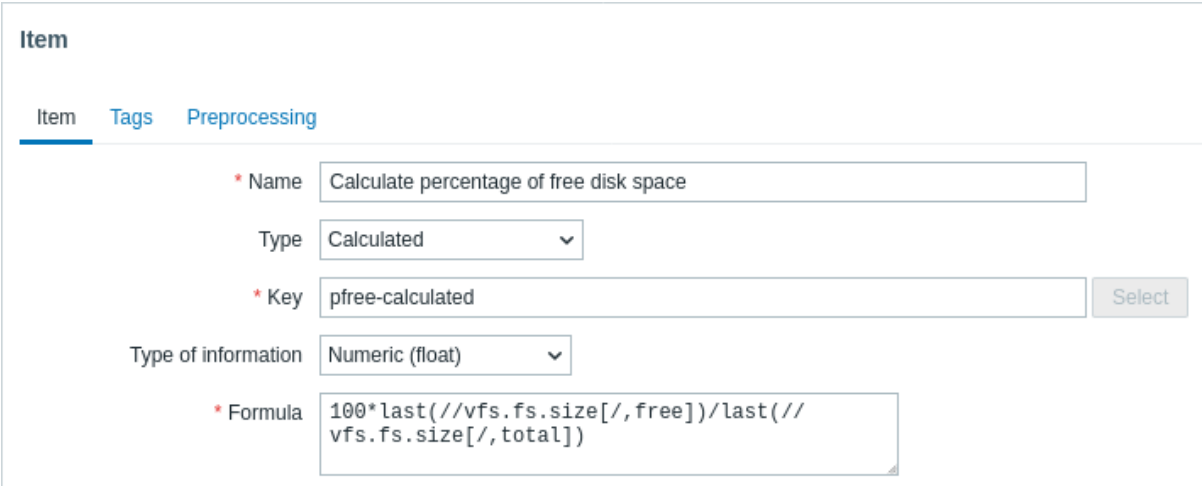
La segunda opción permite calcular promedios de grupos, totales, etc.

Todos los cálculos los realiza el Zabbix server y los gestionan los procesos `history poller`; el proxy o los agents no intervienen en absoluto. Tenga en cuenta que el uso extensivo de items calculados con períodos de tiempo prolongados puede afectar al rendimiento del server.

Tenga en cuenta que los items calculados pueden acceder al historial de otros items, incluidos aquellos a los que no tiene acceso el usuario que crea el item calculado. Si no desea este comportamiento, puede deshabilitar los items calculados estableciendo el parámetro de configuración `StartHistoryPollers` en 0.

Campos configurables

Para utilizar items calculados, seleccione el tipo de item **Calculated**.



La **clave** es un identificador único del item (por host). Puede crear cualquier nombre de clave utilizando los símbolos compatibles. La clave permite hacer referencia al item calculado en los triggers.

La definición del cálculo debe introducirse en el campo **Formula**.

No existe ninguna relación entre la fórmula y la clave. Los parámetros de la clave no se utilizan en la fórmula de ninguna manera.

A diferencia de las expresiones de los triggers, Zabbix procesa los items calculados según el intervalo de actualización del item, no al recibir un nuevo valor.

Tenga en cuenta que, si selecciona *Numeric (unsigned)* como tipo de información y el resultado del cálculo es un valor de tipo float, este se trunca a un entero.

Fórmula de cálculo simple

La sintaxis de una fórmula simple es:

`function(/host/key,<parameter1>,<parameter2>,...)`

donde:

Elemento	Descripción
`function`	Una de las funciones compatibles: last, min, max, avg, count, etc.
`host`	Host del item que se utiliza para el cálculo. El host actual puede omitirse (es decir, como en `function(//key,parameter,...)`).
`key`	Clave del item que se utiliza para el cálculo. No se admiten los items que devuelven valores con el tipo de dato binario o JSON. Si cambia la clave del item de un item referenciado, la clave debe actualizarse manualmente en la fórmula. Todos los items a los que hacen referencia las funciones de historial deben existir y estar recopilando datos.
`parameter(s)`	Parámetros de la función, si son necesarios. Se admiten los sufijos de tiempo y los sufijos de tamaño de memoria.

Una fórmula más compleja puede utilizar una combinación de funciones, operadores y paréntesis.

Vea ejemplos de fórmulas de items calculados:

- (1) Tráfico total en la interfaz
- (2) Porcentaje de espacio libre en disco
- (3) Porcentaje de tráfico entrante
- (4) Número promedio de valores procesados

Algunas variables pueden pasarse a la fórmula con macros de usuario. Tenga en cuenta que las macros de usuario:

- se resolverán: si se utilizan para hacer referencia a un parámetro de función, parámetro de filtro de item o una constante.
- **no** se resolverán: si hacen referencia a una función, nombre de host, clave de item, parámetro de clave de item u operador.

Se permite la comparación con cadenas.

Si no hay datos recientes en la caché y no hay un período de consulta definido en la función, Zabbix, de forma predeterminada, buscará en el pasado hasta una semana para consultar la base de datos en busca de valores históricos.

Un item calculado puede pasar a no ser compatible en varios casos:

- item referenciado
  - no se encuentra
  - está deshabilitado o pertenece a un host deshabilitado
  - no es compatible (excepto con la función nodata() y los operadores con valores desconocidos)
- no hay datos para calcular una función
- división por cero
- se ha utilizado una sintaxis incorrecta

Agregaciones de datos

Las agregaciones de datos se basan en un filtro que selecciona varios items. Las agregaciones de datos también son un tipo de item calculado; sin embargo, en la fórmula se utiliza una sintaxis ligeramente diferente.

Para recuperar agregados, utilice una de las funciones de agregación compatibles: `avg`, `max`, `min`, `sum`, etc. Luego añada la función **foreach** como único parámetro y su filtro de item para seleccionar los items necesarios:

```
aggregate_function(function_foreach(/host/key?[group="host group"],timeperiod))
```

Una función **foreach** (por ejemplo, *avg_foreach*, *count_foreach*, etc.) devuelve un valor agregado por cada item seleccionado. Los items se seleccionan mediante el filtro de item (`/host/key?[group="host group"]`), a partir del historial del item. Para más detalles, consulte funciones foreach.

Si algunos de los items no tienen datos para el período solicitado, se ignoran en el cálculo. Si ningún item tiene datos, la función devolverá un error.

Como alternativa, puede enumerar varios items como parámetros para la agregación:

```
aggregate_function(function(/host/key,parameter),function(/host2/key2,parameter),...)
```

Tenga en cuenta que `function` aquí debe ser una función de historial/tendencia.

Vea ejemplos de cálculos agregados:

- (5) Tráfico entrante total para el host

- (6) Espacio total en disco para el grupo
- (7) Carga media del procesador para el grupo
- (8) Carga media del procesador para el grupo y la etiqueta
- (9) Número medio de valores procesados
- (10) Número medio de consultas para el grupo
- (11) Número total de comprobaciones DNS correctas
- (12) Número total de items no compatibles para el grupo
- (13) Estadísticas de interfaz para el grupo

Se admiten macros de usuario y macros de descubrimiento de bajo nivel en:

- parámetros de clave de item
- parámetros de función
- condiciones de filtro (nombre del grupo de hosts y nombre de etiqueta)
- constantes de expresión

Un cálculo agregado puede pasar a no ser compatible si:

- no se encuentra ninguno de los items referenciados (lo que puede ocurrir si la clave de item es incorrecta, ninguno de los items existe o todos los grupos incluidos son incorrectos)
- no hay datos para calcular una función

Ejemplos de items calculados

(1) Tráfico total en la interfaz

```
last(//net.if.in[eth0,bytes])+last(//net.if.out[eth0,bytes])
```

Calcule el ancho de banda total en eth0.

(2) Porcentaje de espacio libre en disco

```
100*last(//vfs.fs.size[/,free])/last(//vfs.fs.size[/,total])
```

Zabbix tomará los valores más recientes del espacio libre y del espacio total en disco (en /) y calculará el porcentaje de acuerdo con la fórmula proporcionada.

(3) Porcentaje del tráfico entrante

```
100*last(//net.if.in[eth0,bytes])/(last(//net.if.in[eth0,bytes])+last(//net.if.out[eth0,bytes]))
```

Calcule el porcentaje de tráfico entrante del tráfico total.

(4) Número promedio de valores procesados

```
avg(/Zabbix server/zabbix[wcache,values],10m)
```

Calcula un promedio de 10 minutos del número de valores procesados por Zabbix.

(5) Tráfico entrante total para el host

```
sum(last_foreach(/host/net.if.in[*]))
```

Calcula el total de todos los items que coinciden con `net.if.in[*]` en el host.

(6) Espacio total en disco para el grupo

```
sum(last_foreach(/*/vfs.fs.size[/,total]?[group="MySQL Servers"]))
```

Calcule el espacio total en disco para todos los `MySQL Servers`.

(7) Carga media del procesador para el grupo

```
avg(last_foreach(/*/system.cpu.load[,avg1]?[group="MySQL Servers"]))
```

Calcula la carga media del procesador para todos los `MySQL Servers`.

(8) Carga media del procesador por grupo y etiqueta

```
avg(last_foreach(/*/system.cpu.load?[(group="Servers A" or group="Servers B" or group="Servers C") and (ta
```

Calcular la carga media de CPU en todos los hosts de varios grupos de hosts que tienen etiquetas específicas.

(9) Número promedio de consultas para el grupo

```
avg(avg_foreach(/*/mysql.qps?[group="MySQL Servers"],5m))
```

Calcule el promedio de 5 minutos del número de consultas por segundo para `MySQL Servers`.

(10) Número total de comprobaciones DNS exitosas

```
sum(last_foreach(/*/net.dns[*,*,*]))
```

Calcule el número total de comprobaciones DNS exitosas en todos los hosts. El formulario de item mostrado es `net.dns[192.0.2.1,exampl` como ejemplo de una posible clave.

Tenga en cuenta que los comodines deben coincidir con el número de parámetros de la clave (aquí `net.dns` tiene tres parámetros: ip, name, type).

(11) Número total de items no soportados para el grupo

```
sum(last_foreach(/*/zabbix[host,,items_unsupported]?[group="Zabbix servers"]))
```

Calcule el número total de items no soportados para `Zabbix servers`.

(12) Estadísticas de interfaz para el grupo

```
sum(last_foreach(/*/net.if.out[eth0,bytes]?[group="video"])) / sum(last_foreach(/*/nginx_stat.sh[active]?[
```

Calcular las estadísticas de interfaz para el grupo `video`.

Ejemplos de sintaxis correcta/incorrecta

Las expresiones (incluidas las llamadas a funciones) no se pueden utilizar como parámetros de funciones de historial, tendencias o foreach. Sin embargo, esas funciones se pueden utilizar en otros parámetros de funciones (no históricos).

Expresión	Ejemplo
Válida	`avg(last(/host/key1),last(/host/key2)*10,last(/host/key1)*100)`
	`max(avg(avg_foreach(/*/system.cpu.load?[group="Servers A"],5m)),avg(avg_foreach(/*/system.cpu.load?[group="Servers B"],5m)),avg(avg_foreach(/*/system.cpu.load?[group="Servers C"],5m)))`
No válida	`sum(/host/key,10+2)`
	`sum(/host/key, avg(10,2))`
	`sum(/host/key,last(/host/key2))`

Tenga en cuenta que, en una expresión como:

```
sum(sum_foreach(//resptime[*],5m))/sum(count_foreach(//resptime[*],5m))
```

(donde `resptime[*]` es un ejemplo de un patrón de clave de item que coincide con las claves `resptime[1]`, `resptime[2]`, etc., utilizado para obtener el tiempo de respuesta de varios nodos activos del clúster), no se puede garantizar que ambas partes de la ecuación tengan siempre el mismo conjunto de valores. Mientras se evalúa una parte de la expresión, puede llegar un nuevo valor para el período solicitado y, por lo tanto, la otra parte de la expresión tendrá un conjunto de valores diferente.

Elemento dependiente

Descripción general

Hay situaciones en las que un elemento recopila múltiples métricas a la vez o incluso tiene más sentido recopilar métricas relacionadas simultáneamente, por ejemplo:

- Utilización de CPU de núcleos individuales
- Tráfico de red entrante/saliente/total

Para permitir la recopilación masiva de métricas y el uso simultáneo en varios elementos relacionados, Zabbix admite elementos dependientes. Los elementos dependientes dependen del elemento maestro que recopila sus datos simultáneamente, en una sola consulta. Un nuevo valor para el elemento maestro llena automáticamente los valores de los elementos dependientes. Los elementos dependientes no pueden tener un intervalo de actualización diferente al del elemento maestro.

Las opciones de preprocesamiento de Zabbix se pueden utilizar para extraer la parte que se necesita para el elemento dependiente a partir de los datos del elemento maestro.

El preprocesamiento es gestionado por un proceso `preprocessing manager`, junto con hilos de trabajo que realizan los pasos de preprocesamiento. Todos los valores con preprocesamiento, recibidos de diferentes recolectores de datos, pasan por el gestor de preprocesamiento antes de ser añadidos a la caché de historial. Se utiliza comunicación IPC basada en sockets entre los recolectores de datos (pollers, trappers, etc.) y el proceso de preprocesamiento.

El servidor Zabbix o el proxy Zabbix (si el host es monitorizado por un proxy) realizan los pasos de preprocesamiento y procesan los elementos dependientes.

Los elementos de cualquier tipo, incluso los elementos dependientes, pueden configurarse como elementos maestros. Se pueden utilizar niveles adicionales de elementos dependientes para extraer partes más pequeñas del valor de un elemento dependiente existente.

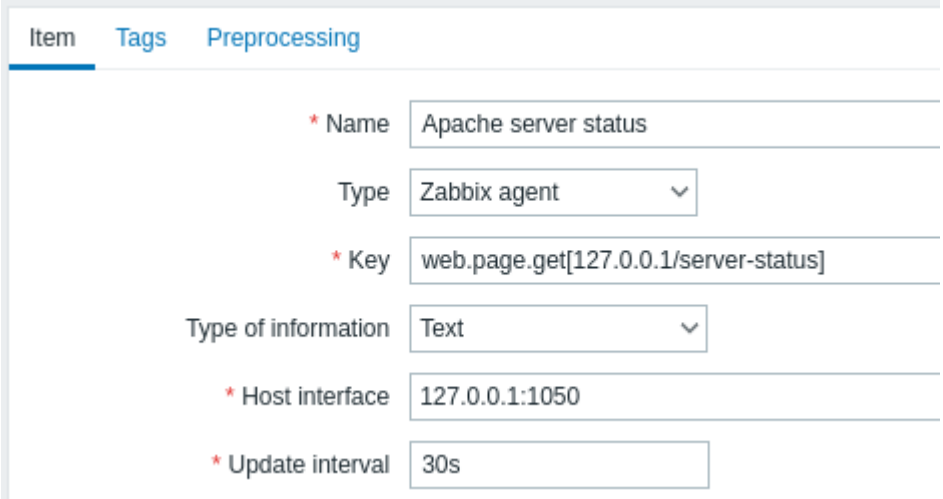
Limitaciones

- Solo se permiten dependencias en el mismo equipo (plantilla)
- Una métrica prototipo puede depender de otra métrica prototipo o de una métrica normal del mismo equipo
- Una métrica dependiente en un equipo con una métrica principal de una plantilla no se exportará a XML

Configuración de item

Un item dependiente depende de su item maestro para obtener datos. Por eso, el **item maestro** debe configurarse (o existir) primero:

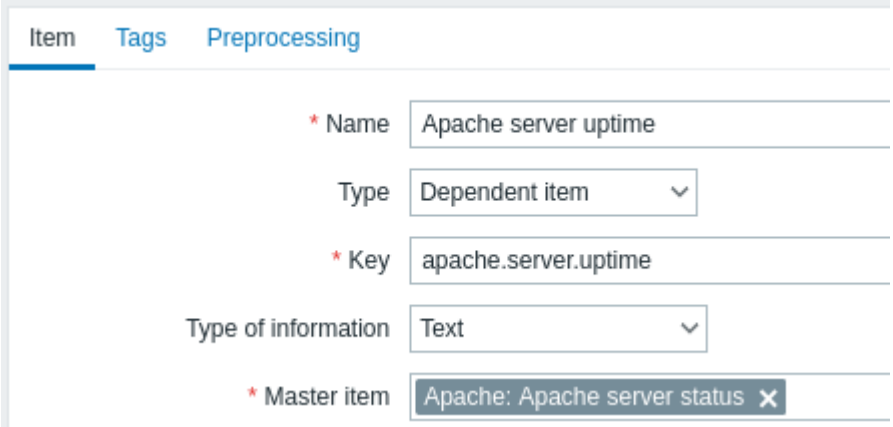
1. Vaya a *Recopilación de datos* > *Hosts*.
2. Haga clic en *Items* en la fila del host.
3. Haga clic en *Crear item*.
4. Introduzca los parámetros del item en el formulario.



Todos los campos obligatorios de entrada están marcados con un asterisco rojo.

Haga clic en *Añadir* para guardar el item maestro.

Después, puede configurar un **item dependiente**.

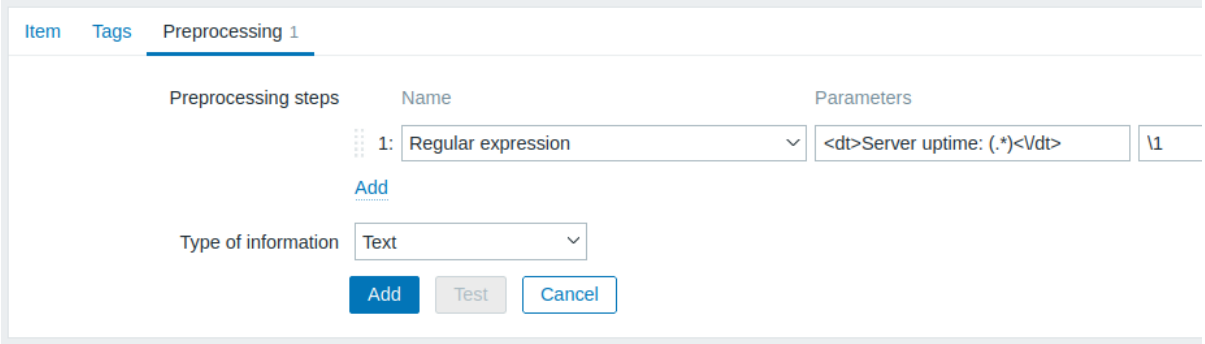


Todos los campos obligatorios de entrada están marcados con un asterisco rojo.

Los campos que requieren información específica para los items dependientes son:

Type	Seleccione **Dependent item** aquí.
Key	Introduzca una clave que se usará para reconocer el item.
Master item	Seleccione el item maestro. El valor del item maestro se usará para rellenar el valor del item dependiente.
Type of information	Seleccione el tipo de información que corresponderá al formato de los datos que se almacenarán.

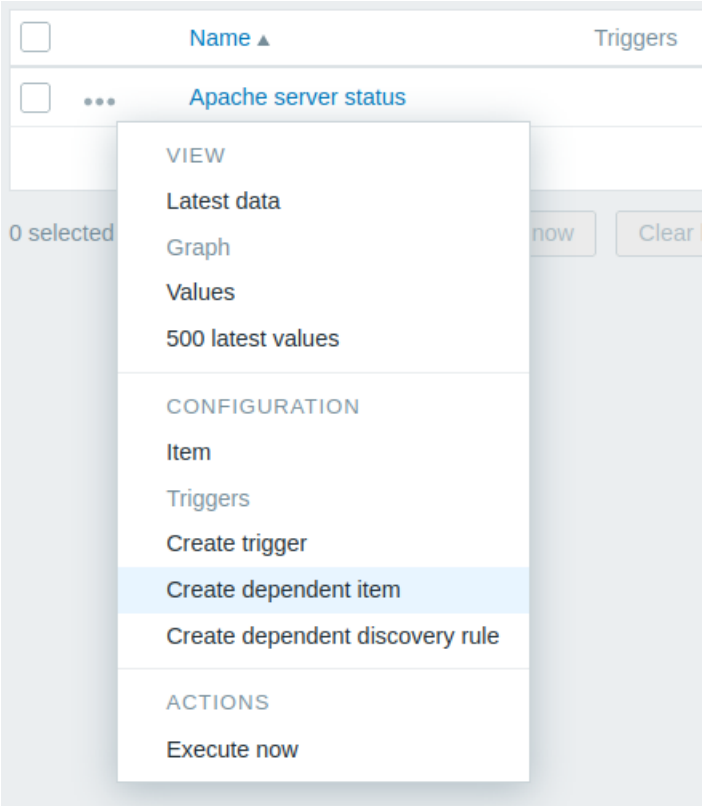
Puede usar el preprocesamiento del valor del item para extraer la parte necesaria del valor del item maestro.



Sin preprocesamiento, el valor del item dependiente será exactamente el mismo que el valor del item maestro.

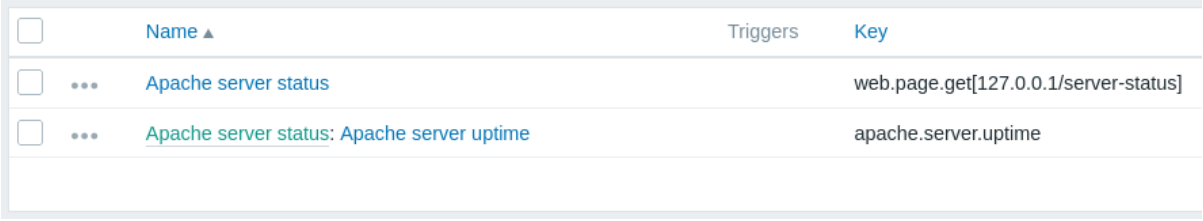
Haga clic en *Añadir* para guardar el item dependiente.

Puede acceder a un acceso directo para crear un item dependiente más rápidamente haciendo clic en el botón ••• en la lista de items y seleccionando *Crear item dependiente*.



Visualización

En la lista de métricas, las métricas dependientes se muestran con el nombre de su métrica principal como prefijo.



Si se elimina una métrica principal, también se eliminan todas sus métricas dependientes.

Elemento de script

Descripción general

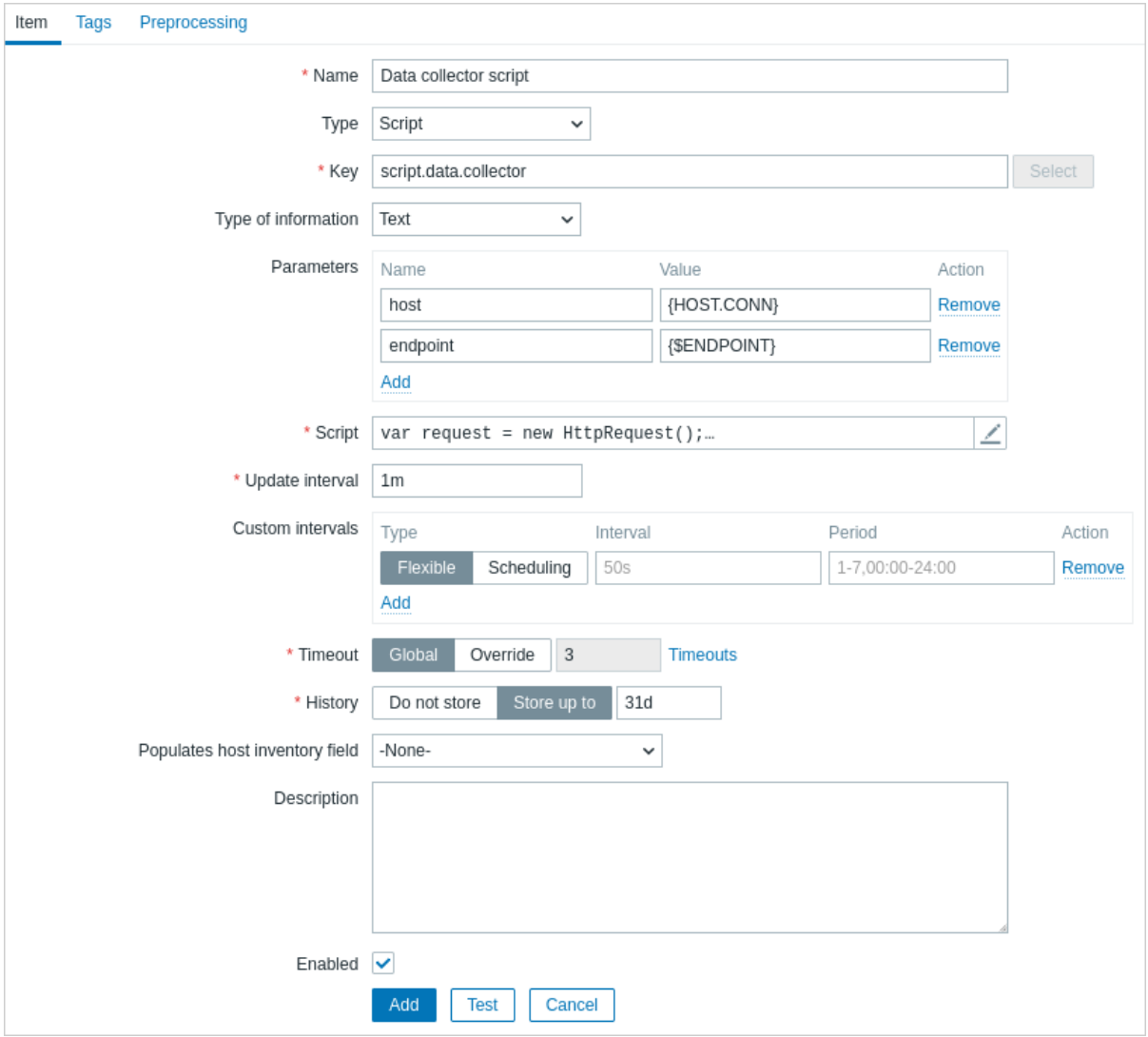
Las métricas de script pueden utilizarse para recopilar datos ejecutando un código JavaScript definido por el usuario con la capacidad de recuperar datos a través de HTTP/HTTPS. Además del script, se puede especificar una lista opcional de parámetros (pares de nombre y valor) y un tiempo de espera.

Este tipo de métrica puede ser útil en escenarios de recopilación de datos que requieran múltiples pasos o lógica compleja. Por ejemplo, una métrica de script puede configurarse para realizar una llamada HTTP, luego procesar los datos recibidos en el primer paso de alguna manera y pasar el valor transformado a la segunda llamada HTTP.

Las métricas de script son procesadas por los pollers del servidor o proxy de Zabbix.

Configuración

En el campo *Tipo* del formulario de configuración de la métrica, seleccione Script y luego complete los campos requeridos.



Todos los campos obligatorios están marcados con un asterisco rojo.

Los campos que requieren información específica para las métricas de tipo Script son:

Campo	Descripción
Clave	Introduzca una clave única que se utilizará para identificar la métrica.
Parámetros	Especifique las variables que se pasarán al script como pares atributo y valor. Se admiten macros de usuario. Para ver qué macros integradas son compatibles, busque "Métrica de tipo Script" en la tabla de macros compatibles.

Campo	Descripción
Script	Introduzca el código JavaScript en el editor modal que se abre al hacer clic en el campo de parámetro o en el icono de lápiz junto a él. Este código debe proporcionar la lógica para devolver el valor de la métrica. El código tiene acceso a todos los parámetros y a todos los objetos JavaScript adicionales añadidos por Zabbix. Consulte también: Guía de JavaScript.
Timeout	Tiempo de espera de ejecución de JavaScript (1-600s; si se excede, se devolverá un error). Tenga en cuenta que, dependiendo del script, puede tardar más en activarse el tiempo de espera. Para obtener más información sobre el parámetro *Timeout*, consulte atributos generales de la métrica.

Ejemplos

Recopilación de datos simple

Recopile el contenido de *https://www.example.com/release_notes*:

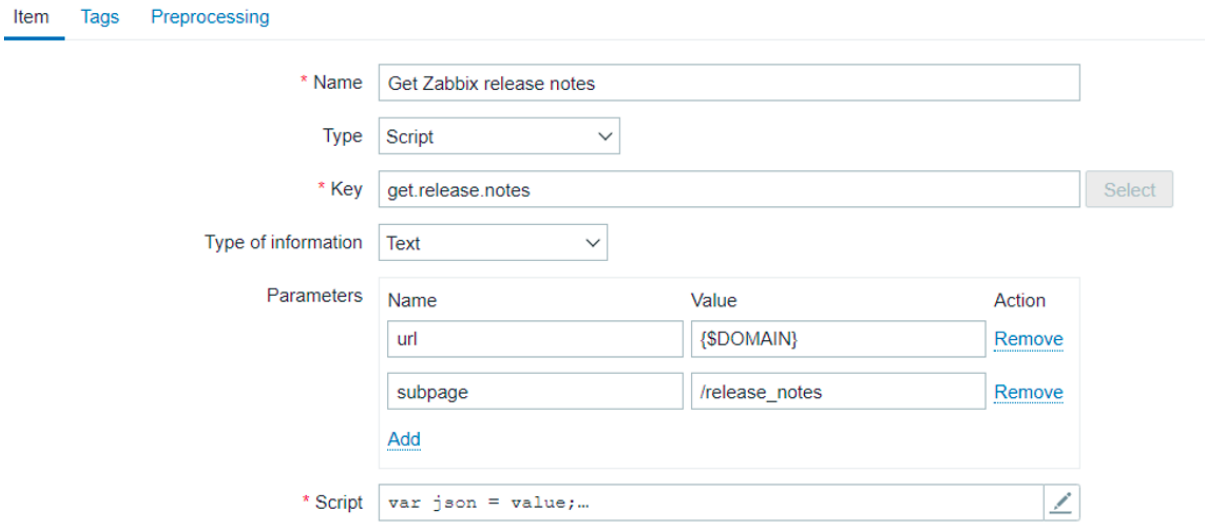
- Cree una métrica con tipo "Script".
- En el campo *Script*, introduzca:

```
var request = new HttpRequest();
return request.get("https://www.example.com/release_notes");
```

Recopilación de datos con parámetros

Recopile el contenido de una página específica y haga uso de parámetros:

- Cree una métrica con tipo "Script" y dos parámetros:
  - **url : {$DOMAIN}** (la macro de usuario {$DOMAIN} debe estar definida, preferiblemente a nivel de equipo)
  - **subpage : /release_notes**



- En el campo *Script*, introduzca:

```
var obj = JSON.parse(value);
var url = obj.url;
var subpage = obj.subpage;
var request = new HttpRequest();
return request.get(url + subpage);
```

Múltiples solicitudes HTTP

Recopile el contenido de *https://www.example.com* y *https://www.example.com/release_notes*:

- Cree una métrica con tipo "Script".
- En el campo *Script*, introduzca:

```
var request = new HttpRequest();
return request.get("https://www.example.com") + request.get("https://www.example.com/release_notes");
```

Registro

Agregue la entrada ”Log test” al registro del servidor Zabbix y reciba el valor de la métrica ”1” como respuesta:

- Cree una métrica con tipo ”Script”.
- En el campo *Script*, introduzca:

```
Zabbix.log(3, 'Log test');
return 1;
```

Elemento del navegador

Overview

Los items de navegador permiten supervisar sitios web y aplicaciones web complejos mediante un navegador.

> **Attention:**
> La compatibilidad con los items de navegador es actualmente experimental.

Los items de navegador recopilan datos ejecutando código JavaScript definido por el usuario y recuperando datos a través de HTTP/HTTPS. Este item puede simular acciones relacionadas con el navegador, como hacer clic, introducir texto, navegar por páginas web y otras interacciones de usuario con sitios web o aplicaciones web.

Además del script, se puede especificar una lista opcional de parámetros (pares de nombre y valor) y un tiempo de espera.

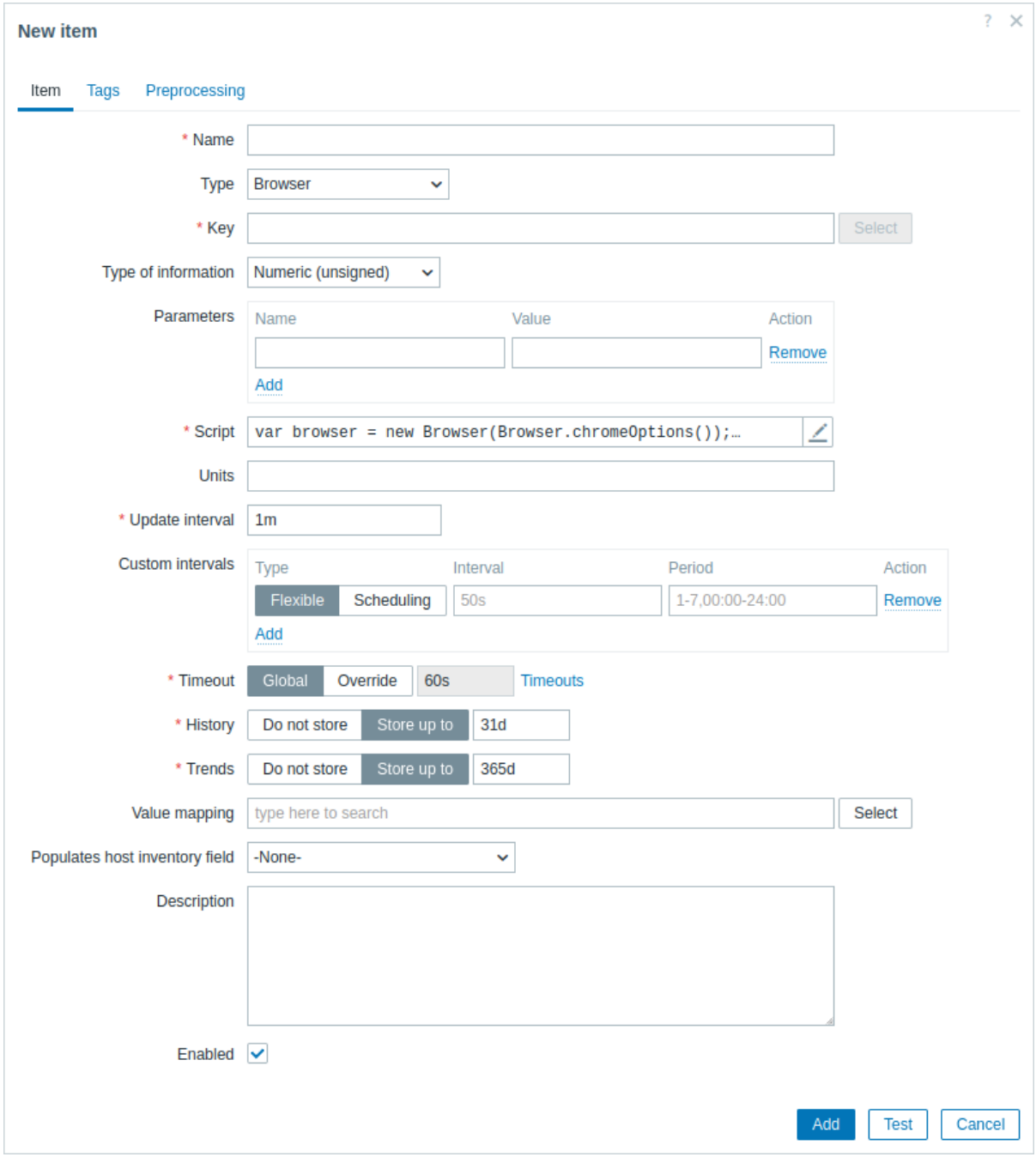
> **Attention:**
> El item implementa parcialmente el estándar W3C WebDriver con Selenium Server o un WebDriver simple (por ejemplo, ChromeDriver) como endpoint de pruebas web. Para que el item funcione, establezca el parámetro `WebDriverURL` en el archivo de configuración de server/proxy de Zabbix (si usa ChromeDriver, consulte Security Considerations). Para un mejor rendimiento, considere usar un server dedicado para el entorno de pruebas web.

Las comprobaciones de items de navegador son ejecutadas y procesadas por los browser pollers de Zabbix server o proxy. Si es necesario, puede ajustar el número de instancias pre-fork de browser pollers en el parámetro `StartBrowserPollers` del archivo de configuración de server/proxy de Zabbix.

Para supervisar sitios web y aplicaciones web complejos, está disponible el template Website by Browser como un template listo para usar.

Configuración

En el campo *Tipo* del formulario de configuración de métricas, seleccione Browser y luego complete los campos requeridos.



Todos los campos obligatorios están marcados con un asterisco rojo.

Los campos que requieren información específica para las métricas de tipo Browser son:

Campo	Descripción
Clave	Introduzca una clave única que se utilizará para identificar la métrica.
Parámetros	Especifique las variables que se pasarán al script como pares atributo y valor. Se admiten macros de usuario. Para ver qué macros integradas son compatibles, busque "métrica de tipo Browser" en la tabla de macros compatibles.
Script	Introduzca el código JavaScript en el editor modal que se abre al hacer clic en el campo de parámetros o en el icono de lápiz junto a él. Este código debe proporcionar la lógica para devolver el valor de la métrica. El código tiene acceso a todos los parámetros, a todos los objetos JavaScript adicionales y a los objetos JavaScript de métricas Browser añadidos por Zabbix. Consulte también: Guía de JavaScript.
Timeout	Tiempo de espera para la ejecución de JavaScript (1-600s; si se supera, se devolverá un error). Tenga en cuenta que, dependiendo del script, puede tardar más en activarse el tiempo de espera. Para obtener más información sobre el parámetro *Timeout*, consulte atributos generales de la métrica.

Ejemplos

Para ver un ejemplo de cómo configurar Zabbix para supervisar sitios web mediante la template *Website by Browser*, consulte Monitor websites with Browser items.

Script por defecto

El siguiente script:

1. Inicializa una sesión de navegador.
2. Navega a una URL especificada.
3. Recopila entradas de rendimiento y estadísticas de sesión, y las devuelve como una cadena JSON.

En el campo *Script*, introduzca:

```
var browser = new Browser(Browser.chromeOptions());

try {
    browser.navigate("http://example.com");
    browser.collectPerfEntries();
}
finally {
    return JSON.stringify(browser.getResult());
}
```

Inicializar el navegador con capacidades personalizadas

El siguiente script:

1. Inicializa una sesión de navegador para el navegador disponible según el primer navegador coincidente en el orden especificado dentro del script.
2. Define las capacidades del navegador, incluida la estrategia de carga de página y las opciones específicas para cada navegador, como el modo sin interfaz gráfica para los navegadores Chrome, Firefox y Microsoft Edge.

En el campo *Script*, introduzca:

```
var browser = new Browser({
    "capabilities":{
        "firstMatch":[
            {
                "browserName":"chrome",
                "pageLoadStrategy":"normal",
                "goog:chromeOptions":{
                    "args":[
                        "--headless=new"
                    ]
                }
            },
            {
                "browserName":"firefox",
                "pageLoadStrategy":"normal",
                "moz:firefoxOptions":{
                    "args":[
                        "--headless"
                    ]
                }
            },
            {
                "browserName":"MicrosoftEdge",
                "pageLoadStrategy":"normal",
                "ms:edgeOptions":{
                    "args":[
                        "--headless=new"
                    ]
                }
            },
            {
                "browserName":"safari",
```

```
                "pageLoadStrategy":"normal"
            }
        ]
    }
});
```

Inicializar el navegador con GUI

Por defecto, las sesiones del navegador (excepto Safari) se inicializan en modo sin cabeza, lo que significa que la interfaz gráfica de usuario (GUI) del navegador no se muestra.

El siguiente script inicializa una sesión del navegador con la GUI habilitada.

Tenga en cuenta que si el WebDriver no puede localizar el binario del navegador, puede especificar la ruta manualmente.

```
var opts = Browser.chromeOptions();
opts.capabilities.alwaysMatch['goog:chromeOptions'].args = [];

// Para inicializar una sesión de Firefox con GUI, descomente las siguientes líneas:
// var opts = Browser.firefoxOptions();
// opts.capabilities.alwaysMatch['moz:firefoxOptions'].binary = 'usr/bin/firefox';
// opts.capabilities.alwaysMatch['moz:firefoxOptions'].args = [];

// Para inicializar una sesión de Microsoft Edge con GUI, descomente las siguientes líneas:
// var opts = Browser.edgeOptions();
// opts.capabilities.alwaysMatch['ms:edgeOptions'].binary = 'usr/bin/microsoft-edge';
// opts.capabilities.alwaysMatch['ms:edgeOptions'].args = [];

var browser = new Browser(opts);
```

> **Note:**
> Si sus pruebas se están ejecutando en un servidor remoto o en un contenedor, puede utilizar un cliente de Virtual Network Computing (VNC) para conectarse al servidor VNC de la máquina. Esto le permite ver e interactuar con la GUI del navegador de forma remota.

Tomar capturas de pantalla

El siguiente script:

1. Inicializa una sesión de navegador.
2. Establece el tamaño de la ventana del navegador para determinar el tamaño de la captura de pantalla (especificado como parámetros, ver más abajo).
3. Navega a una URL (especificada como parámetro, ver más abajo).
4. Recopila estadísticas de la sesión, captura una captura de pantalla y la añade a las estadísticas recopiladas.
5. Maneja errores capturando mensajes de error y una captura de pantalla.
6. Devuelve los resultados recopilados como una cadena JSON.

El script también utiliza parámetros del formulario de configuración de la métrica:

- webURL - http://example.com
- width - 1920
- height - 1080

En el campo *Script*, introduzca:

```
var browser, result;

var browser = new Browser(Browser.chromeOptions());

try {
    var params = JSON.parse(value); // Analiza la cadena JSON que contiene los parámetros pasados desde Zo

    browser.setScreenSize(Number(params.width), Number(params.height))

    browser.navigate(params.webURL);

    result = browser.getResult();
    result.screenshot = browser.getScreenshot();
```

```
}
catch (err) {
    if (!(err instanceof BrowserError)) {
        browser.setError(err.message);
    }
    result = browser.getResult();
    result.error.screenshot = browser.getScreenshot();
}
finally {
    return JSON.stringify(result);
}
```

Comprobar inicio de sesión en Zabbix

El siguiente script:

1. Inicializa una sesión de navegador.
2. Navega a una página (especificada como parámetro, ver más abajo).
3. Introduce el nombre de usuario y la contraseña (especificados como parámetros, ver más abajo).
4. Busca y hace clic en el botón de inicio de sesión.
5. Busca y hace clic en el botón de cierre de sesión.
6. Recoge datos de rendimiento antes y después del inicio de sesión, así como después del cierre de sesión.
7. Gestiona los errores capturando mensajes de error y una captura de pantalla.
8. Devuelve los resultados recogidos como una cadena JSON.

El script también utiliza parámetros del formulario de configuración de la métrica:

- webURL - http://{HOST.CONN}/index.php
- username - {$USERNAME}
- password - {$PASSWORD}

En el campo *Script*, introduzca:

```
var browser, result;

browser = new Browser(Browser.chromeOptions());

try {
    var params = JSON.parse(value); // Analiza la cadena JSON que contiene los parámetros pasados desde Za

    browser.navigate(params.webURL);
    browser.collectPerfEntries("open page");

    var el = browser.findElement("xpath", "//input[@id='name']");
    if (el === null) {
        throw Error("cannot find name input field");
    }
    el.sendKeys(params.username);

    el = browser.findElement("xpath", "//input[@id='password']");
    if (el === null) {
        throw Error("cannot find password input field");
    }
    el.sendKeys(params.password);

    el = browser.findElement("xpath", "//button[@id='enter']");
    if (el === null) {
        throw Error("cannot find login button");
    }
    el.click();

    browser.collectPerfEntries("login");

    el = browser.findElement("link text", "Sign out");
    if (el === null) {
        throw Error("cannot find logout button");
```

```
    }
    el.click();

    browser.collectPerfEntries("logout");

    result = browser.getResult();
}
catch (err) {
    if (!(err instanceof BrowserError)) {
        browser.setError(err.message);
    }
    result = browser.getResult();
    result.error.screenshot = browser.getScreenshot();
}
finally {
    return JSON.stringify(result);
}
```

Buscar enlaces

El siguiente script:

1. Inicializa una sesión de navegador.
2. Define una función para eliminar elementos duplicados de un array (ver paso 5).
3. Navega a una página (especificada como parámetros, ver más abajo).
4. Busca enlaces en la página.
5. Elimina los enlaces duplicados para asegurar que sean únicos.
6. Extrae solo los enlaces que comienzan con "http".
7. Da formato a los enlaces extraídos a una estructura específica.
8. Maneja errores capturando mensajes de error y una captura de pantalla.
9. Devuelve los resultados recopilados como una cadena JSON.

El script también utiliza parámetros del formulario de configuración de la métrica:

- scheme - {$WEBSITE.SCHEME}
- domain - {$WEBSITE.DOMAIN}
- path - {$WEBSITE.PATH}

En el campo *Script*, introduzca:

```
var browser, result;

browser = new Browser(Browser.chromeOptions());

try {
    var params = JSON.parse(value);  // Analiza la cadena JSON que contiene los parámetros pasados desde Z

    function uniq(a) {
        return a.sort().filter(function (item, pos, ary) {
            return !pos || item != ary[pos - 1];
        });
    }

    browser.navigate(params.scheme + '://' + params.domain + params.path);

    var el = browser.findElements("link text", "");
    var links = [];
    for (var n = 0; n < el.length; n++) {
        links.push(el[n].getAttribute('href'));
    }

    links = uniq(links);

    result = [];
    for (i = 0; i < links.length; i++) {
        if (links[i].match(/^http.*/)) {
```

```
            var row = {};
            row["{#URL}"] = links[i];
            result.push(row);
        }
    }
}
catch (err) {
    if (!(err instanceof BrowserError)) {
        browser.setError(err.message);
    }
    result = browser.getResult();
    result.error.screenshot = browser.getScreenshot();
}
finally {
    return JSON.stringify(result);
}
```

Consulta de telemetría

Este tipo de item permite recuperar o agregar datos de telemetría. Los datos de telemetría se utilizan para la supervisión del rendimiento de las aplicaciones (APM), lo que permite ver cómo fluyen las solicitudes a través de sus aplicaciones y servicios, medir su rendimiento y encontrar la causa de las operaciones lentas o fallidas.

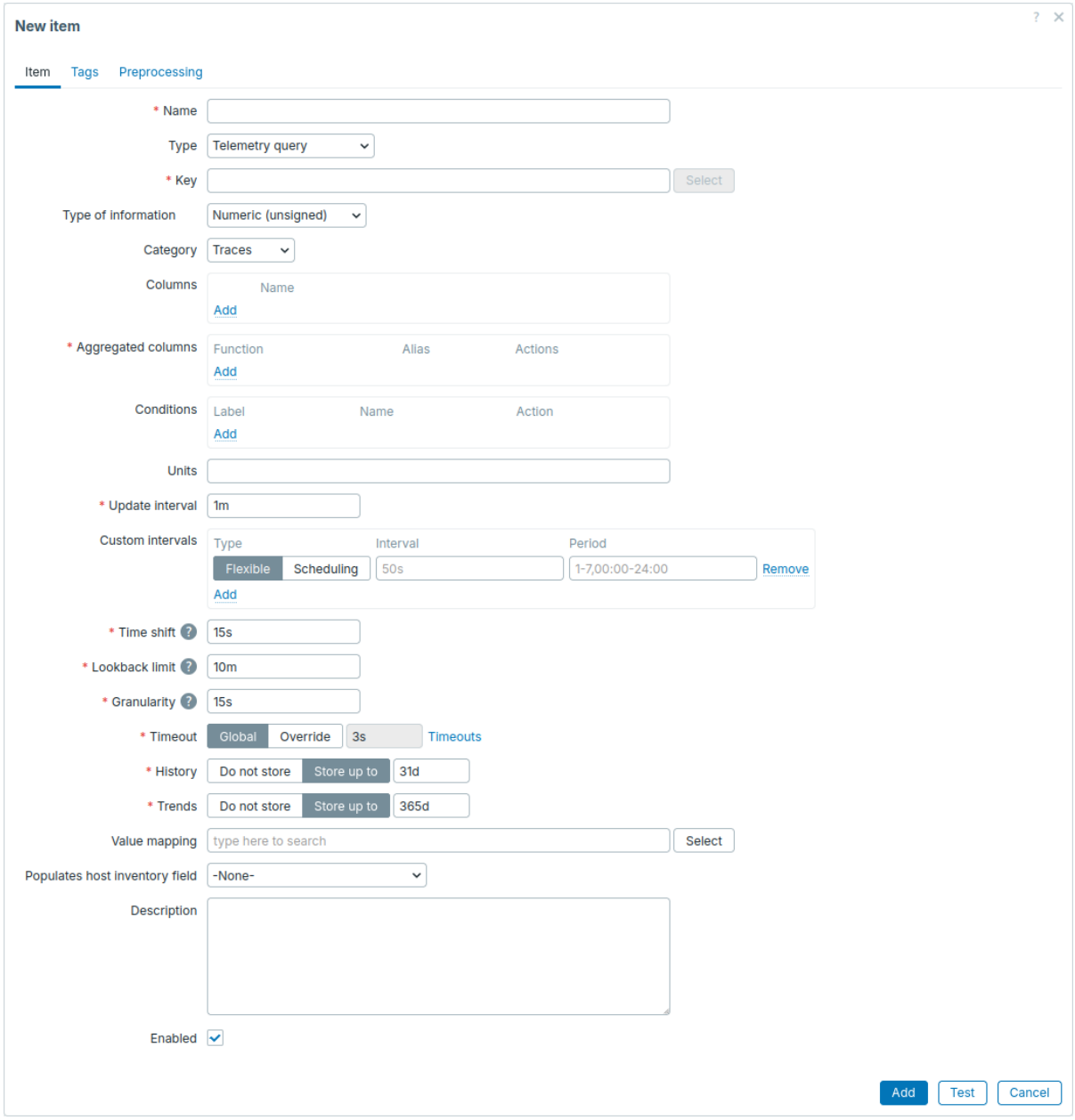
Los datos de telemetría se presentan en tres tipos: trazas (la ruta que sigue una solicitud a través de sus aplicaciones y servicios), métricas (mediciones, como el tiempo de respuesta) y registros (registros de texto de eventos). Mediante el item de consulta de telemetría, puede, por ejemplo, encontrar todas las trazas en las que un servicio devolvió un error, obtener el tiempo de respuesta promedio a partir de sus métricas o buscar un mensaje de error específico en sus registros.

Al configurar un item de consulta de telemetría, se define un conjunto de instrucciones (una consulta) que indica a Zabbix qué datos de telemetría solicitar, cómo filtrarlos y cómo agregarlos. Cuando se ejecuta la comprobación del item, Zabbix envía esta consulta a la fuente de datos de telemetría y recupera el resultado según la consulta.

Para utilizar este item, especifique la fuente de datos de telemetría y establezca el parámetro `StartTelemetryQueryPollers` en la configuración del Zabbix server o del Zabbix proxy, según cuál de ellos supervise el host del item.

Configuración

Al crear un item, establezca su tipo en *Consulta de telemetría*:



Parámetro	Descripción
Nombre	Consulte los parámetros comunes a todos los items.
Tipo	Seleccione *Consulta de telemetría*.
Clave	Consulte los parámetros comunes a todos los items.
Tipo de información	
Categoría	Seleccione el tipo de señal que desea consultar: *Trazas*, *Métricas* o *Registros*.
Puntos de métrica	Seleccione el tipo de punto de métrica que desea consultar: *Suma*, *Indicador*, *Histograma* o *Histograma exponencial*. Este parámetro está disponible si *Categoría* está establecido en *Métricas*.
Columnas	Añada las columnas de datos de telemetría que desea consultar. Para las columnas que contienen pares clave-valor, introduzca también el nombre de la clave. Los registros de estas columnas se incluirán tal cual en el valor del item.

Parámetro	Descripción
Columnas agregadas	Añada las columnas de datos de telemetría que desea consultar y cómo calcular un valor a partir de sus registros: **Función**: seleccione la función que se aplicará a los registros de la columna: *min*, *max*, *avg*, *count*, *sum* o *percentile*. **Porcentaje**: introduzca el percentil que se calculará a partir de los registros de la columna (si *Función* está establecida en *percentile*). **Columna**: seleccione la columna a partir de la cual se calculará el valor (si *Función* no está establecida en *count*). **Alias**: introduzca un nombre único para el valor calculado, que se utilizará para identificarlo en el valor del item.
Tipo de cálculo	Seleccione cómo evaluar las condiciones establecidas en el parámetro *Condiciones* siguiente: **And/Or**: deben cumplirse todas las condiciones; las condiciones con el mismo *Tipo* y *Clave de atributo* se combinan con *Or*. **And**: deben cumplirse todas las condiciones. **Or**: debe cumplirse al menos una condición. **Expresión personalizada**: las condiciones deben cumplirse según lo especificado en la fórmula, por ejemplo, `A and (B or C)`. Este parámetro está disponible si ha establecido dos o más condiciones.
Condiciones	Añada condiciones para filtrar los registros de las columnas de datos de telemetría: **Tipo**: seleccione la columna de datos de telemetría a la que se aplicará la condición. **Clave de atributo**: introduzca la clave a la que se aplicará la condición (si *Tipo* está establecido en una columna con pares clave-valor). **Operador**: seleccione la comparación que se utilizará (consulte más abajo) entre el valor del registro y el *Valor del atributo*. **Valor del atributo**: introduzca el valor con el que se comparará el valor del registro (si *Operador* no está establecido en *Existe*). Si *Tipo* está establecido en una columna de valor único, la condición se aplicará al registro de la columna; si la columna contiene una matriz de valores, la condición coincidirá con el registro si al menos un valor de la matriz cumple la condición. Si *Tipo* está establecido en una columna que contiene pares clave-valor, la condición se aplicará al valor de la clave especificada en *Clave de atributo*; si la columna contiene una matriz de pares clave-valor, la condición coincidirá con el registro si al menos un par con la *Clave de atributo* especificada cumple la condición. Para columnas de valor único, puede utilizar estos operadores: **Igual a**: coincide con los registros que son iguales al *Valor del atributo* (distingue entre mayúsculas y minúsculas). **No es igual a**: coincide con los registros que no son iguales al *Valor del atributo* (distingue entre mayúsculas y minúsculas). **Contiene**: coincide con los registros que contienen el *Valor del atributo* (distingue entre mayúsculas y minúsculas). **No contiene**: coincide con los registros que no contienen el *Valor del atributo* (distingue entre mayúsculas y minúsculas). Para columnas que contienen pares clave-valor, puede utilizar estos operadores: **Igual a**: coincide con los registros en los que existe la *Clave de atributo* y su valor es igual al *Valor del atributo* (distingue entre mayúsculas y minúsculas). **No es igual a**: coincide con los registros en los que existe la *Clave de atributo* y su valor no es igual al *Valor del atributo* (distingue entre mayúsculas y minúsculas). **Existe**: coincide con los registros en los que existe la *Clave de atributo* (distingue entre mayúsculas y minúsculas).
Unidades *Intervalo de actualización* *Intervalos personalizados*	Consulte los parámetros comunes a todos los items.

Parámetro	Descripción
Desplazamiento temporal	Introduzca cuánto tiempo hacia atrás (antes de la hora actual) debe finalizar el intervalo de tiempo que consulta el item. Admite un número de segundos (por ejemplo, 1, 60, 300), un valor con un sufijo de tiempo (por ejemplo, 30s, 1m, 2h, 1d) o macros de usuario. Cada consulta solicita datos para un intervalo de tiempo. Normalmente, el intervalo comienza donde terminó el intervalo de tiempo de la consulta anterior y finaliza en el punto establecido por *Desplazamiento temporal*. Si el intervalo de tiempo de la consulta anterior terminó después de la hora actual menos la *Granularidad*, el intervalo de tiempo de la consulta actual comienza en la hora actual menos la *Granularidad*, superponiéndose así al intervalo de tiempo de la consulta anterior. Si no existe una consulta anterior desde la que continuar, el intervalo comienza en el punto establecido por el *Límite de búsqueda retrospectiva*. Si el host de este item pasa a ser monitorizado por otra entidad (por ejemplo, otro proxy de un grupo de proxies), es posible que el item ya no sepa cuándo terminó la consulta anterior. Esto puede provocar valores duplicados.
Límite de búsqueda retrospectiva	Introduzca cuánto tiempo hacia atrás (antes de la hora actual) debe consultar datos el item cuando no existe una consulta anterior desde la que continuar (por ejemplo, en la primera ejecución del item), o cuando la consulta anterior es más antigua que el *Intervalo de actualización* (por ejemplo, porque el Zabbix server estaba detenido). Admite un número de segundos (por ejemplo, 1, 60, 300), un valor con un sufijo de tiempo (por ejemplo, 30s, 1m, 2h, 1d) o macros de usuario. Si la fuente de datos de telemetría contiene una gran cantidad de datos (por ejemplo, más de un millón de registros por hora), un *Límite de búsqueda retrospectiva* grande puede reducir significativamente el rendimiento cuando el item lo utiliza.
Granularidad	Introduzca la duración de cada segmento en el que se divide el intervalo de tiempo completo de la consulta. Admite un número de segundos (por ejemplo, 1, 60, 300), un valor con un sufijo de tiempo (por ejemplo, 30s, 1m, 2h, 1d) o macros de usuario.
Tiempo de espera *Historial* *Tendencias* *Asignación de valores* *Rellena el campo de inventario del host* *Descripción* *Habilitado*	Consulte los parámetros comunes a todos los items.

Columnas de datos de telemetría compatibles

Esta sección enumera las columnas compatibles con estos parámetros del item: *Columns*, *Aggregated columns* y *Conditions*. Estas son las columnas que ClickHouse utiliza para almacenar datos de telemetría, según el esquema del exporter de ClickHouse de OpenTelemetry.

Seleccione la *Category*:

- Traces
- Metrics: Sum
- Metrics: Gauge
- Metrics: Histogram
- Metrics: Exponential histogram
- Logs

Trazas

Columna	*Columnas*	*Columnas agregadas*	*Condiciones*
Duración	✓	✓	–
Events.Attributes	–	–	✓
Events.Name	–	–	✓
ParentSpanId	✓	–	✓
ResourceAttributes	✓	–	✓
ScopeName	✓	–	✓

Columna	*Columnas*	*Columnas agregadas*	*Condiciones*
ScopeVersion	✓	–	✓
ServiceName	✓	–	✓
SpanAttributes	✓	–	✓
SpanId	✓	–	✓
SpanKind	✓	–	✓
SpanName	✓	–	✓
StatusCode	✓	–	✓
StatusMessage	✓	–	✓
Marca de tiempo	✓	✓	–
TraceId	✓	–	✓
TraceState	✓	–	✓

Métricas: Suma

Columna	*Columnas*	*Columnas agregadas*	*Condiciones*
AggregationTemporality	✓	–	–
Attributes	✓	–	✓
Exemplars.FilteredAttributes	–	–	✓
Flags	✓	–	–
IsMonotonic	✓	–	–
MetricDescription	✓	–	✓
MetricName	✓	–	✓
MetricUnit	✓	–	✓
ResourceAttributes	✓	–	✓
ResourceSchemaUrl	✓	–	✓
ScopeAttributes	✓	–	✓
ScopeDroppedAttrCount	✓	✓	–
ScopeName	✓	–	✓
ScopeSchemaUrl	✓	–	✓
ScopeVersion	✓	–	✓
ServiceName	✓	–	✓
StartTimeUnix	✓	✓	–
TimeUnix	✓	✓	–
Value	✓	✓	–

Métricas: Gauge

Columna	*Columnas*	*Columnas agregadas*	*Condiciones*
Atributos	✓	–	✓
Exemplars.FilteredAttributes	–	–	✓
Flags	✓	–	–
MetricDescription	✓	–	✓
MetricName	✓	–	✓
MetricUnit	✓	–	✓
ResourceAttributes	✓	–	✓
ResourceSchemaUrl	✓	–	✓
ScopeAttributes	✓	–	✓
ScopeDroppedAttrCount	✓	✓	–
ScopeName	✓	–	✓
ScopeSchemaUrl	✓	–	✓
ScopeVersion	✓	–	✓
ServiceName	✓	–	✓
StartTimeUnix	✓	✓	–
TimeUnix	✓	✓	–
Value	✓	✓	–

Métricas: Histograma

Columna	*Columnas*	*Columnas agregadas*	*Condiciones*
AggregationTemporality	✓	–	–
Attributes	✓	–	✓
Count	✓	✓	–
Exemplars.FilteredAttributes	–	–	✓
Flags	✓	–	–
Max	✓	✓	–
MetricDescription	✓	–	✓
MetricName	✓	–	✓
MetricUnit	✓	–	✓
Min	✓	✓	–
ResourceAttributes	✓	–	✓
ResourceSchemaUrl	✓	–	✓
ScopeAttributes	✓	–	✓
ScopeDroppedAttrCount	✓	✓	–
ScopeName	✓	–	✓
ScopeSchemaUrl	✓	–	✓
ScopeVersion	✓	–	✓
ServiceName	✓	–	✓
StartTimeUnix	✓	✓	–
Sum	✓	✓	–
TimeUnix	✓	✓	–

Métricas: histograma exponencial

Columna	*Columnas*	*Columnas agregadas*	*Condiciones*
AggregationTemporality	✓	–	–
Attributes	✓	–	✓
Count	✓	✓	–
Exemplars.FilteredAttributes	–	–	✓
Flags	✓	–	–
Max	✓	✓	–
MetricDescription	✓	–	✓
MetricName	✓	–	✓
MetricUnit	✓	–	✓
Min	✓	✓	–
NegativeOffset	✓	✓	–
PositiveOffset	✓	✓	–
ResourceAttributes	✓	–	✓
ResourceSchemaUrl	✓	–	✓
Scale	✓	✓	–
ScopeAttributes	✓	–	✓
ScopeDroppedAttrCount	✓	✓	–
ScopeName	✓	–	✓
ScopeSchemaUrl	✓	–	✓
ScopeVersion	✓	–	✓
ServiceName	✓	–	✓
StartTimeUnix	✓	✓	–
Sum	✓	✓	–
TimeUnix	✓	✓	–
ZeroCount	✓	✓	–

Registros

Columna	*Columnas*	*Columnas agregadas*	*Condiciones*
Body	✓	–	✓
EventName	✓	–	✓
LogAttributes	✓	–	✓
ResourceAttributes	✓	–	✓

Columna	*Columnas*	*Columnas agregadas*	*Condiciones*
ResourceSchemaUrl	✓	–	✓
ScopeAttributes	✓	–	✓
ScopeName	✓	–	✓
ScopeSchemaUrl	✓	–	✓
ScopeVersion	✓	–	✓
SpanId	✓	–	✓
ServiceName	✓	–	✓
SeverityNumber	✓	✓	–
SeverityText	✓	–	✓
Timestamp	✓	✓	–
TraceFlags	✓	–	–
TraceId	✓	–	✓

Historial y tendencias

Descripción general

El historial y las tendencias son las dos formas de almacenar los datos recopilados en Zabbix.

Mientras que el historial guarda cada valor recopilado, las tendencias mantienen información promediada por hora y, por lo tanto, requieren menos recursos.

Conservación del historial

Puede establecer durante cuántos días se conservará el historial:

- en el formulario de propiedades del item
- al actualizar items de forma masiva
- al configurar las tareas del housekeeper

Cualquier dato más antiguo será eliminado por el housekeeper.

La recomendación general es conservar el historial durante el menor número posible de días y, de ese modo, no sobrecargar la base de datos con una gran cantidad de valores históricos.

En lugar de conservar un historial largo, puede conservar durante más tiempo los datos de tendencias. Por ejemplo, podría conservar el historial durante 14 días y las tendencias durante 5 años.

Puede obtener una buena idea de cuánto espacio requieren los datos de historial frente a los datos de tendencias consultando la página de dimensionamiento de la base de datos.

Aunque conserve un historial más corto, seguirá pudiendo revisar datos antiguos en los gráficos, ya que los gráficos utilizarán valores de tendencias para mostrar los datos más antiguos.

Attention:
Si el historial se establece en '0', el item solo actualizará los items dependientes y el inventario. No se evaluará ninguna función de trigger porque la evaluación de triggers se basa únicamente en datos del historial.

Note:
Como forma alternativa de conservar el historial, considere utilizar la funcionalidad de exportación del historial de los módulos cargables.

Mantener tendencias

Las tendencias son un mecanismo incorporado de reducción de datos históricos que almacena el valor mínimo, máximo, promedio y el número total de valores por cada hora para los tipos de datos numéricos.

Puede establecer durante cuántos días se conservarán las tendencias:

- en el formulario de propiedades del item
- al actualizar items de forma masiva
- al configurar tareas del Housekeeper

Por lo general, las tendencias pueden conservarse durante mucho más tiempo que el historial. Cualquier dato más antiguo será eliminado por el housekeeper.

El servidor Zabbix acumula datos de tendencias en tiempo de ejecución en la caché de tendencias, a medida que llegan los datos. El servidor vuelca las tendencias de la **hora anterior** de cada item en la base de datos (donde el frontend puede encontrarlas) en estas situaciones:

- el servidor recibe el primer valor de la hora actual del item
- quedan 5 minutos o menos de la hora actual y aún no hay valores de la hora actual del item
- el servidor se detiene

Para ver las tendencias en un gráfico, debe esperar al menos hasta el comienzo de la siguiente hora (si el item se actualiza con frecuencia) y como máximo hasta el final de la siguiente hora (si el item se actualiza raramente), lo que supone un máximo de 2 horas.

Cuando el servidor vacía la caché de tendencias y ya hay tendencias en la base de datos para esa hora (por ejemplo, el servidor se ha reiniciado a mitad de hora), el servidor necesita usar sentencias de actualización en lugar de simples inserciones. Por lo tanto, en una instalación grande, si es necesario reiniciar, es recomendable detener el servidor al final de una hora y arrancarlo al principio de la siguiente hora para evitar la superposición de datos de tendencias.

Las tablas de historial no participan en la generación de tendencias de ninguna manera.

Attention:
Si las tendencias se establecen en '0', el servidor Zabbix no calcula ni almacena tendencias en absoluto.

Note:
Las tendencias se calculan y almacenan con el mismo tipo de datos que los valores originales. Como resultado, los cálculos del valor promedio de los valores de tipo de datos sin signo se redondean y cuanto menor sea el intervalo de valores, menos preciso será el resultado. Por ejemplo, si el item tiene valores 0 y 1, el valor promedio será 0, no 0.5.
Además, reiniciar el servidor puede provocar la pérdida de precisión en los cálculos del valor promedio de tipo de datos sin signo para la hora actual.

Parámetros de usuario

Overview

A user parameter lets you define a custom item key and the command used to retrieve its value when no built-in Zabbix agent item provides the data you need. A user parameter is a user-defined command referenced by a user-defined item key and executed by the Zabbix agent when the corresponding item is checked.

Note:
On UNIX systems, the agent executes user parameter commands using `/bin/sh`, under the operating system account running the agent (usually `zabbix`). That account must have sufficient permissions to execute the command and access any files, sockets, or credentials it uses.

There are two types of user parameters:

- **Simple user parameters** do not accept arguments; they define a fixed item key and a command to execute.
- **Flexible user parameters** accept arguments (specified as parameters in the item key) and pass them to the command, allowing a single definition to serve multiple items.

User parameter configuration

A user parameter is defined using the `UserParameter` configuration parameter.

A user parameter has the following syntax:

`UserParameter=<key>,<shell command>`

where:

- `<key>` - a unique item key that identifies the user parameter.
 When configuring an item, specify this key to reference the command defined by the user parameter.
 For a simple user parameter, the key is fixed and does not accept arguments. For a flexible user parameter, append `[*]` to the key (for example, `mysql.status[*]`) to indicate that it accepts arguments. Specify these arguments as parameters in the item key when configuring the item. The command can refer to the parameters using positional references $1–$9.
- `<shell command>` - a shell command or a path to a custom script that the agent executes to obtain the item value.
 For the flexible user parameter, the command can use positional references $1–$9 to refer to the corresponding parameters in the item key (for example, `UserParameter=mysql.status[*],mysqladmin status --user=$1 --password=$2`). Zabbix parses the parameters enclosed in `[]` in the item key (i.e., `mysql.status[zabbix,{$MYSQL.PASSWORD}]`) and

substitutes the positional references in the command accordingly.
 $0 is substituted with the original command before positional references are expanded. Positional references are interpreted regardless of whether they are enclosed in double (") or single (') quotes. To use positional references without substitution, specify a double dollar sign. For example, `awk '{print $$2}'` passes $2 to the command unchanged.
 Note that positional references prefixed with $ are searched for and replaced by the Zabbix agent only in flexible user parameters. For simple user parameters, positional references are not processed, so escaping $ is unnecessary.

Attention:
Certain characters cannot be passed as arguments by default. For the complete list of characters, see UnsafeUserParameters.

`UserParameter` must be configured on each Zabbix agent that uses it: - Directly in the Zabbix agent configuration file (`zabbix_agentd.conf` or `zabbix_agent2.conf`). - In a separate `*.conf` file, provided that the file is included by the agent configuration.

More than one `UserParameter` can be defined per agent, as multiple files can be included with different sets of parameters. Each user parameter must have a unique item key within the monitored host.

The `UserParameterDir` configuration parameter lets you specify the working directory for executing `UserParameter` commands. If configured, the agent changes its working directory to the specified directory before executing a command. Commands can be specified using a relative path with the `./` prefix, for example, `UserParameter=mykey,./myscript.sh`.

To configure a user parameter:

1. Write a custom script or command to retrieve the required data.
2. Add the command and its unique item key to the `UserParameter` configuration parameter in the Zabbix agent configuration file (`zabbix_agentd.conf`).
3. Reload user parameters from the configuration file by running the runtime control command:

   zabbix_agentd -R userparameter_reload

   Alternatively, restart the agent to apply the configuration changes.
4. Create a new item on the monitored host with the item key defined in `UserParameter`.
 Set the item's type to *Zabbix Agent* or *Zabbix Agent (active)*.
 Make sure that the item's *Type of information* is configured correctly; otherwise, Zabbix will reject the returned value.

See Extending Zabbix agents for an example of configuring and using user parameters.

User parameters are subject to the Zabbix agent check timeout. If the timeout is reached, the process executing the user parameter is terminated. See Command execution for more information.

Note that long-running scripts may consume system resources and delay other checks.

Resultado del comando

El valor de retorno del comando es una salida estándar junto con un error estándar producido por el comando.

Attention:
Un item que devuelve texto (carácter, log o tipo de información de texto) no se volverá no soportado en caso de una salida de error estándar.

El valor de retorno está limitado a 16MB (incluyendo los espacios en blanco finales que se truncan); también se aplican los límites de la base de datos.

Los parámetros de usuario que devuelven texto (carácter, log o tipo de información de texto) también pueden devolver un espacio en blanco. En caso de un resultado no válido, el item se volverá no soportado.

Ejemplos de parámetros de usuario simples

Un comando simple:

```
UserParameter=ping,echo 1
```

El agente siempre devolverá '1' para una métrica con la clave 'ping'.

Un ejemplo más complejo:

```
UserParameter=mysql.ping,mysqladmin -uroot ping | grep -c alive
```

El agente devolverá '1' si el servidor MySQL está activo, '0' en caso contrario.

Ejemplo 1

Algo muy simple:

```
UserParameter=ping[*],echo $1
```

Podemos definir un número ilimitado de métricas para monitorizar, todas con el formato ping[algo].

- ping[0] - siempre devolverá '0'
- ping[aaa] - siempre devolverá 'aaa'

Ejemplo 2

¡Vamos a añadir más sentido!

```
UserParameter=mysql.ping[*],mysqladmin -u$1 -p$2 ping | grep -c alive
```

Este parámetro puede usarse para monitorizar la disponibilidad de la base de datos MySQL. Podemos pasar el nombre de usuario y la contraseña:

```
mysql.ping[zabbix,our_password]
```

Ejemplo 3

¿Cuántas líneas coinciden con una expresión regular en un archivo?

```
UserParameter=wc[*],grep -c "$2" $1
```

Este parámetro puede usarse para calcular el número de líneas en un archivo.

```
wc[/etc/passwd,root]
wc[/etc/services,zabbix]
```

Ampliando Zabbix agent

Este tutorial proporciona instrucciones paso a paso sobre cómo ampliar la funcionalidad de Zabbix agent mediante el uso de un user parameter.

Paso 1

Escriba un script o una línea de comandos para recuperar el parámetro requerido.

Por ejemplo, podemos escribir el siguiente comando para obtener el número total de consultas ejecutadas por un server MySQL:

```
mysqladmin -uroot status | cut -f4 -d":" | cut -f1 -d"S"
```

Cuando se ejecuta, el comando devuelve el número total de consultas SQL.

Paso 2

Añada el comando al archivo de configuración del agent de Zabbix (`zabbix_agentd.conf`):

```
UserParameter=mysql.questions,mysqladmin -uroot status | cut -f4 -d":" | cut -f1 -d"S"
```

mysql.questions es un identificador único. Puede ser cualquier identificador de key válido, por ejemplo, *queries*.

Pruebe este parámetro utilizando el agent de Zabbix con la opción `-t` (sin embargo, si se ejecuta como root, tenga en cuenta que el agent puede tener permisos diferentes cuando se inicia como daemon):

```
zabbix_agentd -t mysql.questions
```

Paso 3

Recargue los parámetros de usuario desde el archivo de configuración ejecutando:

```
zabbix_agentd -R userparameter_reload
```

También puede reiniciar el agent en lugar de usar el comando de control en tiempo de ejecución.

Pruebe el parámetro utilizando la utilidad `zabbix_get`.

Paso 4

Agregue un nuevo item con `Key=mysql.questions` al host monitorizado. El tipo del item debe ser *Zabbix Agent* o *Zabbix Agent (active)*.

Tenga en cuenta que el tipo de los valores devueltos debe configurarse correctamente en el server de Zabbix. De lo contrario, Zabbix no los aceptará.

Contadores de rendimiento de Windows

Descripción general

Puede supervisar eficazmente los contadores de rendimiento de Windows mediante la clave `perf_counter[]`.

Por ejemplo:

```
perf_counter["\Processor(0)\Interrupts/sec"]
```

o

```
perf_counter["\Processor(0)\Interrupts/sec", 10]
```

Para obtener más información sobre el uso de esta clave o de su equivalente exclusivo en inglés, `perf_counter_en`, consulte Claves de item específicas de Windows.

Para obtener una lista completa de los contadores de rendimiento disponibles para la supervisión, puede ejecutar:

```
typeperf -qx
```

También puede utilizar el descubrimiento de bajo nivel para descubrir varias instancias de objetos de los contadores de rendimiento de Windows y automatizar la creación de items `perf_counter` para objetos de varias instancias.

Representación numérica

Windows mantiene representaciones numéricas (índices) para los nombres de objetos y contadores de rendimiento. Zabbix admite estas representaciones numéricas como parámetros para las claves de item `perf_counter`, `perf_counter_en` y en los parámetros de configuración `PerfCounter`, `PerfCounterEn`.

Sin embargo, no se recomienda utilizarlas a menos que pueda garantizar que sus índices numéricos se corresponden con las cadenas correctas en hosts específicos. Si necesita crear items portables que funcionen en distintos hosts con varias versiones localizadas de Windows, puede utilizar la clave `perf_counter_en` o el parámetro de configuración `PerfCounterEn`, que permiten utilizar nombres en inglés independientemente de la configuración regional del sistema.

Para averiguar los equivalentes numéricos, ejecute `regedit` y, a continuación, localice *Counter* en `HKEY_LOCAL_MACHINE\SOFTWARE\Micr` `NT\CurrentVersion\Perflib\009`.

La entrada del registro contiene información como la siguiente:

```
1
1847
2
System
4
Memory
6
% Processor Time
10
File Read Operations/sec
12
File Write Operations/sec
14
File Control Operations/sec
16
File Read Bytes/sec
18
File Write Bytes/sec
....
```

Aquí puede encontrar los números correspondientes a cada parte de cadena del contador de rendimiento, como en `\System\% Processor Time`:

```
System → 2
% Processor Time → 6
```

A continuación, puede utilizar estos números para representar la ruta mediante números:

```
\2\6
```

Parámetros de contadores de rendimiento

Puede implementar algunos parámetros de PerfCounter para supervisar los contadores de rendimiento de Windows.

Por ejemplo, puede añadirlos al archivo de configuración del agent de Zabbix:

```
PerfCounter=UserPerfCounter1,"\Memory\Page Reads/sec",30
or
PerfCounter=UserPerfCounter2,"\4\24",30
```

Una vez configurados estos parámetros, puede utilizar simplemente `UserPerfCounter1` o `UserPerfCounter2` como claves para crear los items correspondientes.

Recuerde reiniciar el agent de Zabbix después de realizar cambios en el archivo de configuración.

Actualización masiva

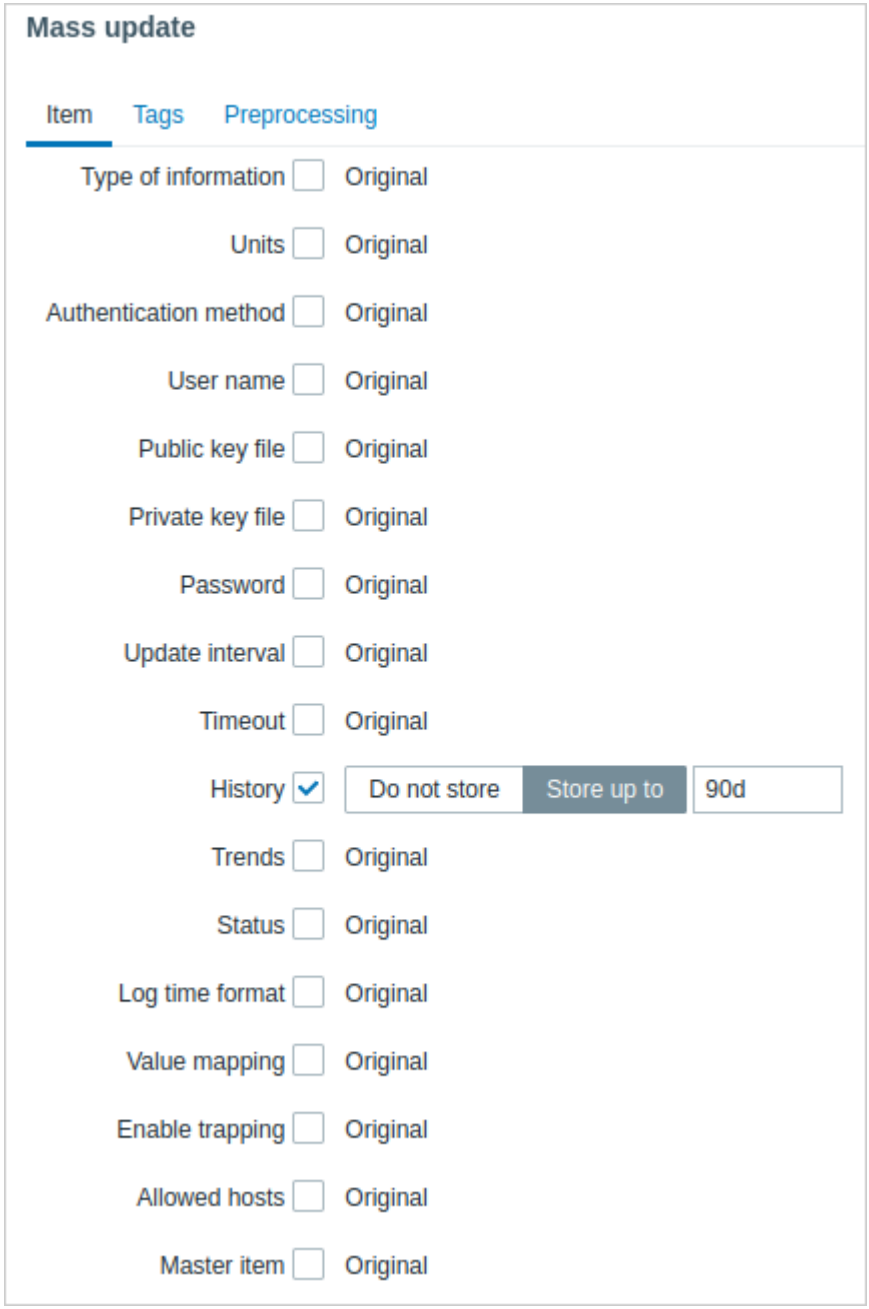
Descripción general

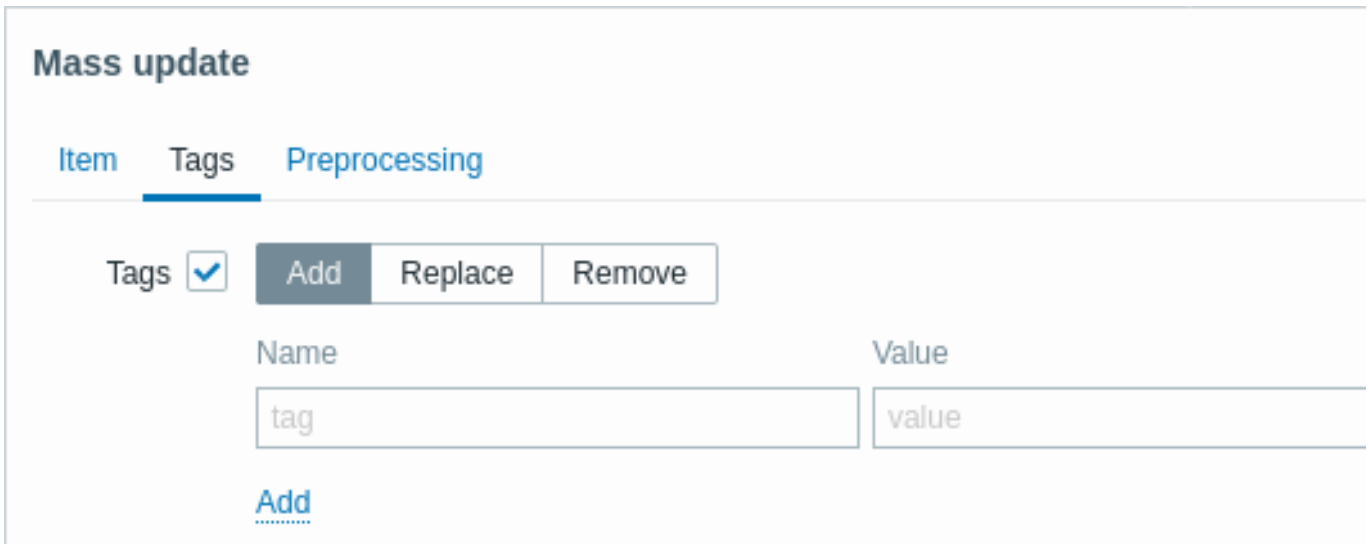
A veces puede que desee cambiar algún atributo para varias métricas a la vez. En lugar de abrir cada métrica individualmente para editarla, puede utilizar la función de actualización masiva para ello.

Uso de la actualización masiva

Para actualizar varios items de forma masiva, haga lo siguiente:

- Marque las casillas de los items que desea actualizar en la lista
- Haga clic en *Actualización masiva* debajo de la lista
- Vaya a la pestaña con los atributos necesarios (*Item*, *Tags* o *Preprocessing*)
- Marque las casillas de los atributos que desea actualizar
- Introduzca nuevos valores para los atributos



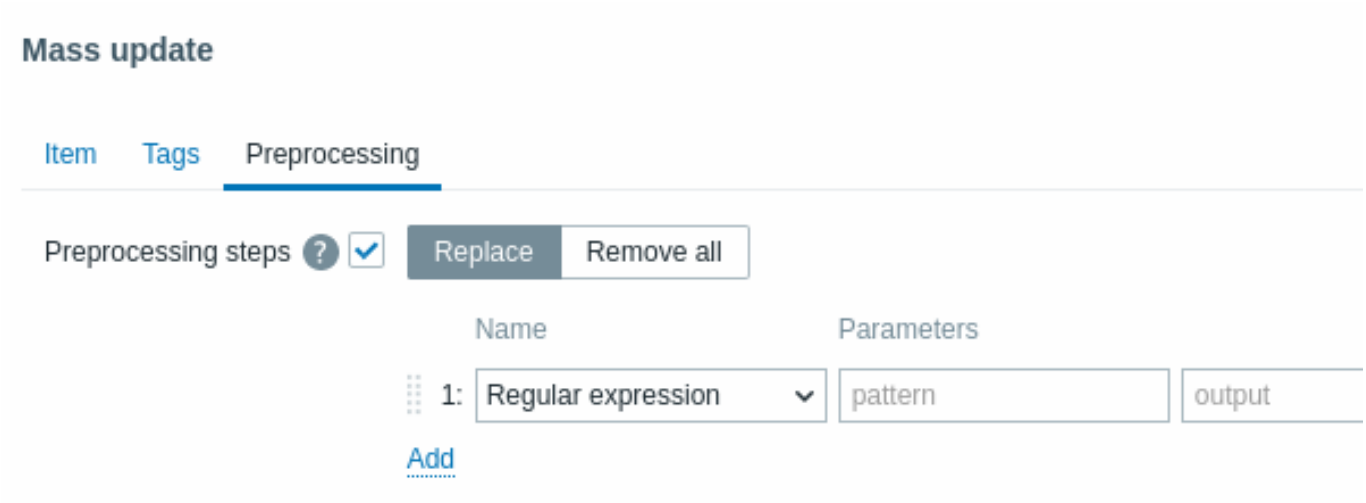


La opción *Tags* permite:

- *Add*: añadir las etiquetas especificadas a los items (las etiquetas con el mismo nombre, pero con valores diferentes no se consideran «duplicadas» y se pueden añadir al mismo item).
- *Replace*: eliminar las etiquetas especificadas y añadir etiquetas con valores nuevos

- *Remove*: eliminar las etiquetas especificadas de los items

Las macros de usuario, las macros {INVENTORY.*}, {HOST.HOST}, {HOST.NAME}, {HOST.CONN}, {HOST.DNS}, {HOST.IP}, {HOST.PORT} y {HOST.ID} son compatibles con las etiquetas.



Marque la casilla situada junto a *Preprocessing steps* para activar la actualización masiva de los pasos de preprocesamiento.

La actualización masiva de *Preprocessing* permite:

- *Replace*: reemplazar los pasos de preprocesamiento existentes en los items por los pasos de preprocesamiento especificados a continuación
- *Remove all*: eliminar todos los pasos de preprocesamiento de los items

Cuando haya terminado, haga clic en *Update*.

Mapeo de valores

Descripción general

El mapeo de valores permite configurar una representación más amigable para el usuario de los valores recibidos utilizando asignaciones entre valores numéricos/cadena y representaciones de cadena.

Por ejemplo, cuando el valor de una métrica es ”0” o ”1”, se pueden usar mapeos de valores para representar estos valores de una manera más amigable para el usuario:

- 0 → No disponible
- 1 → Disponible

Los mapeos de valores para tipos de respaldo de datos podrían configurarse de la siguiente manera:

- F → Completo
- D → Diferencial
- I → Incremental

Los rangos de valores para voltaje podrían configurarse de la siguiente manera:

- <=209 → Bajo
- 210-230 → OK
- >=231 → Alto

El mapeo de valores se utiliza en el frontend de Zabbix y en las notificaciones enviadas por tipos de medios.

Attention:
La sustitución del valor recibido por la representación configurada se realiza tanto en el frontend de Zabbix como en el servidor; sin embargo, el servidor maneja la sustitución solo en los siguientes casos:

- al completar el inventario de equipos;
- al expandir macros soportadas {ITEM.VALUE}, {ITEM.LASTVALUE}, {EVENT.OPDATA} y {EVENT.CAUSE.OPDATA}.

Los mapeos de valores se configuran en plantillas o equipos. Una vez configurados, están disponibles para todas las métricas dentro de la respectiva plantilla o equipo. Al configurar métricas, especifique el nombre de un mapeo de valores previamente configurado en el parámetro *Mapeo de valores*.

Note:
No existe herencia de mapeo de valores: los equipos y plantillas no heredan mapeos de valores de las plantillas enlazadas. Las métricas de plantilla en un equipo continuarán utilizando los mapeos de valores configurados en la plantilla.

Note:
Los mapeos de valores pueden usarse con métricas que tengan tipos de información *Numérico (sin signo)*, *Numérico (coma flotante)* y *Carácter*.

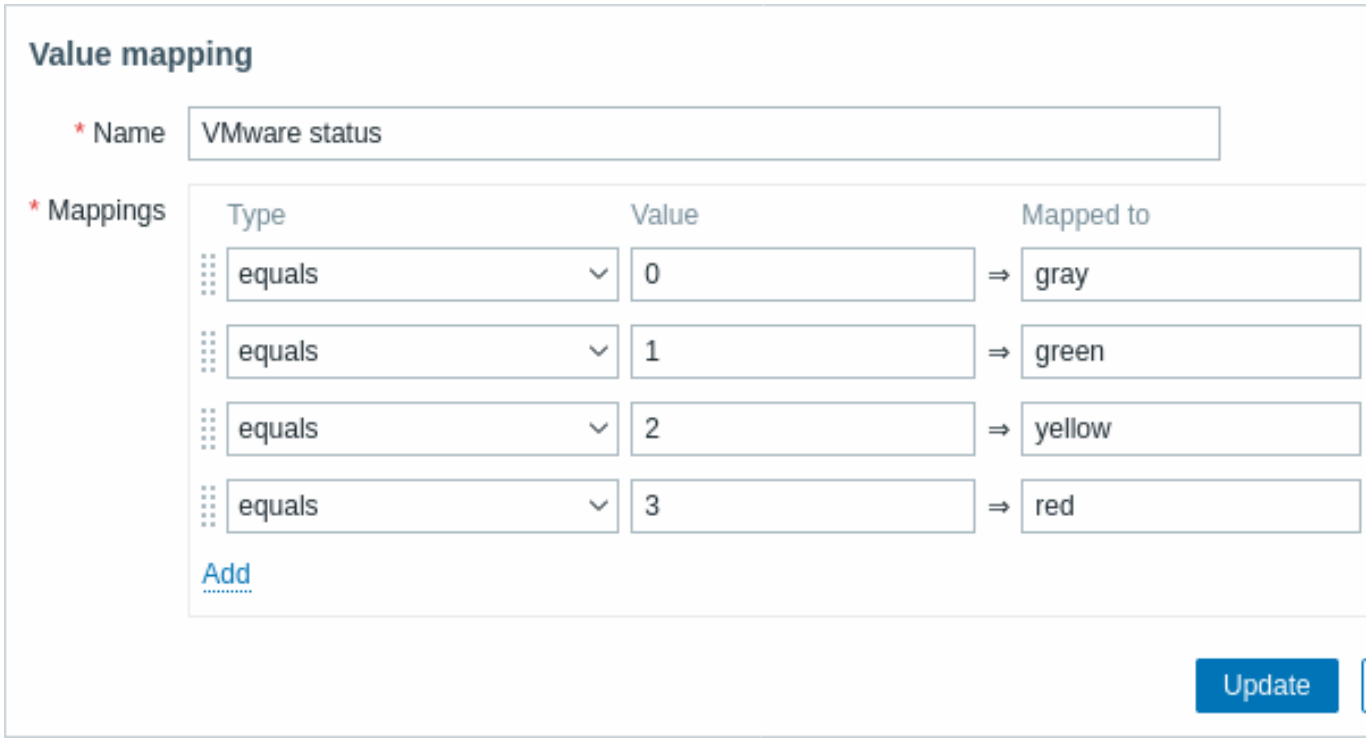
Los mapeos de valores se exportan/importan con la respectiva plantilla o equipo. También pueden actualizarse masivamente utilizando los formularios de actualización masiva de equipo y plantilla.

Configuración

Para configurar una asignación de valores, siga estos pasos:

1. Abra el formulario de configuración del host o template.

2. En la pestaña *Asignación de valores*, haga clic en *Añadir* para añadir una nueva asignación de valores, o haga clic en el nombre de una asignación existente para editarla.



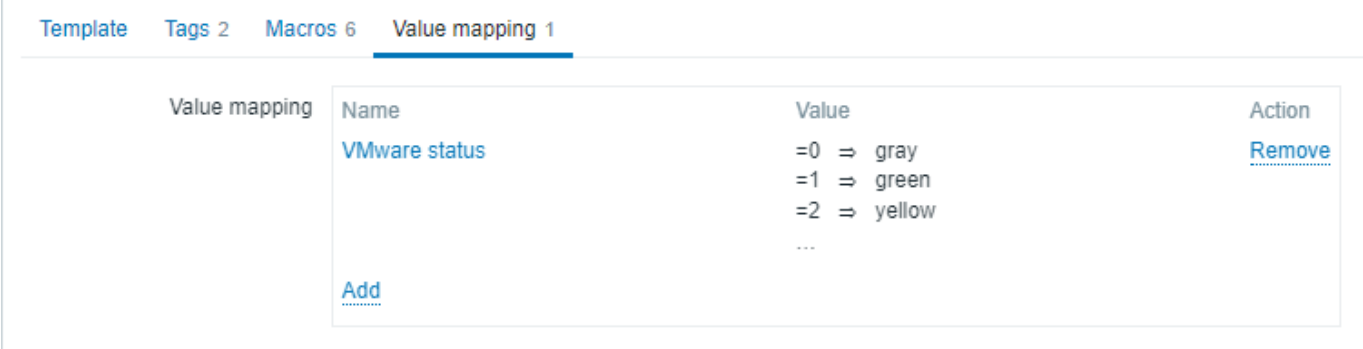
Parámetros de una asignación de valores:

Parámetro	Descripción
Nombre	Nombre único para el conjunto de asignaciones de valores.
Asignaciones	Reglas individuales para asignar valores numéricos o de cadena a representaciones de cadena. La asignación se aplica en el orden de las reglas, que se puede cambiar arrastrándolas.

Parámetro	Descripción
Tipo	Tipo de asignación: **equals** - se asignarán los valores iguales; **is greater than or equals** - se asignarán los valores iguales o mayores; **is less than or equals** - se asignarán los valores iguales o menores; **in range** - se asignarán los valores dentro del rango; el rango se expresa como <number1>-<number2> o <number>; se admiten varios rangos (por ejemplo, 1-10,101-110,201); **regexp** - se asignarán los valores que correspondan a la expresión regular (no se admiten expresiones regulares globales); **default** - se asignarán todos los valores restantes, excepto aquellos con asignaciones específicas. Para asignar rangos, solo se admiten tipos de valores numéricos (*is greater than or equals*, *is less than or equals*, *in range*).
Valor	Valor entrante (puede contener un rango o una expresión regular, según el tipo de asignación).
Asignado a	Representación de cadena (de hasta 64 caracteres) para el valor entrante.

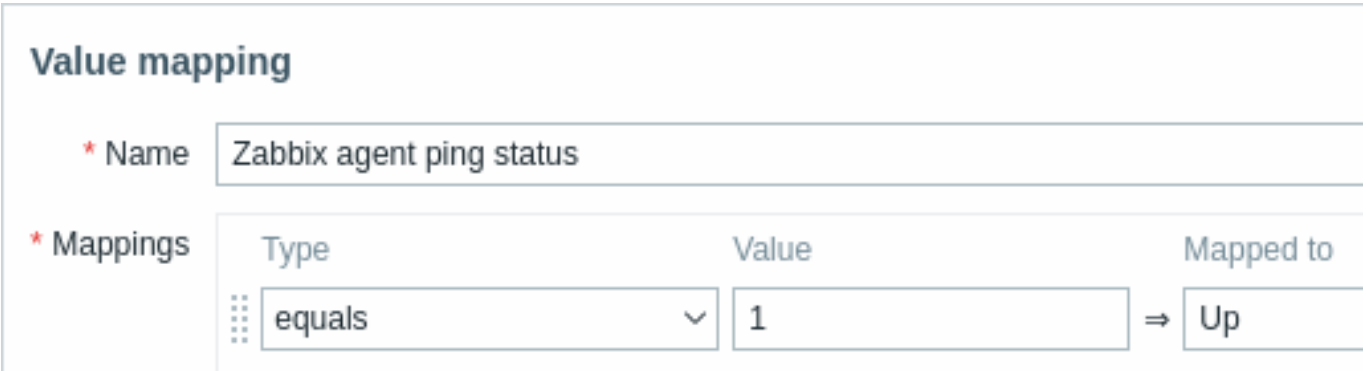
Todos los campos de entrada obligatorios están marcados con un asterisco rojo.

Al visualizar la asignación de valores en la lista, solo se muestran las tres primeras asignaciones; los tres puntos indican que existen más asignaciones.



Ejemplo de asignación de valores

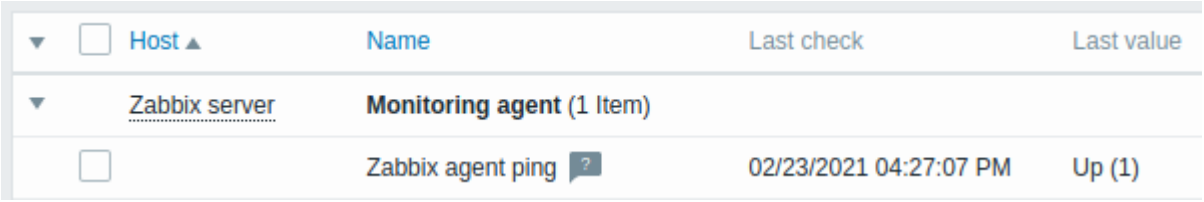
Uno de los items de agent predefinidos, *Zabbix agent ping*, utiliza una asignación de valores de nivel de template denominada "Zabbix agent ping status" para mostrar sus valores.



En el formulario de configuración del item, puede encontrar una referencia a esta asignación de valores en el campo *Value mapping*:

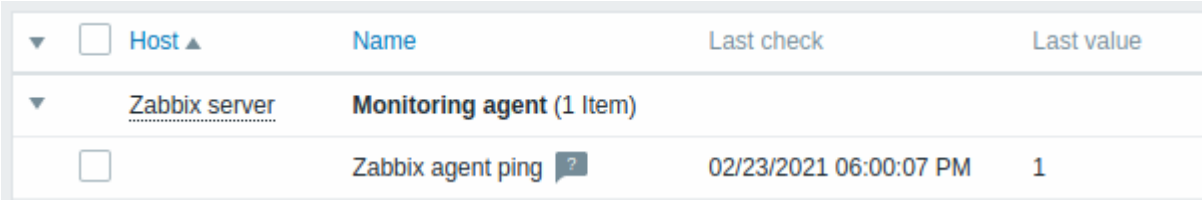


Esta asignación se utiliza en la sección *Monitoring* > *Latest data* para mostrar "Up" (con el valor sin procesar entre paréntesis).



Note:
En la sección *Latest data*, los valores mostrados se acortan a 20 símbolos. Si se utiliza una asignación de valores, este acortamiento no se aplica al valor asignado, sino únicamente al valor sin procesar (mostrado entre paréntesis).

Sin una asignación de valores predefinida, solo vería "1", lo que podría resultar difícil de entender.



Cola

Descripción general

La cola muestra las métricas que están esperando una actualización. La cola es solo una representación **lógica** de los datos. No existe una cola IPC ni ningún otro mecanismo de cola en Zabbix.

Las métricas monitorizadas por proxies también se incluyen en la cola; se contarán como en cola durante el período de actualización de datos históricos del proxy.

Solo se muestran en la cola las métricas con tiempos de actualización programados. Esto significa que los siguientes tipos de métricas se excluyen de la cola:

- métricas activas de agente Zabbix de tipo log, logrt y event log
- métricas SNMP trap
- métricas trapper
- métricas de monitorización web
- métricas dependientes

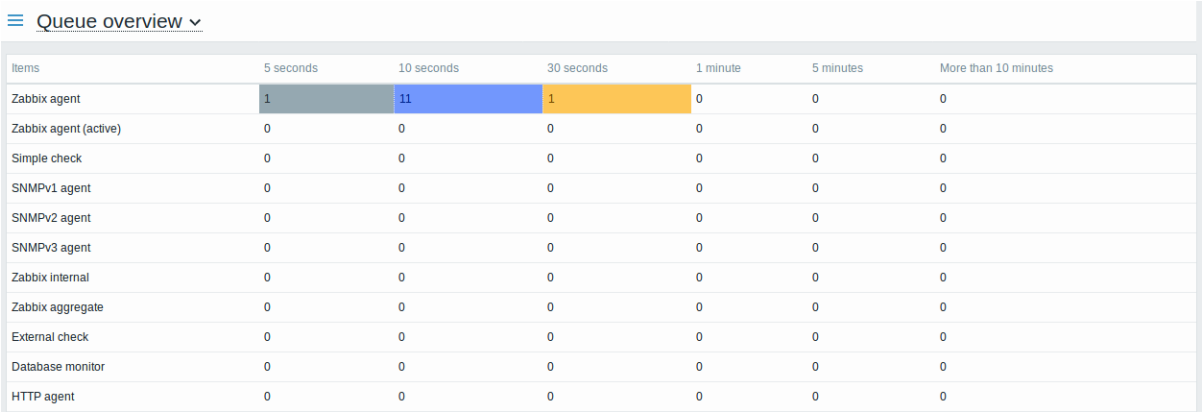
Las estadísticas mostradas por la cola son un buen indicador del rendimiento del servidor Zabbix.

La cola se obtiene directamente del servidor Zabbix utilizando el protocolo JSON. La información solo está disponible si el servidor Zabbix está en funcionamiento.

Attention:
Las métricas no se cuentan en la cola si la interfaz de la métrica se vuelve inaccesible debido a problemas de conexión o a que el agente no funciona correctamente.

Lectura de la cola

Para leer la cola, vaya a *Administración* > *Cola* > *Descripción general de la cola*.



La imagen que se muestra aquí es, en general, «correcta», por lo que podemos suponer que el server funciona correctamente.

La cola muestra algunos items que están esperando hasta 30 segundos. Sería útil saber cuáles son esos items.

Para ello, seleccione *Detalles de la cola* en la lista desplegable del título. Ahora podrá ver una lista de esos items retrasados.

Queue details

Scheduled check	Delayed by	Host	Name	Proxy
2019-09-02 11:46:40	58s	My host	CPU idle time	Remote proxy
2019-09-02 11:46:41	57s	My host	CPU interrupt time	Remote proxy
2019-09-02 11:46:42	56s	My host	CPU iowait time	Remote proxy
2019-09-02 11:46:43	55s	My host	CPU nice time	Remote proxy
2019-09-02 11:46:44	54s	My host	CPU softirq time	Remote proxy
2019-09-02 11:46:45	53s	My host	CPU steal time	Remote proxy
2019-09-02 11:46:46	52s	My host	CPU system time	Remote proxy

Con estos detalles, es posible averiguar por qué estos items podrían estar retrasados.

Si hay uno o dos items retrasados, quizá no haya motivo de alarma. Es posible que se actualicen en un segundo. Sin embargo, si observa que muchos items se retrasan durante demasiado tiempo, podría existir un problema más grave.

Véase también: Alineación de las zonas horarias al utilizar intervalos de programación.

Queue overview

Items	5 seconds	10 seconds	30 seconds	1 minute	5 minutes	More than 10 minutes
Zabbix agent	0	1	1	26	0	0
Zabbix agent (active)	0	0	0	0	0	0
Simple check	0	0	0	0	0	0
SNMPv1 agent	0	0	0	0	0	0
SNMPv2 agent	0	0	0	0	0	0
SNMPv3 agent	0	0	0	0	0	0
Zabbix internal	0	0	0	0	0	0
Zabbix aggregate	0	0	0	0	0	0
External check	0	0	0	0	0	0
Database monitor	0	0	0	0	0	0
HTTP agent	0	0	0	0	0	0
IPMI agent	0	0	0	0	0	0
SSH agent	0	0	0	0	0	0
TELNET agent	0	0	0	0	0	0
JMX agent	0	0	0	0	0	0
Calculated	0	0	0	0	0	0

item de cola

Se puede utilizar un item interno especial `zabbix[queue,<from>,<to>]` para supervisar el estado de la cola en Zabbix. Devolverá el número de items retrasados por la cantidad de tiempo establecida. Para obtener más información, consulte Items internos.

Caché de valores

Descripción general

Para hacer que el cálculo de expresiones de disparadores, métricas calculadas y algunas macros sea mucho más rápido, el servidor Zabbix admite una opción de caché de valores.

Esta caché en memoria puede usarse para acceder a datos históricos, en lugar de realizar llamadas SQL directas a la base de datos. Si los valores históricos no están presentes en la caché, los valores faltantes se solicitan a la base de datos y la caché se actualiza en consecuencia.

Los valores de la métrica permanecen en la caché de valores hasta que:

- la métrica es eliminada (los valores en caché se eliminan después de la siguiente sincronización de configuración);
- el valor de la métrica está fuera del rango de tiempo o de cantidad especificado en la expresión del disparador/métrica calculada (el valor en caché se elimina cuando se recibe un nuevo valor);
- el rango de tiempo o de cantidad especificado en la expresión del disparador/métrica calculada se cambia de modo que se requieran menos datos para el cálculo (los valores en caché innecesarios se eliminan después de 24 horas).

Note:
El estado de la caché de valores puede observarse utilizando la opción de control en tiempo de ejecución del servidor `diaginfo` (o `diaginfo=valuecache`) e inspeccionando la sección de información de diagnóstico de la caché de valores. Esto puede ser útil para determinar disparadores o métricas calculadas mal configuradas.

Para habilitar la funcionalidad de caché de valores, se puede especificar el parámetro opcional `ValueCacheSize` en la configuración del server de Zabbix.

Se admiten dos items internos para supervisar la caché de valores: `zabbix[vcache,buffer,<mode>]` y `zabbix[vcache,cache,<parame`
Consulte más detalles sobre los items internos.

Ejecutar ahora

Descripción general

La opción *Ejecutar ahora* permite ejecutar comprobaciones pasivas de inmediato.

La recopilación de valores de elementos en Zabbix sigue los intervalos de actualización configurados. Algunos elementos (por ejemplo, reglas de descubrimiento de bajo nivel) tienen intervalos largos y puede que necesite obtener un nuevo valor de inmediato, por ejemplo, para detectar cambios en los recursos descubribles.

La opción *Ejecutar ahora* es compatible con los siguientes tipos de elementos:

- Agente Zabbix (pasivo)
- Comprobación simple
- Agente SNMP (v1/v2/v3)
- Interno de Zabbix
- Comprobación externa
- Monitor de base de datos
- Agente HTTP
- Agente IPMI
- Agente SSH
- Agente TELNET
- Agente JMX
- Calculado
- Elemento dependiente
- Script
- Navegador

Si el elemento es de tipo ”Elemento dependiente”, entonces su elemento principal también debe ser uno de los tipos anteriores.

Configuración

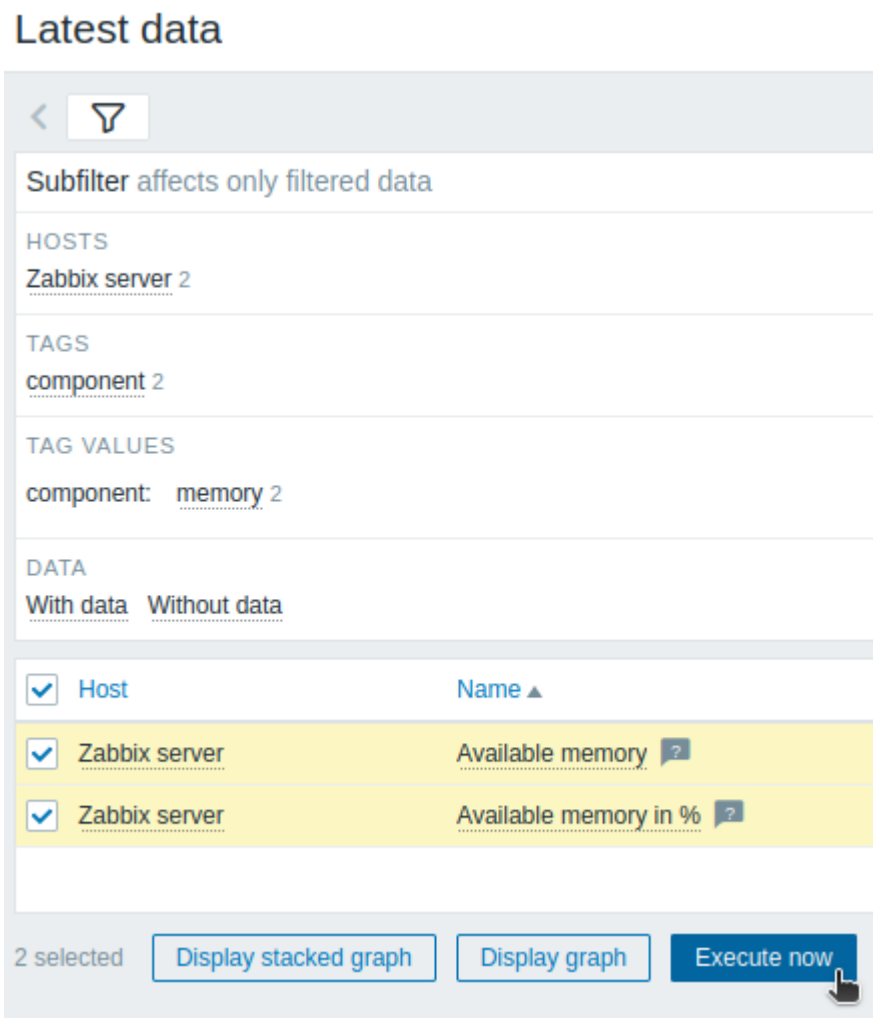
Puede ejecutar comprobaciones pasivas desde:

- *Monitoring* > *Latest data*
- *Data collection* > *Hosts* > *Items* o *Discovery*
- El menú de item
- Al editar items o reglas de descubrimiento de bajo nivel (en sus formularios de configuración)

Attention:
La comprobación debe existir en la caché de configuración para poder ejecutarse; consulte también CacheUpdateFrequency. La caché **no se actualiza** antes de la ejecución de la comprobación, por lo que los cambios recientes en la configuración de un item o de una regla de descubrimiento de bajo nivel no se tendrán en cuenta. Para probar items o reglas LLD que se estén creando o que se hayan creado recientemente, puede usar la opción Test en su lugar.

Por ejemplo, para ejecutar una comprobación pasiva en *Monitoring* > *Latest data*:

1. Seleccione items; puede ejecutar la comprobación para varios items a la vez.
2. Haga clic en *Execute now*.



Note:
En *Latest data*, los usuarios pueden *Execute now* en hosts para los que tengan permiso de lectura y escritura, o permiso de lectura y la acción *Invoke "Execute now" on read-only hosts* habilitada para su rol.

Restricción de comprobaciones de agent

Resumen

Puede controlar qué claves de item Zabbix agent o agent 2 tiene permitido o denegado usar al ejecutar comprobaciones de item, comandos remotos o scripts.

Para ello, use estos parámetros de configuración del agent para definir reglas de □□□□□□/denegación:

- `AllowKey=<pattern>`
- `DenyKey=<pattern>`
- `AllowKeyRegexp=<pattern>`
- `DenyKeyRegexp=<pattern>`

El `<pattern>` debe contener una sola clave de item.
 Para `AllowKey` y `DenyKey`, `<pattern>` admite comodines (`*`). El comodín coincide con cualquier número de caracteres en su posición y puede usarse para coincidir con claves de item o parámetros (por ejemplo, `vfs.file.*[*]`).
 Para `AllowKeyRegexp` y `DenyKeyRegexp`, `<pattern>` admite expresiones regulares. El patrón de expresión regular puede usarse para coincidir con claves de item y sus cadenas de parámetros. Consulte también ejemplos de escape correcto.
 Tenga en cuenta que Zabbix agent y Zabbix agent 2 pueden admitir una sintaxis de expresiones regulares diferente.

Attention:
Para mejorar la seguridad, se recomienda usar claves de item exactas en lugar de comodines para `AllowKey` y `DenyKey`. También se recomienda usar los patrones de expresión regular más específicos para `AllowKeyRegexp` y `DenyKeyRegexp`. Para más detalles, consulte Protección de las reglas de □□□□□□/denegación.

A diferencia de otros parámetros de configuración del agent, puede especificar una cantidad ilimitada de parámetros AllowKey, DenyKey, AllowKeyRegexp o DenyKeyRegexp.

Notas importantes

- Todos los items system.run están deshabilitados de forma predeterminada (incluso cuando DenyKey y DenyKeyRegexp están vacíos), como si se hubiera establecido DenyKey=system.run[*] o DenyKeyRegexp=^system\.run\[.*\]$ como la última regla. Debido a esto, puede permitir items específicos de system.run sin denegar explícitamente otros items de system.run.
- Si es posible, use AllowKey/AllowKeyRegexp para permitir solo los items necesarios y denegar todo lo demás. Algunas claves pueden aprovecharse para leer archivos no previstos mediante path traversal (por ejemplo, vfs.file.contents["../../../
y las nuevas versiones de Zabbix agent pueden introducir claves no cubiertas por sus reglas DenyKey/DenyKeyRegexp.
- La configuración AllowKey, DenyKey, AllowKeyRegexp y DenyKeyRegexp no afecta a los parámetros de agent HostnameItem, HostMetadataItem ni HostInterfaceItem.
- Los items denegados pasan a no ser compatibles sin ninguna indicación ni mensaje de error; por ejemplo:
  - El parámetro de línea de comandos --print (-p) de Zabbix agent no mostrará las claves de item denegadas.
  - El parámetro de línea de comandos --test (-t) de Zabbix agent devolverá "Unsupported item key." para las claves de item denegadas.
  - Cuando la registración está activada (LogRemoteCommands=1), el archivo de registro de Zabbix agent no registrará los comandos remotos denegados.

Orden de reglas allow/deny

Puede especificar un número ilimitado de reglas AllowKey, DenyKey, AllowKeyRegexp o DenyKeyRegexp, aunque su orden importa.

- Las reglas se evalúan una por una, de arriba hacia abajo.
- Cuando una clave de item coincide con una regla, se permite o se deniega, y la evaluación de reglas se detiene.

Por ejemplo, al evaluar vfs.file.contents[/etc/passwd], las reglas se procesan de la siguiente manera:

```
AllowKey=vfs.file.contents[/tmp/app.log]    # El patrón de la clave de item no coincide; el agent pasa a l
AllowKey=vfs.file.contents[/etc/passwd]     # El patrón de la clave de item coincide; el agent permite la
DenyKey=vfs.file.*[*]                       # El agent ignora la regla, ya que la evaluación se ha detenid
```

El siguiente orden de reglas denegará la comprobación del item:

```
DenyKey=vfs.file.*[*]                       # El patrón de la clave de item coincide; el agent deniega la
AllowKey=vfs.file.contents[/etc/passwd]     # El agent ignora la regla, ya que la evaluación se ha detenid
AllowKey=vfs.file.contents[/tmp/app.log]    # El agent ignora la regla, ya que la evaluación se ha detenid
```

Ejemplos

Los siguientes ejemplos muestran patrones de configuración comunes para AllowKey, DenyKey, AllowKeyRegexp y DenyKeyRegexp.

Permitir comprobaciones y comandos específicos

Permita solo dos comprobaciones de item vfs.file y dos comandos system.run:

usando AllowKey y DenyKey:

```
AllowKey=vfs.file.contents[/tmp/app.log]
AllowKey=vfs.file.size[/tmp/app.log]
AllowKey=system.run[/usr/bin/uptime]
AllowKey=system.run[/usr/bin/df -h /]
DenyKey=vfs.file.*[*]
```

usando AllowKeyRegexp y DenyKeyRegexp:

```
AllowKeyRegexp=^vfs\.file\.(contents|size)\[/tmp/app\.log\]$
AllowKeyRegexp=^system\.run\[/usr/bin/(uptime|df -h /)\]$
DenyKeyRegexp=^vfs\.file\..*\[.*\]$
```

Note:
No es necesario establecer DenyKey=system.run[*] ni DenyKeyRegexp=^system\.run\[.*\]$, porque todos los demás comandos system.run se deniegan de forma predeterminada.

Permitir scripts

Permita que Zabbix agent ejecute scripts en los hosts mediante todos los métodos disponibles:

- Scripts globales que pueden ejecutarse en el frontend o mediante API (este método siempre usa la clave `system.run[myscript.sh]`)
- Comandos remotos de operaciones de acción (este método siempre usa la clave `system.run[myscript.sh,nowait]`)
- Elementos `system.run` de Zabbix agent con el script, por ejemplo:
  - `system.run[myscript.sh]`
  - `system.run[myscript.sh,wait]`
  - `system.run[myscript.sh,nowait]`

```
AllowKey=system.run[myscript.sh,*]
```

o

```
AllowKeyRegexp=^system\.run\[myscript\.sh,.*\]$
```

Para controlar el parámetro `wait/nowait`, debe establecer una regla diferente. Por ejemplo, puede permitir solo elementos `system.run[myscript.sh,wait]`, excluyendo así otros métodos:

```
AllowKey=system.run[myscript.sh,wait]
```

o

```
AllowKeyRegexp=^system\.run\[myscript\.sh,wait\]$
```

Protección de reglas allow/deny

Estos ejemplos muestran cómo proteger reglas `AllowKey/AllowKeyRegexp` o `DenyKey/DenyKeyRegexp` demasiado permisivas.

Considere las siguientes reglas:

usando `AllowKey` y `DenyKey`:

```
AllowKey=system.run["C:\Program^ Files\Zabbix^ Agent^ 2\scripts\test.bat*"]
DenyKey=vfs.file.*
DenyKey=system.cpu.load[*]
```

usando `AllowKeyRegexp` y `DenyKeyRegexp`:

```
AllowKeyRegexp=^system\.run\["C:\\Program\sFiles\\Zabbix\sAgent\s2\\scripts\\test\.bat.*"\]$
DenyKeyRegexp=^vfs\.file\..*$
DenyKeyRegexp=^system\.cpu\.load\[.*\]$
```

Note:
En Windows, debe escapar los espacios en la ruta usando un caret (^) para `AllowKey` o `DenyKey` y `\s` para `AllowKeyRegexp` o `DenyKeyRegexp`.

Estas reglas contienen un comodín (`*`) (o `.*` para patrones basados en expresiones regulares), que puede ser mal utilizado:

- El script `test.bat` puede ejecutarse con cualquier argumento, incluidos los no deseados.
- El patrón `vfs.file.*` (`^vfs\.file\..*$`) coincide con claves de item tanto con parámetros como sin ellos; sin embargo, todos los items `vfs.file` requieren parámetros.
- El patrón `system.cpu.load[*]` (`^system\.cpu\.load\[.*\]$`) coincide solo con claves de item con parámetros; sin embargo, los items `system.cpu.load` no requieren parámetros.

Para proteger estas reglas, permita explícitamente la ejecución de `test.bat` solo con argumentos específicos y deniegue patrones correctos de claves de item; por ejemplo:

usando `AllowKey` y `DenyKey`:

```
AllowKey=system.run["C:\Program^ Files\Zabbix^ Agent^ 2\scripts\test.bat status"]
AllowKey=system.run["C:\Program^ Files\Zabbix^ Agent^ 2\scripts\test.bat version"]
DenyKey=vfs.file.*[*]
DenyKey=system.cpu.load
DenyKey=system.cpu.load[*]
```

usando `AllowKeyRegexp` y `DenyKeyRegexp`:

```
AllowKeyRegexp=^system\.run\["C:\\Program\sFiles\\Zabbix\sAgent\s2\\scripts\\test\.bat (status|version)"\]
DenyKeyRegexp=^vfs\.file\..+\[.*\]$
DenyKeyRegexp=^system\.cpu\.load(\[.*\])?$
```

Puede probar las reglas ejecutando los siguientes comandos, que devolverán ZBX_NOTSUPPORTED.

```
cd "C:\Program Files\Zabbix Agent 2"
zabbix_agent2.exe -t system.run["C:\Program^ Files\Zabbix^ Agent^ 2\scripts\test.bat debug"]
zabbix_agent2.exe -t vfs.file.size["C:\ProgramData\MyApp\config.ini"]
zabbix_agent2.exe -t vfs.file.contents["C:\Windows\System32\drivers\etc\hosts"]
zabbix_agent2.exe -t system.cpu.load
zabbix_agent2.exe -t system.cpu.load[all,avg1]
```

Ejemplos de patrones

Las siguientes tablas muestran cómo se comparan los patrones de clave de item:

- Una clave coincide con el patrón solo si cumple **todas** las condiciones de la columna *Matches*.
- Los parámetros deben estar completamente encerrados entre corchetes (por ejemplo, `vfs.file.contents[*` y `vfs.file.contents*utf8]` son patrones no válidos).
- Los ejemplos ilustran solo la coincidencia de patrones; los items reales requieren parámetros.

Para AllowKey y DenyKey:

Pattern	Matches	Examples
*	Cualquier clave con o sin parámetros	
vfs.file.*	La clave comienza con vfs.file. Con o sin parámetros	Matches: vfs.file.size vfs.file.contents vfs.file.contents[] vfs.file.size[/var/log/app.log]
vfs.*.contents	La clave comienza con vfs. La clave termina con .contents Sin parámetros	Matches: vfs..contents vfs.mount.point.file.contents Does not match: vfs.contents vfs.file.contents[]
vfs.file.*[*]	La clave comienza con vfs.file. Parámetros cualesquiera o vacíos	Matches vfs.file.get.custom[] vfs.file.size[/var/log/app.log, utf8] Does not match: vfs.file.get.custom
vfs.file.contents	La clave es vfs.file.contents Sin parámetros	Matches: vfs.file.contents Does not match: vfs.file.contents[/etc/passwd]
vfs.file.contents[]	La clave es vfs.file.contents[] Parámetros vacíos	Matches: vfs.file.contents[] Does not match: vfs.file.contents
vfs.file.contents[*]	La clave es vfs.file.contents Parámetros cualesquiera o vacíos	Matches: vfs.file.contents[/path/to/file] Does not match: vfs.file.contents
vfs.file.contents[/etc/passwd,*]	La clave es vfs.file.contents El primer parámetro es /etc/passwd El segundo parámetro es cualquiera o vacío	Matches: vfs.file.contents[/etc/passwd,] vfs.file.contents[/etc/passwd,utf8] Does not match: vfs.file.contents[] vfs.file.contents[/etc/passwd]

Pattern	Matches	Examples
vfs.file.contents[*passwd*]	La clave es vfs.file.contents Parámetros cualesquiera, al menos uno contiene passwd	Matches: vfs.file.contents[/etc/passwd] vfs.file.contents[/etc/passwd,] vfs.file.contents[/etc/passwd,utf8]
vfs.file.contents[*passwd*,*]	La clave es vfs.file.contents El primer parámetro contiene passwd El segundo parámetro es cualquiera o vacío	Matches: vfs.file.contents[/etc/passwd,] vfs.file.contents[/etc/passwd,utf8] Does not match: vfs.file.contents[/etc/passwd] vfs.file.contents[/tmp/test]
vfs.file.contents[/etc/passwd,utf8]	La clave es vfs.file.contents El primer parámetro es /etc/passwd El segundo parámetro es utf8	Matches: vfs.file.contents[/etc/passwd,utf8] Does not match: vfs.file.contents[/etc/passwd,] vfs.file.contents[/etc/passwd,utf16]

Para AllowKeyRegexp y DenyKeyRegexp:

Pattern	Matches	Examples
^.*$	Cualquier clave con o sin parámetros	
^vfs\..*$	La clave comienza con vfs.file. Con o sin parámetros	Matches: vfs.file.size vfs.file.contents vfs.file.contents[] vfs.file.size[/var/log/app.log]
^vfs\..*\.contents$	La clave comienza con vfs. La clave termina con .contents Sin parámetros	Matches: vfs..contents vfs.mount.point.file.contents Does not match: vfs.contents vfs.file.contents[]
^vfs\.file\..*\[.*\]$	La clave comienza con vfs.file. Parámetros cualesquiera o vacíos	Matches vfs.file.get.custom[] vfs.file.size[/var/log/app.log, utf8] Does not match: vfs.file.get.custom
^vfs\.file\.contents$	La clave es vfs.file.contents Sin parámetros	Matches: vfs.file.contents Does not match: vfs.file.contents[/etc/passwd]
^vfs\.file\.contents\[\]$	La clave es vfs.file.contents[] Parámetros vacíos	Matches: vfs.file.contents[] Does not match: vfs.file.contents
^vfs\.file\.contents\[.*\]$	La clave es vfs.file.contents Parámetros cualesquiera o vacíos	Matches: vfs.file.contents[/path/to/file] Does not match: vfs.file.contents

Pattern	Matches	Examples

`^vfs\.file\.contents\[/etc/passwd,es*\]$`	`vfs.file.contents`
	El primer parámetro es `/etc/passwd`
	El segundo parámetro es cualquiera o vacío
		Matches:
		`vfs.file.contents[/etc/passwd,]`
		`vfs.file.contents[/etc/passwd,utf8]`
		
		Does not match:
		`vfs.file.contents[]`
		`vfs.file.contents[/etc/passwd]`

`^vfs\.file\.contents\[.*passwd.*,es\]$`	`vfs.file.contents`
	Parámetros cualesquiera, al menos uno contiene `passwd`
		Matches:
		`vfs.file.contents[/etc/passwd]`
		`vfs.file.contents[/etc/passwd,]`
		`vfs.file.contents[/etc/passwd,utf8]`

`^vfs\.file\.contents\[.*passwd.*,es\]$`	`vfs.file.contents`
	El primer parámetro contiene `passwd`
	El segundo parámetro es cualquiera o vacío
		Matches:
		`vfs.file.contents[/etc/passwd,]`
		`vfs.file.contents[/etc/passwd,utf8]`
		
		Does not match:
		`vfs.file.contents[/etc/passwd]`
		`vfs.file.contents[/tmp/test]`

`^vfs\.file\.contents\[/etc/passwd,(utf8|windows-1252)\]$`	`vfs.file.contents`
	El primer parámetro es `/etc/passwd`
	El segundo parámetro es `utf8` o `windows-1252`
		Matches:
		`vfs.file.contents[/etc/passwd,utf8]`
		`vfs.file.contents[/etc/passwd,windows-1252]`
		
		Does not match:
		`vfs.file.contents[/etc/passwd,]`
		`vfs.file.contents[/etc/passwd,utf16]`

Detección de problemas con triggers

La detección de problemas en Zabbix se basa en los **triggers**.

Un trigger describe la condición del problema, lo que permite a Zabbix reaccionar si el problema ocurre.

El flujo completo de detección de problemas es:

1. Asegúrese de tener un item que recopile datos.
2. Defina el trigger para este item. Sin un trigger, Zabbix solo recopila datos, pero nunca reacciona ante ellos.
3. Observe los problemas detectados:
   - En *Monitoring* > *Problems*
   - En el widget *Problems*
4. Reciba notificaciones, ejecute comandos remotos (si están definidos).

Tenga en cuenta que las templates oficiales de Zabbix contienen triggers predefinidos. Consulte la guía para aplicar template para monitor Linux.

¿Qué es un "trigger"?

Un trigger describe la *condición de problema*, mediante la expresión de trigger.

La condición de problema, cuando ocurre en la vida real, es cuando el trigger "se dispara" (se crea un problema en Zabbix).

La condición de problema puede basarse en el último valor, el valor promedio, una cadena detectada y muchas otras funciones. No se pueden crear triggers para items que devuelvan valores con el tipo de dato binario o JSON data type.

Una expresión de trigger simple calcula la función (por ejemplo, `max()` para el valor máximo) para los datos especificados entre paréntesis (normalmente el item de datos y la ventana de tiempo), y luego compara este cálculo con un umbral.

Por ejemplo:

```
max(/host/vfs.fs.size[/,free],5m)<10G
```

Este trigger se dispara si el espacio libre en disco, medido durante 5 minutos, ha estado constantemente por debajo de 10 GB.

Un trigger puede tener los siguientes estados:

Estado del trigger	Descripción	En Zabbix
OK	La expresión del trigger se ha evaluado como "0" (o FALSO)	No se crea ningún problema El problema se resuelve (si existía)
Problem	La expresión del trigger se ha evaluado como "1" (o VERDADERO)	Se crea un problema

A veces la evaluación de la expresión del trigger no es posible por alguna razón. Consulte Estado de expresión desconocido.

Otros recursos:

- Ejemplos de triggers
- Configurar un trigger
- Buenas prácticas

Frecuencia de cálculo

Un trigger se recalcula cada vez que el servidor Zabbix recibe un **nuevo valor** que forma parte de la expresión. Cuando se recibe un nuevo valor, cada función que está incluida en la expresión se recalcula (no solo la que recibió el nuevo valor).

Además, un trigger se recalcula cada 30 segundos (por el history syncer) si la expresión contiene:

- Funciones de Fecha y hora
- La función nodata()

Los triggers que contienen solo funciones de tendencia se evalúan una vez por el periodo de tiempo más pequeño en la expresión. Mientras que muchas funciones de trigger utilizan datos de histórico de valores de item, las funciones de tendencia utilizan datos de tendencia.

Ejemplos de trigger

Esta página es una colección de ejemplos de trigger.

Los ejemplos están ordenados por caso de uso:

- host no disponible
- proxy no disponible
- clúster SMTP no disponible
- host inalcanzable
- reinicio inesperado
- cambios en archivos importantes
- cambios en la consulta DNS
- software no coincidente en diferentes host
- relojes desincronizados
- agent desactualizado
- interfaz inestable
- alto tráfico entrante
- poco espacio en disco
- poco espacio en disco (umbral dinámico)
- alta carga de CPU
- comparación de carga de CPU
- comparación de carga de CPU a largo plazo
- cadena que contiene caracteres especiales

Detectar host no disponible

```
max(/host/zabbix[host,agent,available],5m)=0
```

Este trigger se dispara si el agent de Zabbix en el host ha estado no disponible durante 5 minutos.
 Función: `max`
 Item de datos: `zabbix[host,agent,available]`

Alternativa:

```
nodata(/host/agent.ping,5m)=1
```

Este trigger se dispara si no se han recibido datos del agent de Zabbix durante 5 minutos.
 Función: `nodata`
 Item de datos: `agent.ping`

Detectar proxy no disponible

```
fuzzytime(/host/zabbix[proxy,{$PROXY_NAME},lastaccess],1m)=0
```

Este trigger se activa si los datos del proxy de Zabbix se retrasan respecto al tiempo del server de Zabbix en 1 minuto.
 Función: fuzzytime
 Item de datos: zabbix[proxy,{$PROXY_NAME},lastaccess]

Detectar clúster SMTP no disponible

```
last(/smtp1.example.com/net.tcp.service[smtp])=0 and last(/smtp2.example.com/net.tcp.service[smtp])=0
```

Este trigger se activa si ambos servidores SMTP están caídos.
 Función: last
 Item de datos: net.tcp.service

Detectar host inaccesible

```
count(/host/icmpping,30m,,"0")>5
```

Este trigger se activa si el host no responde al ping más de 5 veces en los últimos 30 minutos.
 Función: count
 Item de datos: icmpping

Detectar reinicio inesperado

```
change(/host/system.uptime)<0
```

Este trigger se activa si se detecta un cambio negativo en el valor del tiempo de actividad del sistema (lo que indica un reinicio).
 Función: change
 Item de datos: system.uptime

Detectar cambios en archivos importantes

```
last(/host/vfs.file.cksum[/etc/passwd],#1)<>last(/host/vfs.file.cksum[/etc/passwd],#2)
```

Este trigger se activa si /etc/passwd ha sido modificado. La expresión es verdadera cuando la suma de verificación anterior de /etc/passwd difiere de la más reciente. Expresiones similares pueden ser útiles para monitorizar cambios en archivos importantes, como /etc/passwd, /etc/inetd.conf, /kernel, etc.
 Función: last
 Item de datos: vfs.file.cksum

Detectar cambios en la consulta DNS

```
last(/Zabbix server/net.dns.record[192.0.2.1,{$WEBSITE_NAME},{$DNS_RESOURCE_RECORD_TYPE},2,1])<>"{$WEBSITE
```

Observe las comillas alrededor del segundo operando.

Este trigger se activa si el resultado de la consulta no es igual a lo que normalmente devuelve:

```
example.com             MX       0 mail.example.com
```

Función: last
 Data item: net.dns.record[192.0.2.1,{$WEBSITE_NAME},{$DNS_RESOURCE_RECORD_TYPE},2,1], con macros definidas como:

```
{$WEBSITE_NAME} = example.com
{$DNS_RESOURCE_RECORD_TYPE} = MX
```

Detectar software que no coincide en diferentes hosts

```
last(/host/vfs.file.contents[/etc/os-release])<>last(/host2/vfs.file.contents[/etc/os-release])
```

Este trigger se activa si la versión de Ubuntu es diferente en distintos hosts. Observe cómo los operandos aquí son funciones que devuelven cadenas de texto.
 Función: last
 Item de datos: vfs.file.contents

Detectar relojes desincronizados

```
fuzzytime(/host/system.localtime,10s)=0
```

El trigger se dispara si la hora local del cliente y la hora del server de Zabbix difieren en más de 10 segundos.
 Función: fuzzytime
 Elemento de datos: system.localtime

Tenga en cuenta que system.localtime debe configurarse como una comprobación pasiva para Zabbix agent; en Zabbix agent 2 puede configurarse como una comprobación activa.

Detectar agent desactualizado

```
find(/host/agent.version,,"like","beta")=1
```

Este trigger se activa si el agent de Zabbix tiene una versión beta. Es necesario actualizar el agent de Zabbix.
 Función: find
 Item de datos: agent.version

Detectar interfaz inestable

```
changecount(/host/vfs.file.contents["/sys/class/net/eth0/operstate"],1h)>5
```

Este trigger se activa si el estado operativo (up/down/unknown) de eth0 ha cambiado más de 5 veces en una hora.
 Función: changecount
 Item de datos: vfs.file.contents

Detectar tráfico de entrada alto

```
min(/host/net.if.in[eth0,bytes],5m)>100K
```

Este trigger se activa si el número de bytes recibidos en eth0 durante los últimos cinco minutos fue siempre superior a 100 kilobytes. Probablemente alguien está descargando un archivo grande.
 Función: min
 Item de datos: net.if.in[eth0,bytes]

Detectar poco espacio en disco

```
max(/host/vfs.fs.size[/,free],5m)<10G
```

El trigger se dispara si el espacio libre en disco es consistentemente (5 minutos) inferior a 10 GB.

También puede definir una expresión de recuperación:

```
min(/host/vfs.fs.size[/,free],10m)>40G
```

El problema se **resuelve** sólo cuando el espacio libre en disco es consistentemente (10 minutos) superior a 40 GB.
 Función: min
 Item de datos: vfs.fs.size

Detectar poco espacio en disco (umbral dinámico)

```
last(/template/hrStorageFree[{#SNMPVALUE}])<last(/template/hrStorageSize[{#SNMPVALUE}])*0.1
```

El trigger se activa si el espacio libre (en unidades de asignación) cae por debajo del 10 por ciento. Observe que se utiliza el valor de otro item para obtener un umbral de trigger adaptativo, aplicable al almacenamiento descubierto de varios tamaños. Función: last

Detectar alta carga de CPU

```
last(/host/system.cpu.load[all,avg1])>5
```

El trigger se activa cuando la carga promedio del procesador ha estado por encima de 5 durante un minuto.

Variaciones:

```
min(/host/system.cpu.load[all,avg1],5m)>2 and time()<060000
```

```
min(/host/system.cpu.load[all,avg1],5m)>2 and not (dayofweek()=7 and time()>230000) and not (dayofweek()=1
```

Estos triggers analizan 5 minutos de datos y se activan solo si la carga de CPU nunca está por debajo de 2. Además, estos triggers se activan:

- solo en horario nocturno (00:00-06:00)
- en cualquier momento excepto durante 2 horas en el cambio de semana (domingo, 23:00 - lunes, 01:00)

```
(last(/host/system.cpu.load[all,avg1])>5) + (last(/host2/system.cpu.load[all,avg1])>5) + (last(/host3/syst
```

Este trigger se activa si la carga del procesador es demasiado alta en al menos dos de los tres hosts.
 Funciones: last, min, dayofweek, time
 Item de datos: system.cpu.load

Comparar cargas de CPU

```
avg(/Zabbix server/system.cpu.load,1h)/avg(/Zabbix server/system.cpu.load,1h:now-1d)>2
```

El trigger se activa si la carga promedio de hoy supera la carga promedio de la misma hora de ayer (usando el desplazamiento de tiempo como now-1d) más de dos veces.
 Función: avg
 Item de datos: system.cpu.load

Comparar cargas de CPU a largo plazo

```
trendavg(/host/system.cpu.load,1M:now/M)>1.1*trendavg(/host/system.cpu.load,1M:now/M-1M)
```

Este trigger se dispara si la carga de CPU en el host aumentó más del 10% el mes pasado
 Función: trendavg
 Item de datos: system.cpu.load

También puede utilizar el campo Nombre del evento en la configuración del trigger para construir un mensaje de alerta significativo, por ejemplo, para recibir algo como

```
"La carga del servidor Exchange aumentó un 24% en julio (0,69) en comparación con junio (0,56)"
```

el nombre del evento debe definirse como:

```
Load of {HOST.HOST} server increased by {{?100*trendavg(//system.cpu.load,1M:now/M)/trendavg(//system.cpu.
```

También es útil permitir el cierre manual en la configuración del trigger para este tipo de problema.

Detectar cadena que contiene caracteres especiales

```
last(/host/vfs.file.contents[/tmp/hello])={$HELLO_MACRO}
```

El trigger se activa si el contenido de `/tmp/hello` es igual a la cadena definida en `{$HELLO_MACRO}`:

```
{$HELLO_MACRO} = \" //hello ?\"
```

Alternativamente, puede comparar con la cadena directamente:

```
last(/Zabbix server/vfs.file.contents[/tmp/hello])="\\\" //hello ?\\\""
```

Observe cómo los caracteres especiales (\\ y ") se escapan cuando la cadena se compara directamente.

Función: `last`
 Item de datos: `vfs.file.contents`

Configuración de un trigger

Esta página describe cómo configurar un trigger en el frontend de Zabbix.

Tenga en cuenta que las templates oficiales de Zabbix contienen triggers predefinidos. Consulte la guía para aplicar la template a monitor Linux. También es posible actualizar en masa los triggers configurados.

Otros recursos:

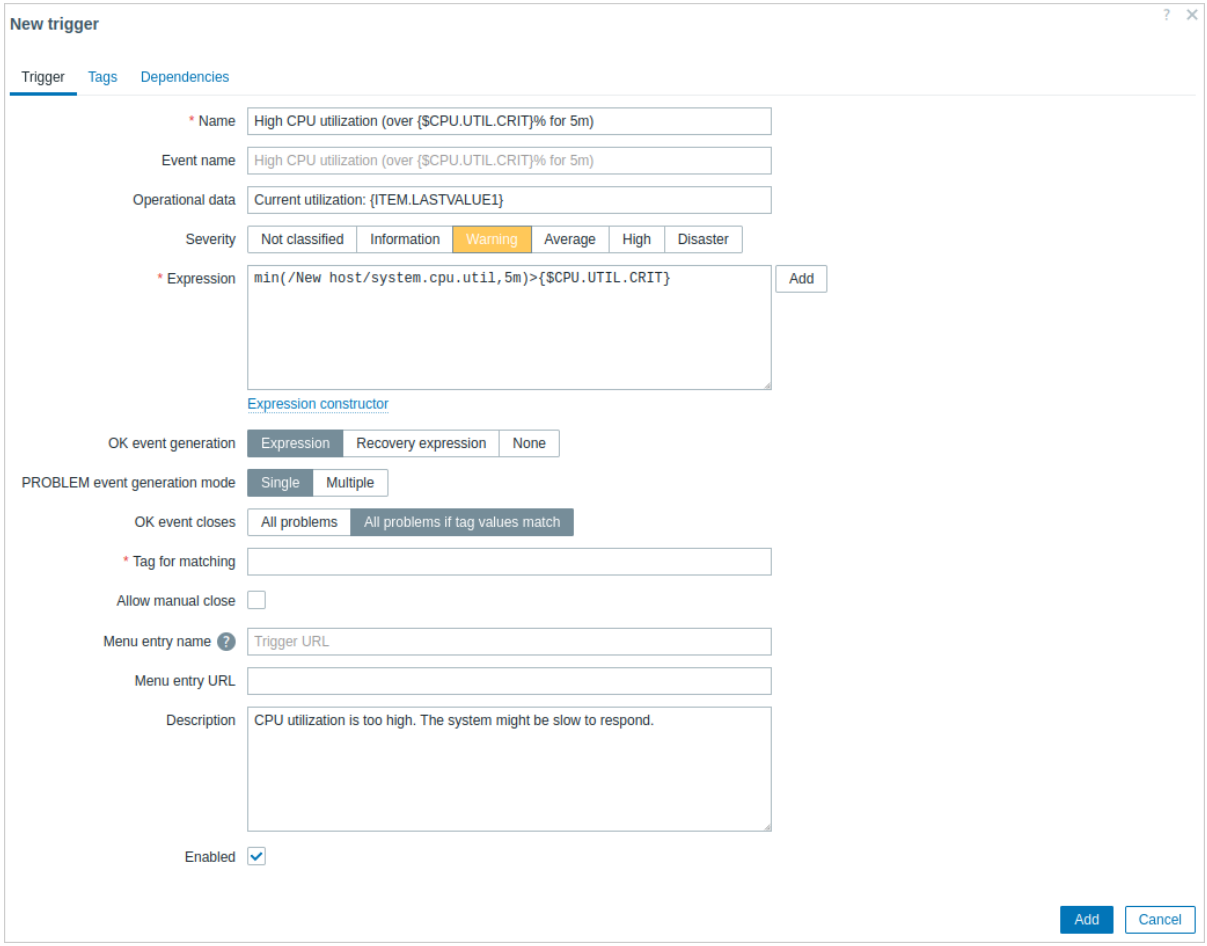
- Detección de problemas con triggers (introducción general)
- Ejemplos de triggers (según el caso de uso)
- Expresión y funciones de trigger (detalles de sintaxis)
- Buenas prácticas

Configuración

Para configurar un trigger, haga lo siguiente:

1. Vaya a *Data collection* > *Hosts*.
2. Haga clic en *Triggers* en la fila del host.
3. Haga clic en *Create trigger* a la derecha (o en el nombre del trigger para editar un trigger existente).
4. Introduzca los parámetros del trigger en el formulario.

La pestaña **Trigger** contiene todos los atributos esenciales del trigger.



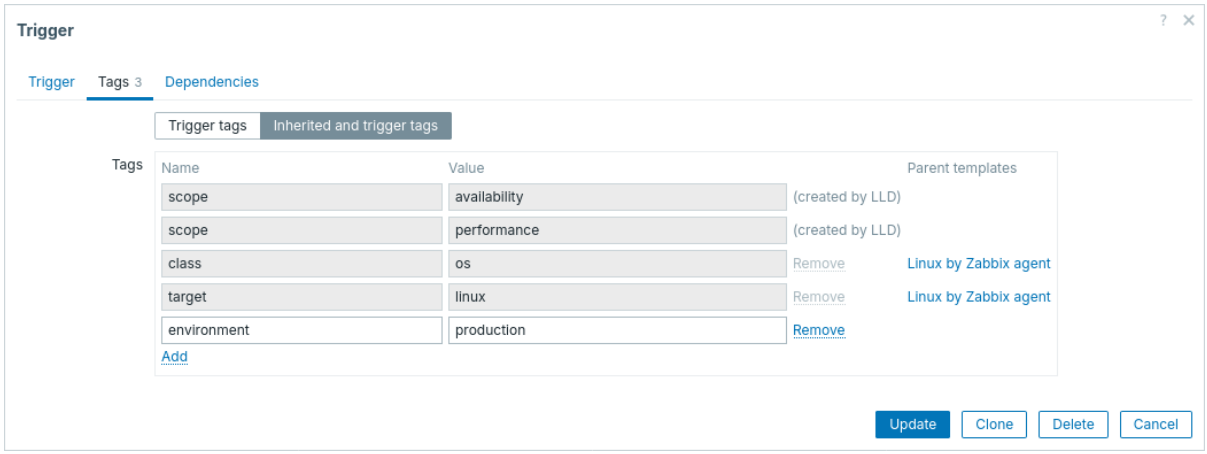
Todos los campos de entrada obligatorios están marcados con un asterisco rojo.

Parameter	Description
Name	Nombre del trigger. Los macros admitidos son: {HOST.HOST}, {HOST.NAME}, {HOST.PORT}, {HOST.CONN}, {HOST.DNS}, {HOST.IP}, {ITEM.VALUE}, {ITEM.VALUE.AGE}, {ITEM.VALUE.DATE}, {ITEM.VALUE.TIME}, {ITEM.VALUE.TIMESTAMP}, {ITEM.LASTVALUE}, {ITEM.LASTVALUE.AGE}, {ITEM.LASTVALUE.DATE}, {ITEM.LASTVALUE.TIME}, {ITEM.LASTVALUE.TIMESTAMP}, {ITEM.LOG.*} y los user macros {$MACRO}. Los macros **$1, $2...$9** se pueden usar para referirse a la primera, segunda...novena constante de la expresión. *Nota*: los macros $1-$9 se resolverán correctamente si hacen referencia a constantes en expresiones relativamente simples y directas. Por ejemplo, el nombre "Processor load above $1 on {HOST.NAME}" cambiará automáticamente a "Processor load above 5 on New host" si la expresión es last(/New host/system.cpu.load[percpu,avg1])>5.
Event name	Cuando se define, este nombre se usará para crear el nombre del evento de problema. De forma predeterminada, el nombre del evento es el mismo que el nombre del trigger. El nombre del evento puede usarse para generar alertas significativas que contengan datos del problema (consulte el ejemplo). Se admite el mismo conjunto de macros que en el nombre del trigger, además de los macros de expresión {TIME}, {TIMESTAMP} y {?EXPRESSION}.

Parameter	Description
Operational data	Introduzca una cadena con macros para mostrar datos dinámicos y en tiempo real en *Monitoring* > *Problems*. Como alternativa, deje este campo vacío para mostrar los últimos valores de todos los items de la expresión del trigger. Se admite el mismo conjunto de macros que en el nombre del trigger, con la capacidad de resolverse dinámicamente. Por ejemplo: {ITEM.VALUE<1-9>} se resuelve en los valores del item en el momento en que cambia el estado del trigger (problema creado, resuelto, cerrado manualmente o cerrado por correlación). {ITEM.LASTVALUE<1-9>} se resuelve en los últimos valores del item. Tenga en cuenta que cerrar un problema manualmente no produce un nuevo valor, por lo que ambos macros seguirán mostrando el valor del momento del problema. Tenga en cuenta también que ambos macros se resuelven como *UNKNOWN* si el último valor es más antiguo que *Max history display period* (consulte *Administration > General*).
Severity	Establezca la severity requerida del trigger haciendo clic en los botones.
Expression	Expresión lógica usada para definir las condiciones de un problema. Se admiten sufijos de tiempo y sufijos de tamaño de memoria. Se crea un problema después de que se cumplan todas las condiciones incluidas en la expresión, es decir, cuando la expresión se evalúa como TRUE. El problema se resolverá en cuanto la expresión se evalúe como FALSE, a menos que se especifiquen condiciones adicionales de recuperación en *Recovery expression*.
OK event generation	Opciones de generación de eventos OK: **Expression** - los eventos OK se generan en función de la misma expresión que los eventos de problema; **Recovery expression** - los eventos OK se generan si la expresión del problema se evalúa como FALSE y la expresión de recuperación se evalúa como TRUE; **None** - en este caso, el trigger nunca volverá por sí solo a un estado OK.
Recovery expression	Expresión lógica opcional que define condiciones adicionales que deben cumplirse antes de que se resuelva el problema, después de que la expresión original del problema ya se haya evaluado como FALSE. La expresión de recuperación es útil para la histéresis del trigger. **No** es posible resolver un problema solo mediante la expresión de recuperación si la expresión del problema sigue siendo TRUE. Este campo solo está disponible si se selecciona 'Recovery expression' para *OK event generation*.
PROBLEM event generation mode	Modo de generación de eventos de problema: **Single** - se genera un único evento cuando un trigger entra por primera vez en el estado 'Problem'; **Multiple** - se genera un evento en *cada* evaluación 'Problem' del trigger.
OK event closes	Seleccione si el evento OK cierra: **All problems** - todos los problemas de este trigger; **All problems if tag values match** - solo aquellos problemas del trigger con valores coincidentes de las etiquetas del evento.
Tag for matching	Introduzca el nombre de la etiqueta del evento que se usará para la correlación de eventos. Este campo se muestra si se selecciona 'All problems if tag values match' para la propiedad *OK event closes* y es obligatorio en este caso.
Allow manual close	Marque esta opción para permitir el cierre manual de los eventos de problema generados por este trigger. El cierre manual es posible al reconocer los eventos de problema.
Menu entry name	Si no está vacío, el nombre introducido aquí (hasta 64 caracteres) se usa en varias ubicaciones del frontend como etiqueta para la URL del trigger especificada en el parámetro *Menu entry URL*. Si está vacío, se usa el nombre predeterminado *Trigger URL*. Se admite el mismo conjunto de macros que en el nombre del trigger, además de {EVENT.ID}, {HOST.ID} y {TRIGGER.ID}.
Menu entry URL	Si no está vacío, la URL introducida aquí (hasta 2048 caracteres) está disponible como enlace en el event menu en varias ubicaciones del frontend, por ejemplo, al hacer clic en el nombre del problema en el widget de panel *Monitoring* > *Problems* o *Problems*. Se admite el mismo conjunto de macros que en el nombre del trigger, además de {EVENT.ID}, {HOST.ID} y {TRIGGER.ID}. Nota: los user macros con valores secretos no se resolverán en la URL.
Description	Campo de texto usado para proporcionar más información sobre este trigger. Puede contener instrucciones para corregir un problema específico, datos de contacto del personal responsable, etc. Se admite el mismo conjunto de macros que en el nombre del trigger.

Parameter	Description
Enabled	Si desmarca esta casilla, el trigger se deshabilitará si es necesario. Los problemas de un trigger deshabilitado ya no se muestran en el frontend, pero no se eliminan.

La pestaña **Tags** le permite definir etiquetas a nivel de trigger. Todos los problemas de este trigger se etiquetarán con los valores introducidos aquí.



Además, la opción *Inherited and trigger tags* permite ver las etiquetas definidas a nivel de template si el trigger procede de ese template. Si hay varios templates con la misma etiqueta, estas etiquetas se muestran una sola vez y los nombres de los templates se separan con comas. Un trigger no "hereda" ni muestra etiquetas a nivel de host.

Parameter	Description
Name/Value	Defina etiquetas personalizadas para marcar eventos de trigger. Las etiquetas son un par de nombre y valor de etiqueta. Puede usar solo el nombre o combinarlo con un valor. Un trigger puede tener varias etiquetas con el mismo nombre, pero con valores diferentes. Se admiten user macros, user macros con contexto, macros de descubrimiento de bajo nivel y funciones de macro functions con `{{ITEM.VALUE}}`, `{{ITEM.LASTVALUE}}` en las etiquetas de evento. Las macros de descubrimiento de bajo nivel se pueden usar dentro del contexto de macro. La macro {TRIGGER.ID} se admite en los valores de las etiquetas del trigger. Puede ser útil para identificar triggers creados a partir de prototipos de trigger y, por ejemplo, suprimir problemas de estos triggers durante el mantenimiento. Si la longitud total del valor expandido supera 255, se recortará a 255 caracteres. Consulte todos los macros admitidos para las etiquetas de evento. Las etiquetas de evento se pueden usar para la correlación de eventos, en condiciones de acciones y también se verán en *Monitoring* > *Problems* o en el widget *Problems*.

La pestaña **Dependencies** contiene todas las dependencias del trigger.

Haga clic en *Add* para añadir una nueva dependencia.

> **Note:**
> También puede configurar un trigger abriendo uno existente, haciendo clic en el botón *Clone* y guardándolo después con un nombre diferente.

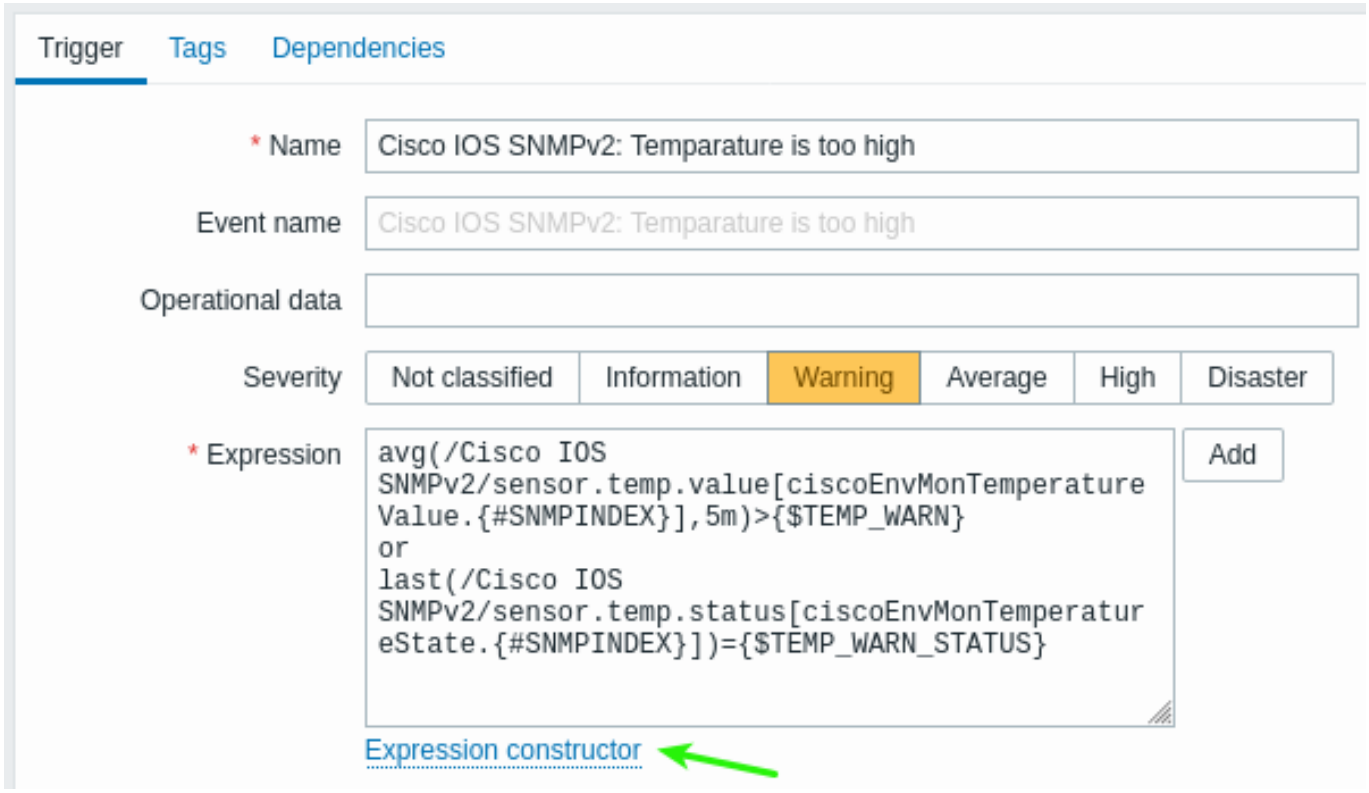
Prueba de la expresión del trigger

Es posible probar cuál sería el resultado de la expresión en función del valor recibido.

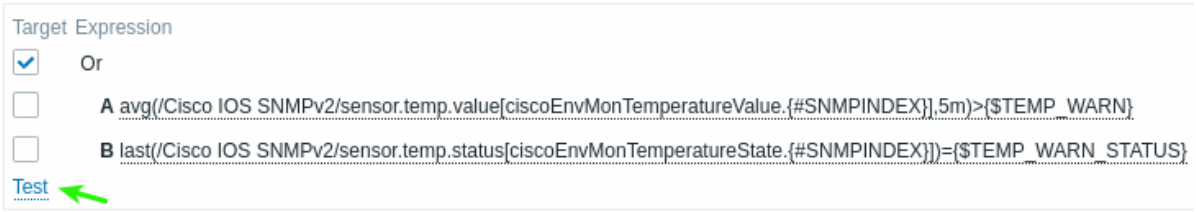
La siguiente expresión de una template oficial se toma como ejemplo:

```
avg(/Cisco IOS SNMPv2/sensor.temp.value[ciscoEnvMonTemperatureValue.{#SNMPINDEX}],5m)>{$TEMP_WARN}
or
last(/Cisco IOS SNMPv2/sensor.temp.status[ciscoEnvMonTemperatureState.{#SNMPINDEX}])={$TEMP_WARN_STATUS}
```

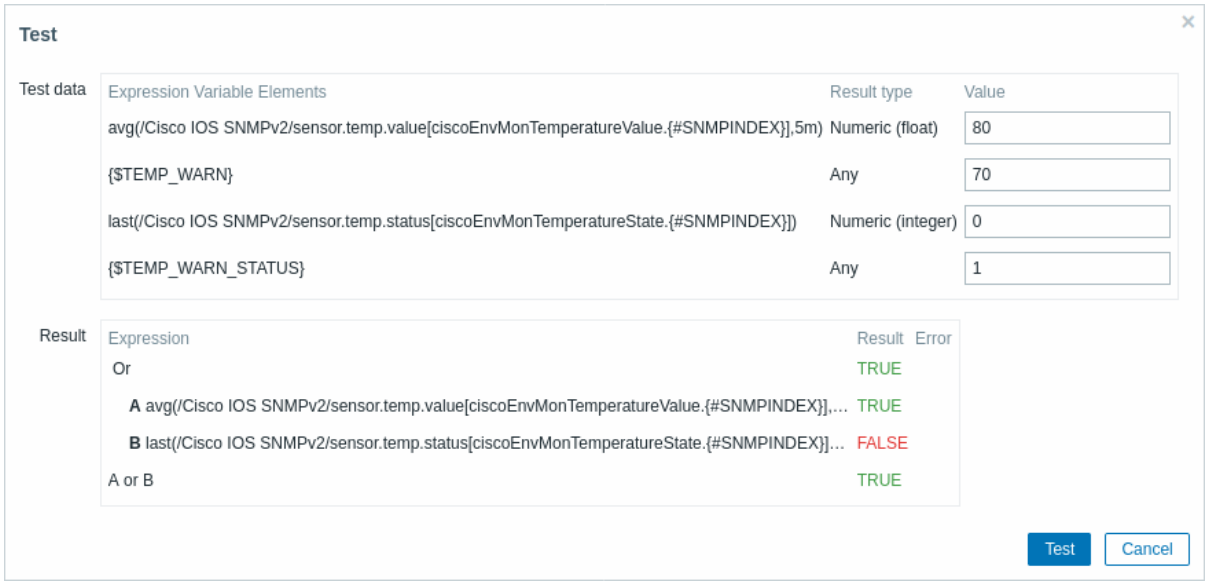
Para probar la expresión, haga clic en *Constructor de expresiones* debajo del campo de expresión.



En el Constructor de expresiones, se enumeran todas las expresiones individuales. Para abrir la ventana de prueba, haga clic en *Probar* debajo de la lista de expresiones.



En la ventana de prueba puede introducir valores de ejemplo ('80', '70', '0', '1' en este ejemplo) y luego ver el resultado de la expresión, haciendo clic en el botón *Probar*.



Se puede ver el resultado de las expresiones individuales, así como de la expresión completa.

"TRUE" significa que la expresión especificada es correcta. En este caso A, "80" es mayor que el valor especificado `{$TEMP_WARN}`, "70" en este ejemplo. Como era de esperar, aparece un resultado "TRUE".

"FALSE" significa que la expresión especificada es incorrecta. En este caso B, `{$TEMP_WARN_STATUS}` "1" debe ser igual al valor especificado, "0" en este ejemplo. Como era de esperar, aparece un resultado "FALSE".

El tipo de expresión elegido es ”OR”. Si al menos una de las condiciones especificadas (A o B en este caso) es TRUE, el resultado global también será TRUE. Esto significa que el valor actual supera el valor de advertencia y se ha producido un problema.

Expresión de trigger y funciones

Descripción general

Esta página describe la sintaxis de las expresiones de trigger y los detalles de su evaluación.

La sintaxis de una expresión simple es:

`función(/host/item,periodo_de_tiempo)<operador><constante>`

En esta expresión, el primer operando (a la izquierda del operador) es una función con sus parámetros entre paréntesis (normalmente el item de datos y el periodo de tiempo).

La función se utiliza para analizar los datos recibidos durante el periodo de tiempo establecido, dando como resultado un valor calculado.

Este valor se compara entonces con el segundo operando, utilizando el operador. En este ejemplo, el segundo operando es una constante, pero también puede ser otra función.

Por ejemplo:

```
min(/Zabbix server/net.if.in[eth0,bytes],5m)>100K
```

Este trigger se activa si el número de bytes recibidos en `eth0` durante los últimos cinco minutos fue siempre superior a 100 kilobytes. En este caso, la expresión es verdadera y se crea un problema.

Las expresiones de trigger son extremadamente flexibles. En expresiones más complejas, se pueden combinar varias funciones, operadores y constantes.

Consulte también:

- Ejemplos de triggers
- Detección de problemas con triggers (introducción general)

Funciones

Las funciones permiten analizar los valores recopilados, por ejemplo, calcular el promedio o buscar una cadena específica.

Haga clic en el grupo de funciones correspondiente para ver más detalles.

Grupo de funciones	Funciones
Funciones de agregación	avg, bucket_percentile, count, histogram_quantile, item_count, kurtosis, mad, max, min, skewness, stddevpop, stddevsamp, sum, sumofsquares, varpop, varsamp
Funciones Foreach	avg_foreach, bucket_rate_foreach, count_foreach, exists_foreach, last_foreach, max_foreach, min_foreach, sum_foreach
Funciones bit a bit	bitand, bitlshift, bitnot, bitor, bitrshift, bitxor
Funciones de fecha y hora	date, dayofmonth, dayofweek, now, time
Funciones de historial	change, changecount, count, countunique, find, first, firstclock, fuzzytime, jsonpath, last, lastclock, logeventid, logseverity, logsource, logtimestamp, monodec, monoinc, nodata, percentile, rate, xmlxpath

Grupo de funciones	Funciones
Funciones de tendencias	baselinedev, baselinewma, trendavg, trendcount, trendmax, trendmin, trendstl, trendsum
Funciones matemáticas	abs, acos, asin, atan, atan2, avg, cbrt, ceil, cos, cosh, cot, degrees, e, exp, expm1, floor, log, log10, max, min, mod, pi, power, radians, rand, round, signum, sin, sinh, sqrt, sum, tan, truncate
Funciones de operadores	between, in
Funciones predictivas	forecast, timeleft
Funciones de cadenas	ascii, bitlength, bytelength, char, concat, contains, insert, jsonpath, left, length, ltrim, mid, repeat, replace, right, rtrim, substring, trim, xmlxpath

Salvo que se indique lo contrario, estas funciones son compatibles con:

- Expresiones de trigger
- Items calculados
- Macros de expresión

Las funciones Foreach solo son compatibles con los cálculos de agregación.

Normalmente, las funciones devuelven valores numéricos para su comparación. Cuando devuelven cadenas, la comparación es posible con los operadores **=** y **<>** (consulte el ejemplo Detectar software no coincidente en distintos hosts).

Parámetros de las funciones

Los parámetros de las funciones permiten especificar:

- la clave del item (como `/host/key`) para las funciones que hacen referencia al historial de un item del host
- el período de tiempo (y otros parámetros específicos de la función)
- otras expresiones

Clave del item

El item referenciado debe estar en estado soportado (excepto para la función **nodata()**, que también se calcula para items no soportados).

La omisión del nombre del host en el primer parámetro (es decir, como en `function(//key,parameter,...)`) solo se admite en ciertos contextos:

- En la fórmula de items calculados
- En macros de expresión, que pueden usarse en:
  - El campo *Nombre del evento*
  - Nombre del gráfico
  - La etiqueta de los elementos de mapa "Host" y "Trigger"

En estos contextos, también puede usar la macro `{HOST.HOST}`. `{HOST.HOST<1-9>}` podría usarse en el caso del campo *Nombre del evento* y el elemento de mapa "Trigger" para referirse a un item específico en la expresión del trigger. Cuando el nombre del host se omite o se reemplaza por `{HOST.HOST}` en estos contextos, la referencia apunta al primer item en la expresión del trigger o al primer item en el gráfico. Fuera de estos contextos admitidos, omitir el nombre del host en las expresiones de triggers resultará en un error. Consulte el ejemplo Comparar cargas de CPU a largo plazo para ver una ilustración del uso de la doble barra en macros de nombre de evento.

Período de tiempo

Los parámetros específicos de la función se colocan después de la clave del item y se separan de ella mediante una coma.

La mayoría de las funciones numéricas aceptan un **período de tiempo** como parámetro.
Esto permite especificar el intervalo que nos interesa.
Puede especificarse como un período de tiempo (30s, 10m, 1h) o como un rango de valores (#5 - para los cinco valores más recientes).

Puede utilizar segundos o sufijos de tiempo para indicar el período de tiempo.
Cuando el parámetro está precedido por un signo de almohadilla, tiene un significado diferente:

Expression	Description
sum(/host/key,**10m)**	Suma de los valores de los últimos 10 minutos.
sum(/host/key,**#10)**	Suma de los diez últimos valores.

Los parámetros con un signo de almohadilla tienen un significado diferente con la función **last**: indican el valor número N anterior. Por lo tanto, dados los valores 30, 70, 20, 60, 50 (del más reciente al menos reciente):

- `last(/host/key,#2)` devolvería '70'
- `last(/host/key,#5)` devolvería '50'

El período de tiempo se mide hasta «ahora», donde «ahora» es la hora del último recálculo del trigger (consulte Frecuencia de cálculo); «ahora» no es la hora actual del server.

El período de tiempo especifica una de las siguientes opciones:

- Todos los valores entre «ahora - período de tiempo» y «ahora» (o, con un desplazamiento temporal aplicado, entre «ahora - desplazamiento temporal - período de tiempo» y «ahora - desplazamiento temporal»).
- El número especificado de valores del pasado, hasta «ahora».

Si no existen datos para el período de evaluación especificado, el trigger o el item calculado que utiliza esta función pasa a no soportado.

Tenga en cuenta lo siguiente:

- Si en el trigger se utiliza una sola función (que hace referencia al historial de datos), «ahora» siempre es el último valor recibido. Por ejemplo, si el último valor se recibió hace una hora, el período de tiempo se considerará hasta el último valor recibido hace una hora.
- Un nuevo trigger se calcula en cuanto se recibe el primer valor (funciones de historial); se calculará en un plazo de 30 segundos para las funciones de fecha y hora y nodata(). Por lo tanto, el trigger se calculará aunque todavía no haya transcurrido el período de tiempo establecido (por ejemplo, una hora) desde su creación. El trigger también se calculará después del primer valor, aunque el intervalo de tiempo se haya establecido, por ejemplo, en los diez valores más recientes.

Desplazamiento temporal

Se admite un desplazamiento temporal opcional con el tiempo o el recuento de valores como parámetro de la función. Este parámetro permite hacer referencia a datos de un periodo de tiempo pasado.

El desplazamiento temporal comienza con `now`, que especifica la hora actual, y va seguido de `+N<time unit>` o `-N<time unit>`, para sumar o restar N unidades de tiempo.

Por ejemplo, `avg(/host/key,1h:now-1d)` devolverá el valor promedio de una hora correspondiente a hace un día.

Attention:
El desplazamiento temporal especificado en meses (M) y años (y) solo se admite para las funciones de tendencias. Otras funciones admiten segundos (s), minutos (m), horas (h), días (d) y semanas (w).

Desplazamiento temporal con periodos de tiempo absolutos

Los periodos de tiempo absolutos se admiten en el parámetro de desplazamiento temporal; por ejemplo, de medianoche a medianoche para un día, de lunes a domingo para una semana y del primer al último día del mes para un mes.

El desplazamiento temporal para periodos absolutos comienza con `now`, que especifica la hora actual, y va seguido de cualquier número de operaciones de tiempo: `/<time unit>`, que define el inicio y el final de la unidad de tiempo, por ejemplo, de medianoche a medianoche para un día; `+N<time unit>` o `-N<time unit>`, para sumar o restar N unidades de tiempo.

Tenga en cuenta que el valor del desplazamiento temporal puede ser mayor o igual que 0, mientras que el valor mínimo del periodo de tiempo es 1.

Parámetro	Descripción
1d:now/d	Ayer
1d:now/d+1d	Hoy
2d:now/d+1d	Últimos 2 días
1w:now/w	La semana pasada
1w:now/w+1w	Esta semana

Otras expresiones

Los parámetros de función pueden contener otras expresiones, como en la siguiente sintaxis:

```
min(min(/host/key,1h),min(/host2/key2,1h)*10)
```

Tenga en cuenta que no se pueden utilizar otras expresiones si la función hace referencia al historial del item. Por ejemplo, la siguiente sintaxis **no** está permitida:

~~`min(/host/key,#5*10)`~~

Mientras que otras expresiones de trigger como parámetros de función están limitadas a funciones que no sean de historial en los triggers, esta limitación no se aplica en los items calculados.

Operadores

Se admiten los siguientes operadores para triggers **(en orden de prioridad de ejecución descendente)**:

Prioridad	Operador	Definición	**Notas para valores desconocidos**	Forzar conversión de operando a float [1]
1	**-**	Menos unario	**-**Desconocido → Desconocido	Sí
2	**not**	Negación lógica	**not** Desconocido → Desconocido	Sí
3	*	Multiplicación	0 * Desconocido → Desconocido (sí, Desconocido, distinto de 0 - para no perder Desconocido en operaciones aritméticas) 1.2 * Desconocido → Desconocido	Sí
	/	División	Desconocido / 0 → error Desconocido / 1.2 → Desconocido 0.0 / Desconocido → Desconocido	Sí
4	+	Suma aritmética	1.2 + Desconocido → Desconocido	Sí
	-	Resta aritmética	1.2 - Desconocido → Desconocido	Sí
5	<	Menor que. El operador se define como: A<B ⇔ (A<B-0.000001)	1.2 < Desconocido → Desconocido	Sí
	<=	Menor o igual que. El operador se define como: A<=B ⇔ (A≤B+0.000001)	Desconocido <= Desconocido → Desconocido	Sí
	>	Mayor que. El operador se define como: A>B ⇔ (A>B+0.000001)		Sí

Prioridad	Operador	Definición	**Notas para valores desconocidos**	Forzar conversión de operando a float [1]
	>=	Mayor o igual que. El operador se define como: A>=B ⇔ (A≥B-0.000001)		Sí
6	**=**	Igual a. El operador se define como: A=B ⇔ (A≥B-0.000001) y (A≤B+0.000001)		No [1]
	<>	Distinto de. El operador se define como: A<>B ⇔ (A<B-0.000001) o (A>B+0.000001)		No [1]
7	**and**	AND lógico	0 **and** Desconocido → 0 1 **and** Desconocido → Desconocido Desconocido **and** Desconocido → Desconocido	Sí
8	**or**	OR lógico	1 **or** Desconocido → 1 0 **or** Desconocido → Desconocido Desconocido **or** Desconocido → Desconocido	Sí

[1] El operando de tipo string aún se convierte a numérico si:

- el otro operando es numérico
- se utiliza un operador distinto de **=** o **<>** sobre un operando

(Si la conversión falla, el operando numérico se convierte a string y ambos operandos se comparan como cadenas.)

Los operadores **not**, **and** y **or** distinguen entre mayúsculas y minúsculas y deben estar en minúsculas. También deben estar rodeados de espacios o paréntesis.

Todos los operadores, excepto el **-** unario y **not**, tienen asociatividad de izquierda a derecha. El **-** unario y **not** no son asociativos (lo que significa que se debe usar **-(-1)** y **not (not 1)** en lugar de **--1** y **not not 1**).

Resultado de la evaluación:

- Los operadores **<**, **<=**, **>**, **>=**, **=**, **<>** devolverán '1' en la expresión del trigger si la relación especificada es verdadera y '0' si es falsa. Si al menos un operando es Desconocido, el resultado es Desconocido;
- **and** para operandos conocidos devolverá '1' si ambos operandos son distintos de '0'; en caso contrario, devolverá '0'; para operandos desconocidos, **and** devolverá '0' solo si un operando es igual a '0'; en caso contrario, devolverá 'Desconocido';
- **or** para operandos conocidos devolverá '1' si alguno de sus operandos es distinto de '0'; en caso contrario, devolverá '0'; para operandos desconocidos, **or** devolverá '1' solo si un operando es distinto de '0'; en caso contrario, devolverá 'Desconocido';
- El resultado del operador de negación lógica **not** para un operando conocido es '0' si el valor de su operando es distinto de '0'; '1' si el valor de su operando es igual a '0'. Para un operando desconocido, **not** devuelve 'Desconocido'.

Estado de expresión desconocido

Es posible que aparezca un operando desconocido en una expresión de trigger cuando:

- se utiliza un item no compatible
- la evaluación de una función, para un item compatible, da como resultado un error

En este caso, la expresión del trigger generalmente se evalúa como `Unknown` (ya que no puede evaluarse)

Es posible recibir notificaciones sobre triggers desconocidos.

Excepciones

A pesar de un operando desconocido, las expresiones de trigger pueden evaluarse a un resultado conocido (Problem/OK) en algunos casos:

- La función `nodata()` se evalúa independientemente de si el item referenciado es compatible o no.
- Las expresiones con AND/OR pueden evaluarse a un resultado conocido en dos casos:
  - **Caso 1**: "`1 or some_function(unsupported_item1) or some_function(unsupported_item2) or ...`" se evalúa a un resultado conocido ('1' o "Problem"),
  - **Caso 2**: "`0 and some_function(unsupported_item1) and some_function(unsupported_item2) and ...`" se evalúa a un resultado conocido ('0' o "OK").
- Si la evaluación de la función para un item compatible da como resultado un error, el valor de la función pasa a ser `Unknown` y participa como operando desconocido en la evaluación posterior de la expresión.

Los operandos desconocidos pueden "desaparecer" solo en expresiones lógicas, como se describe arriba. En expresiones aritméticas, los operandos desconocidos siempre conducen a `Unknown` (excepto la división por 0).

Attention:
El estado de expresión desconocido no cambia el estado del trigger (Problem/OK). Por lo tanto, si el estado del trigger era "Problem" (véase el Caso 1), permanece en estado de problema incluso si la parte conocida se resuelve ('1' pasa a '0'), porque la expresión ahora se evalúa como `Unknown` y eso no cambia el estado del trigger.

Si una expresión de trigger con varios items no compatibles se evalúa como `Unknown`, el mensaje de error en el frontend hace referencia al último item no compatible evaluado.

Almacenamiento en caché de valores

Los valores necesarios para la evaluación de disparadores son almacenados en caché por el servidor Zabbix. Debido a esto, la evaluación de disparadores provoca una mayor carga en la base de datos durante algún tiempo después de reiniciar el servidor. La caché de valores no se borra cuando se eliminan los valores históricos de los ítems (ya sea manualmente o por el limpiador), por lo que el servidor utilizará los valores almacenados en caché hasta que sean más antiguos que los períodos de tiempo definidos en las funciones de los disparadores o hasta que se reinicie el servidor.

Note:
Si no hay datos recientes en la caché y no se ha definido un período de consulta en la función, Zabbix, por defecto, buscará en la base de datos valores históricos hasta una semana atrás.

Funciones agregadas

Salvo que se indique lo contrario, todas las funciones enumeradas aquí son compatibles con:

- Expresiones de trigger
- Fórmulas de item calculado
- Macros de expresión

Las funciones agregadas pueden trabajar con cualquiera de las siguientes opciones:

- historial de items, por ejemplo, `min(/host/key,1h)`
- funciones foreach como único parámetro, por ejemplo, `min(last_foreach(/*/key))` (solo en fórmulas de item calculado; para usar en triggers, haga referencia al item calculado por su clave)

Las funciones se enumeran sin información adicional. Haga clic en la función para ver los detalles completos.

Function	Description
avg	El valor promedio de un item dentro del período de evaluación definido.
bucket_percentile	Calcula el percentil a partir de los buckets de un histograma.
count	El recuento de valores en un array devuelto por una función foreach.
histogram_quantile	Calcula el cuantil φ a partir de los buckets de un histograma.
item_count	El recuento de items existentes en la configuración que coinciden con los criterios de filtro.
kurtosis	La "curtosis" de la distribución de probabilidad en los valores recopilados dentro del período de evaluación definido.
mad	La desviación absoluta mediana en los valores recopilados dentro del período de evaluación definido.
max	El valor más alto de un item dentro del período de evaluación definido.
min	El valor más bajo de un item dentro del período de evaluación definido.
skewness	La asimetría de la distribución de probabilidad en los valores recopilados dentro del período de evaluación definido.

Function	Description
stddevpop	La desviación estándar poblacional en los valores recopilados dentro del período de evaluación definido.
stddevsamp	La desviación estándar muestral en los valores recopilados dentro del período de evaluación definido.
sum	La suma de los valores recopilados dentro del período de evaluación definido.
sumofsquares	La suma de cuadrados en los valores recopilados dentro del período de evaluación definido.
varpop	La varianza poblacional de los valores recopilados dentro del período de evaluación definido.
varsamp	La varianza muestral de los valores recopilados dentro del período de evaluación definido.

Parámetros comunes

- `/host/key` es un primer parámetro común obligatorio para las funciones que hacen referencia al historial del item del host.
- `(sec|#num)<:time shift>` es un segundo parámetro común para las funciones que hacen referencia al historial del item del host, donde:
  - **sec**: período máximo de evaluación en segundos (se pueden utilizar sufijos de tiempo).
  - **#num**: rango máximo de evaluación en los valores recopilados más recientes (si va precedido de un signo de almohadilla).
  - **time shift** (opcional) permite mover el punto de evaluación hacia atrás en el tiempo. Consulte más detalles sobre cómo especificar el desplazamiento temporal.

Detalles de la función

Algunas notas generales sobre los parámetros de la función:

- Los parámetros de la función están separados por una coma
- Los parámetros opcionales de la función (o partes del parámetro) se indican mediante < >
- Los parámetros específicos de la función se describen con cada función
- Los parámetros `/host/key` y `(sec|#num)<:desplazamiento de tiempo>` nunca deben ir entre comillas

avg(/host/key,(sec|#num)<:time shift>)

El valor promedio de un item dentro del período de evaluación definido.
 Tipos de valor admitidos: *Float*, *Integer*.
 Funciones foreach admitidas: *avg_foreach*, *count_foreach*, *exists_foreach*, *last_foreach*, *max_foreach*, *min_foreach*, *sum_foreach*.

Parámetros: consulte los parámetros comunes.

El desplazamiento temporal es útil cuando existe la necesidad de comparar el valor promedio actual con el valor promedio de algún momento anterior.

Ejemplos:

```
avg(/host/key,1h) #calcular el valor promedio de la última hora hasta ahora
avg(/host/key,#5) #calcular el valor promedio de los cinco valores más recientes
avg(/host/key,1h:now-1d) #calcular el valor promedio durante una hora, desde hace 25 horas hasta hace 24 h
avg(last_foreach(/*/system.cpu.load[,avg1]?[group="MySQL Servers"])) #calcular la carga promedio del proce
avg(/host/proc.num,5m)>300 #trigger si el número promedio de procesos en los últimos 5 minutos ha sido sup
```

bucket_percentile(item filter,time period,percentage)

Calcula el percentil a partir de los buckets de un histograma.

Parámetros:

- **item filter** - consulte item filter;
- **time period** - consulte time period;
- **percentage** - porcentaje (0-100).

Comentarios:

- Compatible solo en items calculados;
- Esta función es un alias de `histogram_quantile(percentage/100, bucket_rate_foreach(item filter, time period, 1))`.

Ejemplos:

```
bucket_percentile(/*/http_request_duration_seconds_bucket[*],5m,95) #calculate the 95th percentile respons
bucket_percentile(/*/apiserver_request_duration_seconds_bucket[*],10m,99) #calculate the 99th percentile A
```

count(func_foreach(item filter,<time period>),<operator>,<pattern>)

El recuento de valores en un array devuelto por una función foreach.
 Funciones foreach compatibles: *avg_foreach*, *count_foreach*, *exists_foreach*, *last_foreach*, *max_foreach*, *min_foreach*, *sum_foreach*.

Parámetros:

- **func_foreach** - función foreach para la que se debe contar el número de valores devueltos. Consulte foreach functions para obtener más detalles. Tenga en cuenta que count_foreach y bucket_rate_foreach admiten parámetros adicionales.
- **item filter** - consulte item filter;
- **time period** - consulte time period;
- **operator** (debe ir entre comillas dobles). `operators` compatibles:
eq - igual
ne - no igual
gt - mayor
ge - mayor o igual
lt - menor
le - menor o igual
like - coincide si contiene el patrón (distingue mayúsculas y minúsculas)
bitand - AND bit a bit
regexp - coincide con distinción de mayúsculas y minúsculas con la expresión regular dada en `pattern`
iregexp - coincide sin distinción de mayúsculas y minúsculas con la expresión regular dada en `pattern`
- **pattern** - el patrón requerido (los argumentos de tipo string deben ir entre comillas dobles); compatible si se especifica *operator* en el tercer parámetro.

Comentarios:

- Usar **count()** con una función foreach relacionada con el historial (max_foreach, avg_foreach, etc.) puede tener implicaciones de rendimiento, mientras que usar **exists_foreach()**, que solo funciona con datos de configuración, no tendrá ese efecto.
- Los parámetros opcionales *operator* o *pattern* no pueden dejarse vacíos después de una coma; solo pueden omitirse por completo.
- Con *bitand* como tercer parámetro, el cuarto parámetro `pattern` puede especificarse como dos números, separados por '/': **number_to_compare_with/mask**. count() calcula el "AND bit a bit" del valor y la *mask* y compara el resultado con *number_to_compare_with*. Si el resultado del "AND bit a bit" es igual a *number_to_compare_with*, el valor se cuenta.
Si *number_to_compare_with* y *mask* son iguales, solo es necesario especificar la *mask* (sin '/').
- Con *regexp* o *iregexp* como tercer parámetro, el cuarto parámetro `pattern` puede ser una expresión regular normal o global (que comienza con '@'). En el caso de expresiones regulares globales, la distinción entre mayúsculas y minúsculas se hereda de la configuración de la expresión regular global. A efectos de la coincidencia regexp, los valores flotantes siempre se representarán con 4 dígitos decimales después de '.'. Tenga en cuenta también que, para números grandes, la diferencia entre la representación decimal (almacenada en la base de datos) y la binaria (utilizada por el server de Zabbix) puede afectar al cuarto dígito decimal.

Ejemplos:

```
count(max_foreach(/*/net.if.in[*],1h)) #calcula el número de items net.if.in que recibieron datos en la úl
count(last_foreach(/*/vfs.fs.size[*,pused]),"gt",95) #calcula el número de sistemas de archivos con más de
```

histogram_quantile(quantile,bucket1,value1,bucket2,value2,...)

Calcula el cuantil φ a partir de los buckets de un histograma.
 Compatible con la función foreach: *bucket_rate_foreach*.

Parámetros:

- **quantile** - $0 \leq \varphi \leq 1$;
- **bucketN, valueN** - pares introducidos manualmente (>=2) de parámetros o la respuesta de bucket_rate_foreach.

Comentarios:

- Compatible solo en items calculados;
- Corresponde funcionalmente a 'histogram_quantile' de PromQL;
- Devuelve -1 si los valores del último bucket 'Infinity' ("+*inf*") son iguales a 0.

Ejemplos:

```
histogram_quantile(0.75,1.0,last(/host/rate_bucket[1.0]),"+Inf",last(/host/rate_bucket[Inf]))
histogram_quantile(0.5,bucket_rate_foreach(//item_key,30s))
```

item_count(item filter)

El recuento de items existentes en la configuración que coinciden con los criterios del filtro.
 Tipo de valor admitido: *Integer*.

Parámetro:

- **item filter**: criterios para la selección de items; permite hacer referencia por grupo de hosts, host, clave de item y etiquetas. Se admiten comodines. Consulte item filter para más detalles.

Comentarios:

- Admitido solo en items calculados;

- Funciona como un alias de la función *count(exists_foreach(item_filter))*.

Ejemplos:

```
item_count(/*/agent.ping?[group="Host group 1"]) #calcula el número de hosts con el item agent.ping en "Ho
```

kurtosis(/host/key,(sec|#num)<:desplazamiento temporal>)

La ”curtosis” de la distribución de probabilidad en los valores recopilados dentro del período de evaluación definido. Véase también: Kurtosis.
 Tipos de valor admitidos: *Float*, *Integer*.
 Función foreach admitida: *last_foreach*.

Parámetros: consulte parámetros comunes.

Ejemplos:

```
kurtosis(/host/system.cpu.util[,user],5m)>5 #trigger on sharp spikes of CPU utilization over 5 minutes
```

mad(/host/key,(sec|#num)<:desplazamiento temporal>)

La desviación absoluta mediana de los valores recopilados dentro del período de evaluación definido. Véase también: Desviación absoluta mediana.
 Tipos de valores admitidos: *Float*, *Integer*.
 Función foreach admitida: *last_foreach*.

Parámetros: consulte parámetros comunes.

Ejemplos:

```
last(/host/system.cpu.util[,user])>avg(/host/system.cpu.util[,user],1h)+3*mad(/host/system.cpu.util[,user]
```

max(/host/key,(sec|#num)<:time shift>)

El valor más alto de un item dentro del período de evaluación definido.
 Tipos de valor admitidos: *Float*, *Integer*.
 Funciones foreach admitidas: *avg_foreach*, *count_foreach*, *exists_foreach*, *last_foreach*, *max_foreach*, *min_foreach*, *sum_foreach*.

Parámetros: consulte parámetros comunes.

Ejemplos:

```
max(/host/key,1h) - min(/host/key,1h) #calcula la diferencia entre los valores máximo y mínimo dentro de l
max(last_foreach(/*/vfs.fs.size[*,pused]?[group="Linux servers"])) #devuelve el uso de disco más alto entr
```

min(/host/key,(sec|#num)<:time shift>)

El valor más bajo de un item dentro del período de evaluación definido.
 Tipos de valor admitidos: *Float*, *Integer*.
 Funciones foreach admitidas: *avg_foreach*, *count_foreach*, *exists_foreach*, *last_foreach*, *max_foreach*, *min_foreach*, *sum_foreach*.

Parámetros: consulte los parámetros comunes.

Ejemplos:

```
max(/host/key,1h) - min(/host/key,1h) #calcula la diferencia entre los valores máximo y mínimo durante la
min(last_foreach(/*/vfs.file.contents["/sys/class/net/enp0s3/operstate"]?[group="Linux servers"])) #devuel
```

skewness(/host/key,(sec|#num)<:desplazamiento temporal>)

La asimetría de la distribución de probabilidad en los valores recopilados dentro del período de evaluación definido. Véase también: Skewness.
 Tipos de valor soportados: *Float*, *Integer*.
 Función foreach soportada: *last_foreach*.

Parámetros: consulte parámetros comunes.

Ejemplos:

```
skewness(/host/vm.memory.size[used],5m)<-20 #trigger ante caídas negativas repentinas que indican fallos/r
```

stddevpop(/host/key,(sec|#num)<:time shift>)

La desviación estándar poblacional de los valores recopilados dentro del período de evaluación definido. Véase también: Desviación estándar.
 Tipos de valor admitidos: *Float*, *Integer*.
 Función foreach admitida: *last_foreach*.

Parámetros: consulte parámetros comunes.

Ejemplos:

```
stddevpop(/host/system.cpu.util[,user],10m)>7 #trigger if CPU usage fluctuates heavily (population standar
```

stddevsamp(/host/key,(sec|#num)<:time shift>)

La desviación estándar muestral de los valores recopilados dentro del período de evaluación definido. Véase también: Desviación estándar.
 Tipos de valor admitidos: *Float*, *Integer*.
 Función foreach admitida: *last_foreach*.

Parámetros: consulte parámetros comunes.

Se requieren al menos dos valores de datos para que esta función funcione.

Ejemplos:

```
stddevsamp(/host/system.cpu.util[,user],10m)>7 #trigger if CPU usage fluctuates heavily (sample standard d
```

sum(/host/key,(sec|#num)<:time shift>)

La suma de los valores recopilados dentro del período de evaluación definido.
 Tipos de valor admitidos: *Float*, *Integer*.
 Funciones foreach admitidas: *avg_foreach*, *count_foreach*, *exists_foreach*, *last_foreach*, *max_foreach*, *min_foreach*, *sum_foreach*.

Parámetros: consulte parámetros comunes.

Ejemplos:

```
sum(last_foreach(/*/net.if.in[*]?[group="Linux servers"])) #calculate the total incoming network traffic f
sum(last_foreach(/*/vfs.fs.size[/,total]?[group="MySQL Servers"])) #calculate the total disk space for all
sum(last_foreach(/*/net.dns[*,*,*])) #calculate the total number of successful DNS checks
```

sumofsquares(/host/key,(sec|#num)<:time shift>)

La suma de cuadrados de los valores recopilados dentro del período de evaluación definido.
 Tipos de valores admitidos: *Float*, *Integer*.
 Función foreach admitida: *last_foreach*.

Parámetros: consulte parámetros comunes.

Ejemplos:

```
sumofsquares(/host/calculated.net.if.rate,1m) #la suma de cuadrados de la tasa de tráfico de red (entrante
sumofsquares(last_foreach(/*/net.if.in[*]?[group="Linux servers"])) #la suma de cuadrados de los valores m
```

varpop(/host/key,(sec|#num)<:time shift>)

La varianza poblacional de los valores recopilados dentro del período de evaluación definido. Véase también: Varianza.
 Tipos de valores admitidos: *Float*, *Integer*.
 Función foreach admitida: *last_foreach*.

Parámetros: consulte parámetros comunes.

Ejemplos:

```
varpop(/host/system.cpu.util[,user],10m)>50 #trigger si el uso de CPU fluctúa mucho (la varianza es superi
```

varsamp(/host/key,(sec|#num)<:time shift>)

La varianza muestral de los valores recopilados dentro del período de evaluación definido. Véase también: Varianza.
 Tipos de valores admitidos: *Float*, *Integer*.
 Función foreach admitida: *last_foreach*.

Parámetros: consulte parámetros comunes.

Se requieren al menos dos valores de datos para que esta función funcione.

Ejemplos:

```
varsamp(/host/system.cpu.util[,user],10m)>50 #trigger si el uso de CPU fluctúa mucho (la varianza muestral
```

Consulte todas las funciones soportadas.

Funciones foreach

Descripción general

Las funciones foreach se utilizan en cálculos agregados para devolver un valor agregado por cada item que es seleccionado por el **filtro de item** utilizado. Se devuelve un array de valores.

Por ejemplo, la función `avg_foreach` devolverá un array de valores, donde cada valor es el valor histórico *promedio* del item seleccionado, durante el intervalo de tiempo especificado.

> **Attention:**
> Las funciones foreach solo son compatibles con las fórmulas de Calculated item. No se pueden invocar en claves de item simples ni usar directamente para construir expresiones de trigger. En las expresiones de trigger, se puede usar un calculated item haciendo referencia a su clave.

El filtro de item forma parte de la sintaxis utilizada por las funciones foreach. El uso de comodines es compatible en el filtro de item, por lo que los items requeridos se pueden seleccionar de forma muy flexible.

Funciones compatibles

Function	Description
avg_foreach	Devuelve el valor promedio para cada item.
bucket_rate_foreach	Devuelve pares (límite superior del bucket, valor de tasa) adecuados para su uso en la función `histogram_quantile()`, donde "límite superior del bucket" es el valor del parámetro de clave del item definido por el parámetro <parameter number>.
count_foreach	Devuelve el número de valores para cada item.
exists_foreach	Devuelve '1' para cada item habilitado.
last_foreach	Devuelve el último valor para cada item.
max_foreach	Devuelve el valor máximo para cada item.
min_foreach	Devuelve el valor mínimo para cada item.
sum_foreach	Devuelve la suma de los valores para cada item.

Sintaxis de la función

Las funciones foreach admiten dos parámetros comunes: `filtro de item` (consulte los detalles a continuación) y `período de tiempo`:

```
foreach_function(item filter,time period)
```

Por ejemplo:

```
avg_foreach(/*/mysql.qps?[group="MySQL Servers"],5m)
```

devolverá el promedio de cinco minutos de cada item 'mysql.qps' en el grupo de servidores MySQL.

Tenga en cuenta que algunas funciones admiten parámetros adicionales.

Sintaxis del filtro de item

El filtro de item:

```
/host/key[parameters]?[conditions]
```

consta de cuatro partes, donde:

- host - nombre del host
- key - clave del item (sin parámetros)
- parameters - parámetros de la clave del item
- conditions - condiciones basadas en el grupo de host y/o etiquetas de item (como expresión)

Los espacios sólo se permiten dentro de la expresión de condiciones.

Uso de comodines

- El comodín puede utilizarse para reemplazar el nombre del host, la clave del item o un parámetro individual de la clave del item.
- O bien el host o la clave del item deben especificarse sin comodín. Así, `/host/*` y `/*/key` son filtros válidos, pero `/*/*` no es válido.
- El comodín no puede utilizarse para una *parte* del nombre del host, la clave del item o el parámetro de la clave del item.
- El comodín no coincide con más de un parámetro de la clave del item. Por lo tanto, debe especificarse un comodín para cada parámetro por separado (es decir, `key[abc,*,*]`).

Expresión de condiciones

La expresión de condiciones admite:

- operandos:
  - group - grupo de host
  - tag - etiqueta de item
  - "<texto>" - constante de cadena, con el carácter de escape \ para escapar " y \
- operadores de comparación de cadenas sensibles a mayúsculas y minúsculas: =, <>
- operadores lógicos: `and`, `or`, `not`
- agrupación con paréntesis: ( )

La cita de constantes de cadena es obligatoria. Sólo se admite la comparación de cadenas completas y sensibles a mayúsculas y minúsculas.

Warning:
Al especificar etiquetas en el filtro (es decir, `tag="tagname:value"`), se utiliza el dos puntos ":" como delimitador. Todo lo que sigue se considera el valor de la etiqueta. Por lo tanto, actualmente no se admite especificar un nombre de etiqueta que contenga ":".

Ejemplos

Puede utilizarse un filtro complejo, haciendo referencia a la clave del item, al grupo de host y a las etiquetas, como ilustran los ejemplos:

Ejemplo de sintaxis	Descripción
`/host/key[abc,*]`	Coincide con items similares en este host.
`/*/key`	Coincide con el mismo item de cualquier host.
`/*/key?[group="ABC" and tag="tagname:value"]`	Coincide con el mismo item de cualquier host del grupo ABC que tenga etiquetas 'tagname:value'.
`/*/key[a,*,c]?[(group="ABC" and tag="Tag1") or (group="DEF" and (tag="Tag2" or tag="Tag3:value"))]`	Coincide con items similares de cualquier host del grupo ABC o DEF con las etiquetas correspondientes.

Todos los items referenciados deben existir y recoger datos. Sólo se incluyen en los cálculos los items habilitados en hosts habilitados. No se incluyen los items en estado no soportado.

Attention:
Si se cambia la clave del item de un item referenciado, el filtro debe actualizarse manualmente.

La especificación de un grupo de host principal incluye el grupo principal y todos los grupos de host anidados con sus items.

Período de tiempo

El parámetro **segundo** permite especificar el período de tiempo para la agregación.
El período de tiempo solo puede expresarse como tiempo; no se admite la cantidad de valores (con el prefijo **#**).

En este parámetro se pueden utilizar símbolos de unidades compatibles para mayor comodidad, por ejemplo, `5m` (cinco minutos) en lugar de `300s` (300 segundos) o `1d` (un día) en lugar de `86400` (86400 segundos).

Para la función `last_foreach`, el período de tiempo es un parámetro opcional (compatible desde Zabbix 7.0) que se puede omitir:

```
last_foreach(/*/key?[group="host group"])
```

El período de tiempo no es compatible con la función `exists_foreach`.

Parámetros adicionales

bucket_rate_foreach

La función `bucket_rate_foreach` admite un tercer parámetro opcional:

```
bucket_rate_foreach(item filter,time period,<parameter number>)
```

donde `<parameter number>` es la posición del valor "bucket" en la clave del item. Por ejemplo, si el valor "bucket" en `myItem[aaa,0.2]` es `0.2`, entonces su posición es 2.

El valor predeterminado de <parameter number> es 1.

count_foreach

La función `count_foreach` admite un tercer y un cuarto parámetro opcionales:

```
count_foreach(item filter,time period,<operator>,<pattern>)
```

Donde:

- **operator** es el operador condicional para los valores del item (debe ir entre comillas dobles). Operadores compatibles:
eq - igual
ne - no igual
gt - mayor
ge - mayor o igual
lt - menor
le - menor o igual
like - coincide si contiene el patrón (distingue mayúsculas y minúsculas)
bitand - AND bit a bit
regexp - coincide con distinción de mayúsculas y minúsculas con la expresión regular dada en `pattern`
iregexp - coincide sin distinción de mayúsculas y minúsculas con la expresión regular dada en `pattern`.

- **pattern** es el patrón requerido (los argumentos de tipo cadena deben ir entre comillas dobles); se admite si se especifica *operator* en el tercer parámetro.

Comentarios:

- Los parámetros opcionales *operator* o *pattern* no pueden dejarse vacíos después de una coma; solo pueden omitirse por completo.
- Con *bitand* como tercer parámetro, el cuarto parámetro `pattern` puede especificarse como dos números, separados por '/': **number_to_compare_with/mask**. `count_foreach()` calcula el "AND bit a bit" del valor y la *mask* y compara el resultado con *number_to_compare_with*. Si el resultado del "AND bit a bit" es igual a *number_to_compare_with*, el valor se cuenta.
Si *number_to_compare_with* y *mask* son iguales, solo es necesario especificar la *mask* (sin '/').
- Con *regexp* o *iregexp* como tercer parámetro, el cuarto parámetro `pattern` puede ser una expresión regular ordinaria o global (que comienza con '@'). En el caso de expresiones regulares globales, la distinción entre mayúsculas y minúsculas se hereda de la configuración de expresiones regulares globales. A efectos de la coincidencia con regexp, los valores flotantes siempre se representarán con 4 dígitos decimales después de '.'. Tenga en cuenta también que, para números grandes, la diferencia entre la representación decimal (almacenada en la base de datos) y la binaria (utilizada por el server de Zabbix) puede afectar al cuarto dígito decimal.

Consulte aggregate calculations para obtener más detalles y ejemplos sobre el uso de las funciones foreach.

Comportamiento según la disponibilidad

La siguiente tabla ilustra cómo se comporta cada función en casos de disponibilidad limitada del host/item y de los datos históricos.

Función	Host deshabilitado	Host no disponible con datos	Host no disponible sin datos	Item deshabilitado	Item no soportado	Error de recuperación de datos (SQL)
avg_foreach	ignorar	devolver avg	ignorar	ignorar	ignorar	ignorar
bucket_rate_foreach	ignorar	devolver bucket rate	ignorar	ignorar	ignorar	ignorar
count_foreach	ignorar	devolver count	0	ignorar	ignorar	ignorar
exists_foreach	ignorar	1	1	ignorar	1	n/a
last_foreach	ignorar	devolver last	ignorar	ignorar	ignorar	ignorar
max_foreach	ignorar	devolver max	ignorar	ignorar	ignorar	ignorar
min_foreach	ignorar	devolver min	ignorar	ignorar	ignorar	ignorar
sum_foreach	ignorar	devolver sum	ignorar	ignorar	ignorar	ignorar

Si el item es *ignorado*, no se añade nada a la agregación.

Funciones bit a bit

Todas las funciones enumeradas aquí son compatibles en:

- Expresiones de trigger
- Fórmulas de item calculado
- Macros de expresión

Las funciones se enumeran sin información adicional. Haga clic en la función para ver los detalles completos.

Function	Description
bitand	El valor de "AND bit a bit" de un valor de item y una máscara.
bitlshift	El desplazamiento bit a bit a la izquierda de un valor de item.
bitnot	El valor de "NOT bit a bit" de un valor de item.
bitor	El valor de "OR bit a bit" de un valor de item y una máscara.
bitrshift	El desplazamiento bit a bit a la derecha de un valor de item.
bitxor	El valor de "OR exclusivo bit a bit" de un valor de item y una máscara.

Detalles de la función

Algunas notas generales sobre los parámetros de la función:

- Los parámetros de la función se separan por una coma
- Se aceptan expresiones como parámetros
- Los parámetros opcionales de la función (o partes del parámetro) se indican mediante < >

bitand(value,mask)

El valor de ”AND a nivel de bits” de un valor de item y una máscara.
 Tipos de valores soportados: *Integer*.

Parámetros:

- **value** - el valor a comprobar;
- **mask** (obligatorio) - un entero sin signo de 64 bits (0 - 18446744073709551615).

Aunque la comparación se realiza a nivel de bits, todos los valores deben suministrarse y se devuelven en decimal. Por ejemplo, comprobar el 3er bit se hace comparando con 4, no con 100.

Ejemplos:

```
bitand(last(/host/key),12)=8 or bitand(last(/host/key),12)=4 #3er o 4º bit activado, pero no ambos al mism
bitand(last(/host/key),20)=16 #3er bit no activado y 5º bit activado
```

bitlshift(valor, bits a desplazar)

El desplazamiento a la izquierda a nivel de bits de un valor de item.
 Tipos de valores soportados: *Integer*.

Parámetros:

- **valor** - el valor a comprobar;
- **bits a desplazar** (obligatorio) - el número de bits a desplazar.

Aunque la comparación se realiza a nivel de bits, todos los valores deben suministrarse y se devuelven en decimal. Por ejemplo, la comprobación del tercer bit se realiza comparando con 4, no con 100.

bitnot(value)

El valor de ”bitwise NOT” de un valor de item.
 Tipos de valores soportados: *Integer*.

Parámetro:

- **value** - el valor a comprobar.

Aunque la comparación se realiza a nivel de bits, todos los valores deben proporcionarse y se devuelven en decimal. Por ejemplo, comprobar el tercer bit se hace comparando con 4, no con 100.

bitor(value,mask)

El valor de ”OR a nivel de bits” de un valor de item y una máscara.
 Tipos de valores soportados: *Integer*.

Parámetros:

- **value** - el valor a comprobar;
- **mask** (obligatorio) - un entero sin signo de 64 bits (0 - 18446744073709551615).

Aunque la comparación se realiza a nivel de bits, todos los valores deben suministrarse y devolverse en decimal. Por ejemplo, la comprobación del tercer bit se realiza comparando con 4, no con 100.

bitrshift(valor, bits a desplazar)

El desplazamiento a la derecha a nivel de bits de un valor de item.
 Tipos de valores soportados: *Entero*.

Parámetros:

- **valor** - el valor a comprobar;
- **bits a desplazar** (obligatorio) - el número de bits a desplazar.

Aunque la comparación se realiza a nivel de bits, todos los valores deben suministrarse y se devuelven en decimal. Por ejemplo, la comprobación del tercer bit se realiza comparando con 4, no con 100.

bitxor(value,mask)

El valor de ”OR exclusivo a nivel de bits” de un valor de item y una máscara.
 Tipos de valores soportados: *Integer*.

Parámetros:

- **value** - el valor a comprobar;
- **mask** (obligatorio) - un entero sin signo de 64 bits (0 - 18446744073709551615).

Aunque la comparación se realiza a nivel de bits, todos los valores deben proporcionarse y se devuelven en decimal. Por ejemplo, la comprobación del tercer bit se realiza comparando con 4, no con 100.

Consulte todas las funciones soportadas.

Funciones de fecha y hora

Todas las funciones enumeradas aquí son compatibles en:

- Expresiones de trigger
- Fórmulas de item calculado
- Macros de expresión

Attention:
Las funciones de fecha y hora no pueden usarse por sí solas en la expresión; debe incluirse al menos una función de otro grupo, que haga referencia al item del host, en la expresión (excepto la función nodata()). Para obtener información detallada sobre cómo funcionan las funciones de fecha y hora dentro de las expresiones, consulte Tiempo de cálculo.

Las funciones se enumeran sin información adicional. Haga clic en la función para ver los detalles completos.

Function	Description
date	La fecha actual en formato YYYYMMDD.
dayofmonth	El día del mes en el rango de 1 a 31.
dayofweek	El día de la semana en el rango de 1 a 7.
now	La cantidad de segundos transcurridos desde la Época (00:00:00 UTC, 1 de enero de 1970).
time	La hora actual en formato HHMMSS.

Detalles de la función

date

La fecha actual en formato AAAAMMDD.

Ejemplo:

`date()<20220101`

dayofmonth

El día del mes en el rango de 1 a 31.

Ejemplo:

`dayofmonth()=1`

dayofweek

El día de la semana en el rango de 1 a 7 (Lun - 1, Dom - 7).

Ejemplo (sólo días laborables):

`dayofweek()<6`

Ejemplo (sólo fin de semana):

`dayofweek()>5`

now

El número de segundos desde la Época (00:00:00 UTC, 1 de enero de 1970).

Ejemplo:

`now()<1640998800`

time

La hora actual en formato HHMMSS.

Ejemplo (solo por la noche, 00:00-06:00):

`time()<060000`

Consulte todas las funciones soportadas.

Funciones de historial

Todas las funciones enumeradas aquí son compatibles con:

- Expresiones de trigger
- Fórmulas de item calculado
- Macros de expresión

Las funciones se enumeran sin información adicional. Haga clic en una función para consultar todos los detalles.

Función	Descripción
change	La cantidad de diferencia entre el valor anterior y el más reciente.
changecount	El número de cambios entre valores adyacentes dentro del período de evaluación definido.
count	El número de valores dentro del período de evaluación definido.
countunique	El número de valores únicos dentro del período de evaluación definido.
find	Busca una coincidencia de valor dentro del período de evaluación definido.
first	El primer valor (el más antiguo) dentro del período de evaluación definido.
firstclock	La marca de tiempo del primer valor (el más antiguo) dentro del período de evaluación definido.
fuzzytime	Comprueba cuánto difiere la hora del agent pasivo de la hora del server/proxy de Zabbix.
jsonpath	Devuelve un array de resultados de consulta JSONPath.
last	El valor más reciente.
lastclock	La marca de tiempo del N.º valor más reciente dentro del período de evaluación definido.
logeventid	Comprueba si el ID de evento de la última entrada del registro coincide con una expresión regular.
logseverity	La gravedad del registro de la última entrada del registro.
logsource	Comprueba si el origen del registro de la última entrada del registro coincide con una expresión regular.
logtimestamp	La marca de tiempo del mensaje de registro del N.º valor más reciente del item de registro.
monodec	Comprueba si se ha producido una disminución monótona de los valores.
monoinc	Comprueba si se ha producido un aumento monótono de los valores.
nodata	Comprueba si no se han recibido datos.
percentile	El percentil P de un período, donde P (porcentaje) se especifica mediante el tercer parámetro.
rate	La tasa media por segundo del incremento de un contador que aumenta monótonamente dentro del período definido.
xmlxpath	Devuelve un array de resultados de consulta XML XPath.

Parámetros comunes

- `/host/key` es el primer parámetro obligatorio común para las funciones que hacen referencia al historial del item del host.
- `(sec|#num)<:time shift>` es el segundo parámetro común para las funciones que hacen referencia al historial del item del host, donde:
  - **sec**: período máximo de evaluación en segundos (se pueden utilizar sufijos de tiempo), o
  - **#num**: rango máximo de evaluación en los últimos valores recopilados (si va precedido de un signo de almohadilla).
  - **time shift** (opcional) permite mover el punto de evaluación hacia atrás en el tiempo. Consulte más detalles sobre cómo especificar el desplazamiento temporal.

Detalles de la función

Algunas notas generales sobre los parámetros de las funciones:

- Los parámetros de la función están separados por una coma
- Los parámetros opcionales de la función (o partes del parámetro) se indican con < >
- Los parámetros específicos de la función se describen con cada función
- Los parámetros `/host/key` y `(sec|#num)<:desplazamiento de tiempo>` nunca deben ir entre comillas

change(/host/key)

La cantidad de diferencia entre el valor anterior y el último.
 Tipos de valores soportados: *Float*, *Integer*, *String*, *Text*, *Log*.
 Para cadenas devuelve: 0 - los valores son iguales; 1 - los valores difieren.

Parámetros: ver parámetros comunes.

Comentarios:

- La diferencia numérica se calculará, como se ve con estos valores de ejemplo entrantes ('valor anterior' y 'último valor' = diferencia):
'1' y '5' = +4
'3' y '1' = −2
'0' y '-2.5' = −2.5.
- En triggers, útil para detectar picos repentinos (o caídas), reinicios de contadores, transiciones de estado numérico.

- Ver también: abs para comparación.

Ejemplos:

```
change(/host/system.uptime)<0 #el cambio en el tiempo de actividad del sistema ha sido negativo desde el ú
change(/host/system.cpu.load[all,avg1])>2 #la carga de CPU (durante un minuto) ha aumentado en más de 2 de
change(/host/vfs.fs.size[/,free])<-1G #el espacio libre en disco ha caído en más de 1 GB entre comprobacio
```

changecount(/host/key,(sec|#num)<:time shift>,<mode>)

El número de cambios entre valores adyacentes dentro del período de evaluación definido.
 Tipos de valores soportados: *Float*, *Integer*, *String*, *Text*, *Log*.

Parámetros:

- Ver parámetros comunes
- **mode** (debe ir entre comillas dobles) - valores posibles: *all* - cuenta todos los cambios (por defecto); *dec* - cuenta las disminuciones; *inc* - cuenta los incrementos

Comentarios:

- Para tipos de valores no numéricos, el parámetro *mode* se ignora.
- En triggers, útil para detectar cambios de estado frecuentes (indicando inestabilidad).

Ejemplos:

```
changecount(/host/icmpping,10m)>5 #el estado del ping ha cambiado más de 5 veces en 10 minutos
changecount(/host/vfs.file.contents["/sys/class/net/eth0/operstate"],1h)>5 #el estado operativo de eth0 ha
changecount(/host/proc.num[httpd],15m)>10 #el número de procesos httpd ha cambiado más de 10 veces en 15 m
changecount(/host/key,#10,"inc") #el número de incrementos de valor (en relación con el valor adyacente) e
changecount(/host/key,24h,"dec") #el número de disminuciones de valor (en relación con el valor adyacente)
```

count(/host/key,(sec|#num)<:time shift>,<operator>,<pattern>)

El número de valores dentro del período de evaluación definido.
 Tipos de valor compatibles: *Float*, *Integer*, *String*, *Text*, *Log*.

Parámetros:

- Consulte los parámetros comunes
- **operator** (debe ir entre comillas dobles) operators compatibles:
eq - igual (predeterminado para integer, float)
ne - no igual
gt - mayor que
ge - mayor o igual que
lt - menor que
le - menor o igual que
like (predeterminado para string, text, log) - coincide si contiene el patrón (sensible a mayúsculas y minúsculas)
bitand - AND bit a bit
regexp - coincide con la expresión regular proporcionada en `pattern` (sensible a mayúsculas y minúsculas)
iregexp - coincide con la expresión regular proporcionada en `pattern` (insensible a mayúsculas y minúsculas)
- **pattern** - el patrón requerido (los argumentos de tipo string deben ir entre comillas dobles)

Comentarios:

- Los items de tipo Float coinciden con la precisión de 2.22e-16.
- *like* no es compatible como operador para valores integer.
- *like* y *bitand* no son compatibles como operadores para valores float.
- Para valores string, text y log solo se admiten los operadores *eq*, *ne*, *like*, *regexp* y *iregexp*.
- Con *bitand* como operador, el cuarto parámetro `pattern` puede especificarse como dos números, separados por '/': **number_to_compare_with/mask**. count() calcula el "AND bit a bit" del valor y la *mask* y compara el resultado con *number_to_compare_with*. Si el resultado del "AND bit a bit" es igual a *number_to_compare_with*, el valor se cuenta.
Si *number_to_compare_with* y *mask* son iguales, solo es necesario especificar la *mask* (sin '/').
- Con *regexp* o *iregexp* como operador, el cuarto parámetro `pattern` puede ser una expresión regular normal o global (que comienza con '@'). En el caso de expresiones regulares globales, la distinción entre mayúsculas y minúsculas se hereda de la configuración de la expresión regular global. A efectos de la coincidencia con regexp, los valores float siempre se representarán con 4 dígitos decimales después de '.'. Tenga en cuenta también que, para números grandes, la diferencia entre la representación decimal (almacenada en la base de datos) y la binaria (utilizada por el Zabbix server) puede afectar al cuarto dígito decimal.

Ejemplos:

```
count(/host/icmpping,30m,,"0")>5 #ping ha fallado más de 5 veces en 30 minutos
count(/host/key,10m,"like","error") #el número de valores de los últimos 10 minutos hasta ahora que contie
count(/host/key,10m,,12) #el número de valores de los últimos 10 minutos hasta ahora que son iguales a '12
count(/host/key,10m,"gt",12) #el número de valores de los últimos 10 minutos hasta ahora que son mayores q
count(/host/key,#10,"gt",12) #el número de valores dentro de los últimos 10 valores hasta ahora que son ma
```

```
count(/host/key,10m:now-1d,"gt",12) #el número de valores entre 24 horas y 10 minutos y 24 horas atrás des
count(/host/key,10m,"bitand","6/7") #el número de valores de los últimos 10 minutos hasta ahora que tienen
count(/host/key,10m:now-1d) #el número de valores entre 24 horas y 10 minutos y 24 horas atrás desde ahora
```

countunique(/host/key,(sec|#num)<:time shift>,<operator>,<pattern>)

El número de valores únicos dentro del período de evaluación definido.
 Tipos de valor compatibles: *Float*, *Integer*, *String*, *Text*, *Log*.

Parámetros:

- Consulte los parámetros comunes
- **operator** (debe ir entre comillas dobles). Operadores `compatibles`:
eq - igual (predeterminado para integer, float)
ne - no igual
gt - mayor que
ge - mayor o igual que
lt - menor que
le - menor o igual que
like (predeterminado para string, text, log) - coincide si contiene el patrón (sensible a mayúsculas y minúsculas)
bitand - AND bit a bit
regexp - coincidencia sensible a mayúsculas y minúsculas de la expresión regular indicada en `pattern`
iregexp - coincidencia no sensible a mayúsculas y minúsculas de la expresión regular indicada en `pattern`
- **pattern** - el patrón requerido (los argumentos de tipo string deben ir entre comillas dobles)

Comentarios:

- Los items de tipo Float coinciden con una precisión de 2.22e-16.
- *like* no es compatible como operador para valores integer.
- *like* y *bitand* no son compatibles como operadores para valores float.
- Para valores string, text y log solo se admiten los operadores *eq*, *ne*, *like*, *regexp* y *iregexp*.
- Con *bitand* como operador, el cuarto parámetro `pattern` puede especificarse como dos números, separados por '/': **number_to_compare_with/mask**. countunique() calcula el "AND bit a bit" del valor y la *mask* y compara el resultado con *number_to_compare_with*. Si el resultado del "AND bit a bit" es igual a *number_to_compare_with*, el valor se cuenta.
Si *number_to_compare_with* y *mask* son iguales, solo es necesario especificar la *mask* (sin '/').
- Con *regexp* o *iregexp* como operador, el cuarto parámetro `pattern` puede ser una expresión regular normal o global (que comienza con '@'). En el caso de expresiones regulares globales, la distinción entre mayúsculas y minúsculas se hereda de la configuración de expresiones regulares globales. A efectos de la coincidencia con regexp, los valores float siempre se representarán con 4 dígitos decimales después de '.'. Tenga en cuenta también que, para números grandes, la diferencia entre la representación decimal (almacenada en la base de datos) y binaria (utilizada por el server de Zabbix) puede afectar al cuarto dígito decimal.

Ejemplos:

```
countunique(/host/key,10m) #el número de valores únicos de los últimos 10 minutos hasta ahora
countunique(/host/key,10m,"like","error") #el número de valores únicos de los últimos 10 minutos hasta aho
countunique(/host/key,10m,,12) #el número de valores únicos de los últimos 10 minutos hasta ahora que son
countunique(/host/key,10m,"gt",12) #el número de valores únicos de los últimos 10 minutos hasta ahora que
countunique(/host/key,#10,"gt",12) #el número de valores únicos dentro de los últimos 10 valores hasta aho
countunique(/host/key,10m:now-1d,"gt",12) #el número de valores únicos entre 24 horas y 10 minutos y 24 ho
countunique(/host/key,10m,"bitand","6/7") #el número de valores únicos de los últimos 10 minutos hasta aho
countunique(/host/key,10m:now-1d) #el número de valores únicos entre 24 horas y 10 minutos y 24 horas atrá
```

find(/host/key,(sec|#num)<:time shift>,<operator>,<pattern>)

Encuentra una coincidencia de valor dentro del período de evaluación definido.
 Tipos de valor admitidos: *Float*, *Integer*, *String*, *Text*, *Log*.
 Devuelve: 1 - encontrado; 0 - en caso contrario.

Parámetros:

- Consulte parámetros comunes
- **sec** o **#num** (opcional) - usa el valor más reciente si no se especifica
- **operator** (debe ir entre comillas dobles) `operators` admitidos:
eq - igual (predeterminado para integer, float)
ne - no igual
gt - mayor que
ge - mayor o igual que
lt - menor que
le - menor o igual que
like (predeterminado para string, text, log) - coincide si contiene la cadena indicada en `pattern` (distingue mayúsculas y minúsculas)
bitand - AND bit a bit
regexp - coincidencia sensible a mayúsculas y minúsculas con la expresión regular indicada en `pattern`
iregexp - coincidencia no sensible a mayúsculas y minúsculas con la expresión regular indicada en `pattern`
- **pattern** - el patrón requerido (los argumentos de tipo string deben ir entre comillas dobles); expresión regular compatible con Perl (PCRE) si `operator` es *regexp*, *iregexp*

Comentarios:

- Si se procesa más de un valor, se devuelve '1' si existe al menos un valor coincidente.

- *like* no es compatible como operator para valores integer.
- *like* y *bitand* no son compatibles como operators para valores float.
- Para valores string, text y log solo se admiten los operators *eq*, *ne*, *like*, *regexp* y *iregexp*.
- Con *regexp* o *iregexp* como operator, el cuarto parámetro `pattern` puede ser una expresión regular normal o global (que comienza con '@').
  En el caso de expresiones regulares globales, la distinción entre mayúsculas y minúsculas se hereda de la configuración de la expresión regular global.

Ejemplos:

```
find(/host/key,10m,"like","error") #encuentra un valor que contiene 'error' en los últimos 10 minutos hast
find(/host/agent.version,,"like","beta")=1 #Zabbix agent tiene versión beta, debe actualizarse
find(/host/log[/var/log/nginx/access.log],,"regexp"," 500 ")=1 #se ha encontrado un error interno del web
```

first(/host/key,sec<:time shift>)

El primer (el más antiguo) valor dentro del período de evaluación definido.
 Tipos de valores soportados: *Float*, *Integer*, *String*, *Text*, *Log*.

Parámetros:

- Ver parámetros comunes

Ver también last().

Ejemplo:

```
first(/host/key,1h) #recupera el valor más antiguo dentro de la última hora hasta ahora
```

firstclock(/host/key,sec<:time shift>)

La marca de tiempo del valor más antiguo dentro del período de evaluación definido.
 Tipos de valores soportados: *Float*, *Integer*, *String*, *Text*, *Log*.

Parámetros:

- Ver parámetros comunes.

La función falla con un error si no se han recogido datos en el período dado.

Ver también lastclock().

Ejemplos:

```
firstclock(/host/key,1h) #recupera la marca de tiempo del valor más antiguo dentro de la última hora
firstclock(/host/key,1h:now-24h) #recupera la marca de tiempo del valor más antiguo dentro de la última ho
```

fuzzytime(/host/key,sec)

Comprueba cuánto difiere la hora del agent pasivo de la hora del server/proxy de Zabbix.
 Tipos de valor compatibles: *Float*, *Integer*.
 Devuelve: 1 - la diferencia entre el valor del item pasivo (como marca de tiempo) y la marca de tiempo del server/proxy de Zabbix (el reloj de la recolección del valor) es menor o igual que *sec* segundos; 0 - en caso contrario.

Parámetros:

- Consulte los parámetros comunes.

Comentarios:

- Normalmente se usa con el item 'system.localtime' para comprobar que la hora local está sincronizada con la hora local del server de Zabbix. *Tenga en cuenta* que 'system.localtime' debe configurarse como una comprobación pasiva para Zabbix agent; en Zabbix agent 2 puede configurarse como una comprobación activa.
- También puede usarse con la clave `vfs.file.time[/path/file,modify]` para comprobar que el archivo no ha recibido actualizaciones durante mucho tiempo.
- No se recomienda usar esta función en expresiones de trigger complejas (con varios items implicados), porque puede provocar resultados inesperados (la diferencia de tiempo se medirá con la métrica más reciente), por ejemplo, en `fuzzytime(/Host/system.localtime,60s)=0 or last(/Host/trap)<>0`.

Ejemplos:

```
fuzzytime(/host/system.localtime,5m)=0 #la hora local del cliente difiere de la hora del server/proxy de Z
fuzzytime(/host/system.localtime,5m)=0 and nodata(/host/system.localtime,10m)=0 #la hora local del cliente
```

jsonpath(/host/key,(sec|#num)<:time shift>,pattern)

Devuelve un array de resultados de consultas JSONPath.
 Tipos de valores compatibles: *String*, *Text*, *Log*.

Parámetros:

- Consulte los parámetros comunes
- **pattern** - el patrón requerido (los argumentos de tipo string deben estar entre comillas dobles)

Comentarios:

- Esta función se utiliza con funciones de agregación.
- Se ejecuta una consulta JSONPath en cada valor del historial seleccionado; cada resultado correcto y no vacío de la consulta se añade al array. Si el resultado de una consulta para un único valor del historial contiene varias coincidencias, cada coincidencia se añade al array por separado.
- La función devuelve un error si:
  - El formato de `pattern` no es válido.
  - Cualquiera de los valores del historial contiene JSON no válido o la consulta JSONPath falla para ese valor.
- La función devuelve un array vacío si:
  - No se seleccionó ningún valor del historial.
  - Todos los valores del historial seleccionados contienen JSON no válido.

Ejemplo:

```
max(jsonpath(/host/http.response,5m,"$.response_time")) > 500
```

last(/host/key,<#num<:time shift»)

El valor más reciente.
 Tipos de valores soportados: *Float*, *Integer*, *String*, *Text*, *Log*.

Parámetros:

- Ver parámetros comunes
- **#num** (opcional) - el N-ésimo valor más reciente

Comentarios:

- Tenga en cuenta que un período de tiempo con almohadilla (#N) funciona de manera diferente aquí que con muchas otras funciones. Por ejemplo: `last(/host/key)` siempre es igual a `last(/host/key,#1)`; `last(/host/key,#3)` - el tercer valor más reciente (*no* los tres valores más recientes).
- Zabbix no garantiza el orden exacto de los valores si existen más de dos valores dentro de un segundo en el historial.
- Ver también first().

Ejemplos:

```
last(/host/key) #recuperar el último valor
last(/host/key,#2) #recuperar el valor anterior
last(/host/key,#1)<>last(/host/key,#2) #los dos últimos valores difieren
```

lastclock(/host/key,<#num<:time shift»)

La marca de tiempo del N-ésimo valor más reciente dentro del período de evaluación definido.
 Tipos de valores soportados: *Float*, *Integer*, *String*, *Text*, *Log*.

Parámetros:

- Ver parámetros comunes;
- **#num** (opcional) - el N-ésimo valor más reciente.

La función falla con un error si no se han recolectado datos en el período dado o no se ha recolectado el valor N-ésimo.

Ver también firstclock().

Ejemplos:

```
lastclock(/host/key) #recuperar la marca de tiempo del valor más reciente
lastclock(/host/key,#2) #recuperar la marca de tiempo del segundo valor más reciente
now()-lastclock(/host/heartbeat.item)>300 #han pasado más de 5 minutos desde que se actualizó el item de l
lastclock(/host/system.cpu.load[all,avg1])-lastclock(/host/system.cpu.load[all,avg1],#2)>300 #el intervalo
((now()-lastclock(/host/system.cpu.load[all,avg1])<120 and last(/host/system.cpu.load[all,avg1])>5)) #el v
```

logeventid(/host/key,<#num<:time shift»,<pattern>)

Comprueba si el ID del evento de la última entrada de registro coincide con una expresión regular.
 Tipos de valor compatibles: *Log*.
 Devuelve: 0 - no coincide; 1 - coincide.

Parámetros:

- Consulte parámetros comunes
- **#num** (opcional) - el valor más reciente N-ésimo
- **pattern** (opcional) - la expresión regular que describe el patrón requerido, estilo Perl Compatible Regular Expression (PCRE) (los argumentos de cadena deben ir entre comillas dobles)

Ejemplos:

```
logeventid(/host/eventlog[Security],,"^4625$")=1 #se encontró una entrada de registro con ID que coincide
logeventid(/host/eventlog[System],,"^(6008|41)$")=1 #se encontró una entrada de registro con ID que coinci
```

logseverity(/host/key,<#num<:time shift»)

Gravedad del último registro de log.
 Tipos de valor soportados: *Log*.
 Devuelve: 0 - gravedad por defecto; N - gravedad (entero, útil para logs de eventos de Windows: 1 - Información, 2 - Advertencia, 4 - Error, 7 - Auditoría de error, 8 - Auditoría de éxito, 9 - Crítico, 10 - Detallado).

Parámetros:

- Ver parámetros comunes
- **#num** (opcional) - el N-ésimo valor más reciente

Zabbix toma la gravedad del log del campo **Information** del log de eventos de Windows.

Ejemplos:

```
logseverity(/host/log[/var/log/syslog],10m)>3 #se encontró una entrada de log con gravedad superior a "3"
logseverity(/host/eventlog[System],10m)=4 #se encontró una entrada de log con gravedad igual a "Error"
```

logsource(/host/key,<#num<:time shift»,<pattern>)

Comprueba si la fuente de registro de la última entrada de registro coincide con una expresión regular.
 Tipos de valor compatibles: *Log*.
 Devuelve: 0 - no coincide; 1 - coincide.

Parámetros:

- Consulte parámetros comunes
- **#num** (opcional) - el enésimo valor más reciente
- **pattern** (opcional) - la expresión regular que describe el patrón requerido, estilo Perl Compatible Regular Expression (PCRE) (los argumentos de cadena deben ir entre comillas dobles)

Normalmente se usa para registros de eventos de Windows.

Ejemplos:

```
logsource(/host/eventlog[Application],,"MSSQLSERVER")=1 #se encontró una entrada de registro con una fuent
logsource(/host/eventlog[System],,"Service Control Manager")=1 #se encontró una entrada de registro con un
logsource(/host/eventlog[System],,"Service Control Manager")=1 and logeventid(/host/eventlog[System],,"^70
```

logtimestamp(/host/key,<#num<:time shift»)

La marca de tiempo del mensaje de log del N-ésimo valor de item de log más reciente.
 Tipos de valor soportados: *Log*.

Parámetros:

- Ver parámetros comunes;
- **#num** (opcional) - el N-ésimo valor más reciente.

Comentarios:

- El cálculo del desplazamiento de tiempo se basa en el clock del valor del item, no en la marca de tiempo del mensaje de log;
- La función falla con un error si:
  - se recibe un item que no es de tipo log;
  - no se han recogido datos en el periodo dado;
  - no se ha recogido el valor N-ésimo;
  - el mensaje de log no contiene un valor de marca de tiempo.

Ejemplos:

```
logtimestamp(/host/key) #recuperar la marca de tiempo del último mensaje de log
logtimestamp(/host/key,#2) #recuperar la marca de tiempo del segundo mensaje de log más reciente
logtimestamp(/host/key,#3:now-1d) #recuperar la marca de tiempo del tercer mensaje de log más reciente hac
```

monodec(/host/key,(sec|#num)<:time shift>,<mode>)

Comprueba si ha habido una disminución monótona en los valores.
 Tipos de valores soportados: *Entero*.
 Devuelve: 1 - si todos los elementos en el periodo de tiempo disminuyen continuamente; 0 - en caso contrario.

Parámetros:

- Ver parámetros comunes
- **mode** (debe ir entre comillas dobles) - *weak* (cada valor es menor o igual que el anterior; por defecto) o *strict* (cada valor ha disminuido)

Ejemplos:

```
monodec(/host/system.swap.size[all,free],60s) + monodec(/host2/system.swap.size[all,free],60s) + monodec(/
monodec(/host/proc.num[nginx],10m,"strict")=1 #el número de procesos nginx ha disminuido monótonamente dur
```

monoinc(/host/key,(sec|#num)<:time shift>,<mode>)

Comprueba si ha habido un aumento monótono en los valores.
 Tipos de valores soportados: *Integer*.
 Devuelve: 1 - si todos los elementos en el período de tiempo aumentan continuamente; 0 - en caso contrario.

Parámetros:

- Ver parámetros comunes
- **mode** (debe ir entre comillas dobles) - *weak* (cada valor es mayor o igual que el anterior; por defecto) o *strict* (cada valor ha aumentado)

Ejemplos:

```
monoinc(/host/system.localtime,#3,"strict")=0 #la hora local del sistema no ha aumentado de forma consiste
monoinc(/host/vfs.dir.count[/mnt/data/logs],24h,"weak")=0 #dispara si el recuento de archivos se ha estanc
```

nodata(/host/key,sec,<mode>)

Comprueba que no se hayan recibido datos.
 Tipos de valores compatibles: *Integer*, *Float*, *Character*, *Text*, *Log*.
 Devuelve: 1 - si no se han recibido datos durante el período definido; 0 - en caso contrario.

Parámetros:

- Consulte los parámetros comunes
- **sec** - el período no debe ser inferior a 30 segundos, ya que el proceso history syncer calcula esta función solo cada 30 segundos; `nodata(/host/key,0)` no está permitido
- **mode** - si se establece en *strict* (entre comillas dobles), esta función no tendrá en cuenta la disponibilidad del proxy (consulte los comentarios para obtener más información)

Comentarios:

- De forma predeterminada, los trigger 'nodata' supervisados por un proxy tienen en cuenta la disponibilidad del proxy: si el proxy deja de estar disponible, los trigger 'nodata' no se activarán inmediatamente después de restablecerse la conexión, sino que omitirán los datos correspondientes al período de retraso. Tenga en cuenta que, para los proxy pasivos, la supresión se activa si la conexión se restablece después de más de 15 segundos y no menos de 2 segundos. Para los proxy activos, la supresión se activa si la conexión se restablece después de más de 15 segundos. Para desactivar la consideración de la disponibilidad del proxy, utilice el tercer parámetro, por ejemplo: `nodata(/host/key,5m,"strict")`; en este caso, la función se activará en cuanto haya transcurrido el período de evaluación (cinco minutos) sin datos.
- Esta función mostrará un error si, dentro del período del primer parámetro:
- no hay datos y el Zabbix server se ha reiniciado
- no hay datos y el mantenimiento ha finalizado
- no hay datos y el item se ha añadido o se ha vuelto a habilitar.
- Los errores se muestran en la columna *Info* de la lista de trigger.
- Es posible que esta función no funcione correctamente si existen diferencias horarias entre el Zabbix server, el proxy y el agent. Consulte también: Requisito de sincronización horaria.

Ejemplo:

```
nodata(/host/agent.ping,5m)=1 #trigger si no se han recibido datos del Zabbix agent durante 5 minutos
```

percentile(/host/key,(sec|#num)<:time shift>,percentage)

El percentil P de un período, donde P (porcentaje) se especifica mediante el tercer parámetro.
 Tipos de valores soportados: *Float*, *Integer*.

Parámetros:

- Ver parámetros comunes
- **percentage** - un número de punto flotante entre 0 y 100 (inclusive) con hasta 4 dígitos después del punto decimal

Ejemplos:

```
percentile(/host/net.if.in[eth0,bytes],1h,95)>1000000 #el percentil 95 de la entrada de red (bytes/seg) du
percentile(/host/system.cpu.util,5m,95)>80 #el percentil 95 del porcentaje de utilización de CPU en tiempo
percentile(/host/icmppingsec[192.168.0.2],15m,95)>0.15 #la mayoría de las mediciones de latencia están por
percentile(/host/net.if.in[eth0,bytes],1h,50) #calcula el percentil 50 (el valor mediano) del tráfico de r
(percentile(/host/net.if.in[eth0,bytes],1h,50)+percentile(/host/net.if.in[eth0,bytes],1h,51))/2 #calcula e
```

rate(/host/key,sec<:time shift>)

El aumento promedio por segundo en un contador monotonamente creciente dentro del período de tiempo definido.
 Tipos de valores soportados: *Float*, *Integer*.

Parámetros:

- Ver parámetros comunes

Corresponde funcionalmente a 'rate' de PromQL.

Ejemplos:

```
rate(/host/key,30s) #si el aumento monótono durante 30 segundos es 20, esta función devuelve 0.67.
rate(/host/net.if.in[eth0,bytes],5m)>500000 #la tasa de tráfico entrante en la interfaz eth0 ha superado l
rate(/host/app.requests.count,1m)>100 #el contador de solicitudes ha aumentado a >100 solicitudes por segu
```

xmlxpath(/host/key,(sec|#num)<:time shift>,pattern)

Devuelve un array de resultados de consultas XPath XML.
 Tipos de valor compatibles: *String*, *Text*, *Log*.

Parámetros:

- Consulte los parámetros comunes
- **pattern**: el patrón requerido (los argumentos de tipo string deben estar entre comillas dobles)

Comentarios:

- Esta función se utiliza con funciones de agregación.
- Se ejecuta una consulta XPath XML en cada valor del historial seleccionado; cada resultado correcto no vacío se añade al array. Si el resultado de una consulta para un único valor del historial contiene varias coincidencias, cada coincidencia se añade al array por separado.
- La función devuelve un error si:
  - El formato de `pattern` no es válido.
  - Zabbix se compiló sin compatibilidad con XML.
  - Cualquiera de los valores del historial contiene XML no válido o la consulta XPath falla para ese valor.
- La función devuelve un array vacío si:
  - No se seleccionó ningún valor del historial.
  - Todos los valores del historial seleccionados contienen XML no válido.

Ejemplo:

```
max(xmlxpath(/host/http.response,5m,"/response/response_time")) > 500
```

Consulte todas las funciones soportadas.

Funciones de tendencia

Las funciones de tendencia, a diferencia de las funciones de history, usan datos de trend para los cálculos.

Las tendencias almacenan valores agregados por hora. Las funciones de tendencia usan estos promedios horarios y, por lo tanto, son útiles para el análisis a largo plazo.

Los resultados de las funciones de tendencia se almacenan en caché, por lo que varias llamadas a la misma función con los mismos parámetros obtienen información de la base de datos solo una vez. La caché de funciones de tendencia está controlada por el parámetro de servidor TrendFunctionCacheSize.

Use `zabbix[tcache,cache,paccessed]` para estimar cuánto de la caché se está utilizando realmente. Un `paccessed` bajo, junto con `pitems` y `pmisses` bajos, normalmente significa que el tamaño de la caché se puede reducir.

Los triggers que hacen referencia **solo** a funciones de tendencia se evalúan una vez por el período de tiempo más pequeño de la expresión. Por ejemplo, un trigger como

```
trendavg(/host/key,1d:now/d) > 1 or trendavg(/host/key2,1w:now/w) > 2
```

se evaluará una vez al día. Si el trigger contiene funciones de tendencia y de history (o date and time y/o nodata()), se calcula de acuerdo con los principios habituales.

Todas las funciones enumeradas aquí son compatibles con:

- Expresiones de trigger
- Items calculados
- Macros de expresión

Las funciones se enumeran sin información adicional. Haga clic en la función para ver los detalles completos.

Function	Description
baselinedev	Devuelve el número de desviaciones (mediante el algoritmo stddevpop) entre el último período de datos y los mismos períodos de datos en las temporadas anteriores.
baselinewma	Calcula la línea base promediando datos del mismo intervalo de tiempo en múltiples períodos de tiempo iguales ('temporadas') mediante el algoritmo de media móvil ponderada.
trendavg	El promedio de los valores de tendencia dentro del período de tiempo definido.
trendcount	La cantidad de valores de history recuperados correctamente que se usaron para calcular el valor de tendencia dentro del período de tiempo definido.
trendmax	El máximo de los valores de tendencia dentro del período de tiempo definido.
trendmin	El mínimo de los valores de tendencia dentro del período de tiempo definido.
trendstl	Devuelve la tasa de anomalías durante el período de detección: un valor decimal entre 0 y 1 que es `((the number of anomaly values)/(total number of values))`.
trendsum	La suma de los valores de tendencia dentro del período de tiempo definido.

Parámetros comunes

- `/host/key` es un primer parámetro obligatorio común
- `periodo de tiempo:desplazamiento de tiempo` es un segundo parámetro común, donde:
  - **periodo de tiempo** - el periodo de tiempo (mínimo '1h'), definido como <N><unidad de tiempo> donde `N` - el número de unidades de tiempo, `unidad de tiempo` - h (hora), d (día), w (semana), M (mes) o y (año).
  - **desplazamiento de tiempo** - el desplazamiento del periodo de tiempo (ver ejemplos de funciones)

Detalles de la función

Algunas notas generales sobre los parámetros de la función:

- Los parámetros de la función están separados por una coma
- Los parámetros opcionales de la función (o partes del parámetro) se indican con < >
- Los parámetros específicos de la función se describen con cada función
- Los parámetros `/host/key` y `período de tiempo:desplazamiento de tiempo` nunca deben ir entre comillas

baselinedev(/host/key,data period:time shift,season unit,num seasons)

Devuelve el número de desviaciones (por el algoritmo stddevpop) entre el último periodo de datos y los mismos periodos de datos en temporadas anteriores.

Parámetros:

- Ver parámetros comunes
- **data period** - el periodo de recopilación de datos dentro de una temporada, definido como <N><time unit> donde:
`N` - el número de unidades de tiempo
`time unit` - h (hora), d (día), w (semana), M (mes) o y (año), debe ser igual o menor que la temporada
- **season unit** - la unidad de calendario que define una temporada (h, d, w, M, y); no puede ser menor que el periodo de datos
- **num seasons** - el número de temporadas a evaluar

Ejemplos:

```
baselinedev(/host/key,1d:now/d,"M",6) #calcula el número de desviaciones estándar (población) entre el día
baselinedev(/host/key,1h:now/h,"d",10)  #calcula el número de desviaciones estándar de la población entre
```

baselinewma(/host/key,data period:time shift,season unit,num seasons)

Calcula la línea base promediando los datos del mismo intervalo de tiempo en varios períodos de tiempo iguales ('temporadas') utilizando el algoritmo de media móvil ponderada.

Parámetros:

- Ver parámetros comunes

- **data period** - el período de recopilación de datos dentro de una temporada, definido como <N><time unit> donde:
N - el número de unidades de tiempo
time unit - h (hora), d (día), w (semana), M (mes) o y (año), debe ser igual o menor que la temporada
Time shift - el desplazamiento del período de tiempo, define el final del intervalo de recopilación de datos en temporadas (ver ejemplos)
- **season unit** - la unidad de calendario que define una temporada (h, d, w, M, y); no puede ser menor que el período de datos
- **num seasons** - el número de temporadas a evaluar

Ejemplos:

```
baselinewma(/host/key,1h:now/h,"d",3) #calcula la línea base de la misma hora del día durante los últimos
baselinewma(/host/key,2h:now/h,"d",3) #calcula la línea base del mismo intervalo de dos horas durante los
baselinewma(/host/key,1d:now/d,"M",4) #calcula la línea base del mismo día del mes que 'ayer' durante los
```

trendavg(/host/key,periodo de tiempo:desplazamiento de tiempo)

El promedio de los valores de tendencia dentro del período de tiempo definido.

Parámetros:

- Ver parámetros comunes

Ejemplos:

```
trendavg(/host/key,1h:now/h) #el promedio de la hora anterior (por ejemplo, 12:00-13:00)
trendavg(/host/key,1h:now/h-1h) #el promedio de hace dos horas (11:00-12:00)
trendavg(/host/key,1h:now/h-2h) #el promedio de hace tres horas (10:00-11:00)
trendavg(/host/key,1M:now/M-1y) #el promedio del mes anterior hace un año
```

trendcount(/host/key,periodo de tiempo:desplazamiento de tiempo)

El número de valores de historial recuperados con éxito utilizados para calcular el valor de tendencia dentro del período de tiempo definido.

Parámetros:

- Ver parámetros comunes

Ejemplos:

```
trendcount(/host/key,1h:now/h) #el recuento de valores para la hora anterior (por ejemplo, 12:00-13:00)
trendcount(/host/key,1h:now/h-1h) #el recuento de valores para hace dos horas (11:00-12:00)
trendcount(/host/key,1h:now/h-2h) #el recuento de valores para hace tres horas (10:00-11:00)
trendcount(/host/key,1M:now/M-1y) #el recuento de valores para el mes anterior hace un año
```

trendmax(/host/key,time period:time shift)

El máximo en los valores de tendencia dentro del período de tiempo definido.

Parámetros:

- Ver parámetros comunes

Ejemplos:

```
trendmax(/host/key,1h:now/h) #el máximo de la hora anterior (por ejemplo, 12:00-13:00)
trendmax(/host/key,1h:now/h) - trendmin(/host/key,1h:now/h) → calcula la diferencia entre los valores máxi
trendmax(/host/key,1h:now/h-1h) #el máximo de hace dos horas (11:00-12:00)
trendmax(/host/key,1h:now/h-2h) #el máximo de hace tres horas (10:00-11:00)
trendmax(/host/key,1M:now/M-1y) #el máximo del mes anterior hace un año
```

trendmin(/host/key,time period:time shift)

El mínimo en los valores de tendencia dentro del período de tiempo definido.

Parámetros:

- Ver parámetros comunes

Ejemplos:

```
trendmin(/host/key,1h:now/h) #el mínimo de la hora anterior (por ejemplo, 12:00-13:00)
trendmax(/host/key,1h:now/h) - trendmin(/host/key,1h:now/h) → calcular la diferencia entre los valores máx
trendmin(/host/key,1h:now/h-1h) #el mínimo de hace dos horas (11:00-12:00)
```

```
trendmin(/host/key,1h:now/h-2h) #el mínimo de hace tres horas (10:00-11:00)
trendmin(/host/key,1M:now/M-1y) #el mínimo del mes anterior hace un año
```

trendstl(/host/key,eval period:time shift,detection period,season,<deviations>,<devalg>,<s window>)

Devuelve la tasa de anomalías durante el periodo de detección: un valor decimal entre 0 y 1 que es `((el número de valores anómalos)/(número total de valores))`.

Parámetros:

- Ver parámetros comunes
- **eval period** - el periodo de tiempo que debe descomponerse (mínimo '1h'), definido como <N><unidad de tiempo> donde
`N` - el número de unidades de tiempo
`unidad de tiempo` - h (hora), d (día), w (semana), M (mes) o y (año)
- **detection period** - el periodo de tiempo antes del final del eval period para el que se calculan las anomalías (mínimo '1h', no puede ser mayor que eval period), definido como <N><unidad de tiempo> donde
`N` - el número de unidades de tiempo
`unidad de tiempo` - h (hora), d (día), w (semana)
- **season** - el periodo de tiempo más corto donde se espera un patrón repetitivo ("estacionalidad") (mínimo '2h', no puede ser mayor que eval period, el número de entradas en eval period debe ser mayor que dos veces la frecuencia resultante (season/h)), definido como <N><unidad de tiempo> donde
`N` - el número de unidades de tiempo
`unidad de tiempo` - h (hora), d (día), w (semana)
- **deviations** - el número de desviaciones (calculadas por devalg) para contar como anomalía (puede ser decimal), (debe ser mayor o igual a 1, el valor por defecto es 3)
- **devalg** (debe ir entre comillas dobles) - el algoritmo de desviación, puede ser *stddevpop*, *stddevsamp* o *mad* (por defecto)
- **s window** - el rango (en retardos) de la ventana loess para la extracción estacional (por defecto es 10 * número de entradas en eval period + 1)

Ejemplos:

```
trendstl(/host/key,100h:now/h,10h,2h) #analiza las últimas 100 horas de datos de tendencias, encuentra la
trendstl(/host/key,100h:now/h-10h,100h,2h,2.1,"mad") #analiza el periodo de 100 horas de datos de tendenci
trendstl(/host/key,100d:now/d-1d,10d,1d,4,,10) #analiza 100 días de datos de tendencias hasta hace un día,
trendstl(/host/key,1M:now/M-1y,1d,2h,,"stddevsamp") #analiza el mes anterior hace un año, encuentra la tas
```

trendsum(/host/key,time period:time shift)

La suma de los valores de tendencia dentro del período de tiempo definido.

Parámetros:

- Ver parámetros comunes

Ejemplos:

```
trendsum(/host/key,1h:now/h) #la suma de la hora anterior (por ejemplo, 12:00-13:00)
trendsum(/host/key,1h:now/h-1h) #la suma de hace dos horas (11:00-12:00)
trendsum(/host/key,1h:now/h-2h) #la suma de hace tres horas (10:00-11:00)
trendsum(/host/key,1M:now/M-1y) #la suma del mes anterior hace un año
```

Consulte todas las funciones soportadas.

Funciones matemáticas

Todas las funciones enumeradas aquí son compatibles en:

- Expresiones de trigger
- Fórmulas de item calculado
- Macros de expresión

Las funciones matemáticas son compatibles con tipos de valor float e integer, salvo que se indique lo contrario.

Las funciones se enumeran sin información adicional. Haga clic en la función para ver los detalles completos.

Function	Description
abs	El valor absoluto de un valor.
acos	El arcocoseno de un valor como un ángulo, expresado en radianes.
asin	El arcoseno de un valor como un ángulo, expresado en radianes.
atan	La arcotangente de un valor como un ángulo, expresado en radianes.

Function	Description
atan2	La arcotangente de las coordenadas de ordenada (valor) y abscisa especificadas como un ángulo, expresado en radianes.
avg	El valor promedio de los valores del item referenciado.
cbrt	La raíz cúbica de un valor.
ceil	Redondea el valor hacia arriba al entero mayor o igual más cercano.
cos	El coseno de un valor, donde el valor es un ángulo expresado en radianes.
cosh	El coseno hiperbólico de un valor.
cot	La cotangente de un valor, donde el valor es un ángulo expresado en radianes.
degrees	Convierte un valor de radianes a grados.
e	El número de Euler (2.718281828459045).
exp	El número de Euler elevado a la potencia de un valor.
expm1	El número de Euler elevado a la potencia de un valor menos 1.
floor	Redondea el valor hacia abajo al entero menor o igual más cercano.
log	El logaritmo natural.
log10	El logaritmo decimal.
max	El valor más alto de los valores del item referenciado.
min	El valor más bajo de los valores del item referenciado.
mod	El resto de la división.
pi	La constante Pi (3.14159265358979).
power	La potencia de un valor.
radians	Convierte un valor de grados a radianes.
rand	Devuelve un valor entero aleatorio.
round	Redondea el valor a un número de decimales.
signum	Devuelve '-1' si un valor es negativo, '0' si un valor es cero, '1' si un valor es positivo.
sin	El seno de un valor, donde el valor es un ángulo expresado en radianes.
sinh	El seno hiperbólico de un valor, donde el valor es un ángulo expresado en radianes.
sqrt	La raíz cuadrada de un valor.
sum	La suma de los valores del item referenciado.
tan	La tangente de un valor.
truncate	Trunca el valor a un número de decimales.

Detalles de la función

Algunas notas generales sobre los parámetros de la función:

- Los parámetros de la función están separados por una coma
- Se aceptan expresiones como parámetros
- Los parámetros opcionales de la función (o partes del parámetro) se indican mediante < >

abs(value)

El valor absoluto (desde 0) de un valor.

Parámetro:

- **value** - el valor a comprobar

Por ejemplo, el valor absoluto de '3' o '-3' será '3'.

Ejemplo:

```
abs(last(/host/key))>10
```

acos(value)

El arcocoseno de un valor como ángulo, expresado en radianes.

Parámetro:

- **value** - el valor a comprobar

El valor debe estar entre -1 y 1. Por ejemplo, el arcocoseno de un valor '0.5' será '2.0943951'.

Ejemplo:

```
acos(last(/host/key))
```

asin(value)

El arcoseno de un valor como ángulo, expresado en radianes.

Parámetro:

- **value** - el valor a comprobar

El valor debe estar entre -1 y 1. Por ejemplo, el arcoseno de un valor '0.5' será '-0.523598776'.

Ejemplo:

```
asin(last(/host/key))
```

atan(valor)

El arcotangente de un valor como ángulo, expresado en radianes.

Parámetro:

- **valor** - el valor a comprobar

Por ejemplo, el arcotangente de un valor '1' será '0.785398163'.

Ejemplo:

```
atan(last(/host/key))
```

atan2(value,abscissa)

El arco tangente de las coordenadas de ordenada (valor) y abscisa especificadas como un ángulo, expresado en radianes.

Parámetros:

- **value** - el valor a comprobar;
- **abscissa** - el valor de la abscisa.

Por ejemplo, el arco tangente de las coordenadas de ordenada y abscisa de un valor '1' será '2.21429744'.

Ejemplo:

```
atan2(last(/host/key),2)
```

avg(<valor1>,<valor2>,...)

El valor promedio de los valores de los items referenciados.

Parámetro:

- **valorX**: el valor devuelto por otra función que trabaja con el historial del item.

Ejemplo:

```
avg(avg(/host/key,1h),avg(/host2/key2,1h))
```

cbrt(value)

La raíz cúbica de un valor.

Parámetro:

- **value** - el valor a comprobar

Por ejemplo, la raíz cúbica de '64' será '4', de '63' será '3.97905721'.

Ejemplo:

```
cbrt(last(/host/key))
```

ceil(value)

Redondea el valor al entero mayor o igual más cercano.

Parámetro:

- **value** - el valor a comprobar

Por ejemplo, '2.4' se redondeará a '3'. Ver también floor().

Ejemplo:

```
ceil(last(/host/key))
```

cos(value)

El coseno de un valor, donde el valor es un ángulo expresado en radianes.

Parámetro:

- **value** - el valor a comprobar

Por ejemplo, el coseno de un valor '1' será '0.54030230586'.

Ejemplo:

```
cos(last(/host/key))
```

cosh(value)

El coseno hiperbólico de un valor. Devuelve el valor como un número real, no como notación científica.

Parámetro:

- **value** - el valor a comprobar

Por ejemplo, el coseno hiperbólico de un valor '1' será '1.54308063482'.

Ejemplo:

```
cosh(last(/host/key))
```

cot(value)

El cotangente de un valor, donde el valor es un ángulo expresado en radianes.

Parámetro:

- **value** - el valor a comprobar

Por ejemplo, el cotangente de un valor '1' será '0.54030230586'.

Ejemplo:

```
cot(last(/host/key))
```

degrees(value)

Convierte un valor de radianes a grados.

Parámetro:

- **value** - el valor a comprobar

Por ejemplo, un valor '1' convertido a grados será '57.2957795'.

Ejemplo:

```
degrees(last(/host/key))
```

e

El número de Euler (2.718281828459045).

Ejemplo:

```
e()
```

exp(value)

El número de Euler elevado a la potencia de un valor.

Parámetro:

- **value** - el valor a comprobar

Por ejemplo, el número de Euler elevado a la potencia de un valor '2' será '7.38905609893065'.

Ejemplo:

```
exp(last(/host/key))
```

expm1(value)

El número de Euler elevado a un valor menos 1.

Parámetro:

- **value** - el valor a comprobar

Por ejemplo, el número de Euler elevado al valor '2' menos 1 será '6.38905609893065'.

Ejemplo:

```
expm1(last(/host/key))
```

floor(value)

Redondea el valor hacia abajo al entero más pequeño o igual.

Parámetro:

- **value** - el valor a comprobar

Por ejemplo, '2.6' se redondeará hacia abajo a '2'. Ver también ceil().

Ejemplo:

```
floor(last(/host/key))
```

log(value)

El logaritmo natural.

Parámetro:

- **value** - el valor a comprobar

Por ejemplo, el logaritmo natural de un valor '2' será '0.69314718055994529'.

Ejemplo:

```
log(last(/host/key))
```

log10(value)

El logaritmo decimal.

Parámetro:

- **value** - el valor a comprobar

Por ejemplo, el logaritmo decimal de un valor '5' será '0.69897000433'.

Ejemplo:

```
log10(last(/host/key))
```

max(<valor1>,<valor2>,...)

El valor más alto de los valores de los items referenciados.

Parámetro:

- **valorX**: el valor devuelto por otra función que trabaja con el historial del item.

Ejemplo:

```
max(avg(/host/key,1h),avg(/host2/key2,1h))
```

min(<valor1>,<valor2>,...)

El valor más bajo de los valores de los items referenciados.

Parámetro:

- **valorX**: el valor devuelto por otra función que trabaja con el historial del item.

Ejemplo:

```
min(avg(/host/key,1h),avg(/host2/key2,1h))
```

mod(value,denominator)

El resto de la división.

Parámetros:

- **value** - el valor a comprobar;
- **denominator** - el denominador de la división.

Por ejemplo, el resto de la división de un valor ’5’ con denominador de división ’2’ será ’1’.

Ejemplo:

```
mod(last(/host/key),2)
```

pi

La constante Pi (3.14159265358979).

Ejemplo:

```
pi()
```

power(value,power value)

La potencia de un valor.

Parámetros:

- **value** - el valor a comprobar;
- **power value** - la N-ésima potencia a utilizar.

Por ejemplo, la tercera potencia de un valor ’2’ será ’8’.

Ejemplo:

```
power(last(/host/key),3)
```

radians(value)

Convierte un valor de grados a radianes.

Parámetro:

- **value** - el valor a comprobar

Por ejemplo, un valor ’1’ convertido a radianes será ’0.0174532925’.

Ejemplo:

```
radians(last(/host/key))
```

rand

Devuelve un valor entero aleatorio. Un número generado pseudoaleatoriamente usando el tiempo como semilla (suficiente para propósitos matemáticos, pero no para criptografía).

Ejemplo:

```
rand()
```

round(value,decimal places)

Redondea el valor a los decimales especificados.

Parámetros:

- **value** - el valor a comprobar;
- **decimal places** - especifica los decimales para redondear (0 también es posible).

Por ejemplo, el valor ’2.5482’ redondeado a 2 decimales será ’2.55’.

Ejemplo:

```
round(last(/host/key),2)
```

signum(value)

Devuelve ’-1’ si un valor es negativo, ’0’ si un valor es cero, ’1’ si un valor es positivo.

Parámetro:

- **value** - el valor a comprobar.

Ejemplo:

```
signum(last(/host/key))
```

sin(value)

El seno de un valor, donde el valor es un ángulo expresado en radianes.

Parámetro:

- **value** - el valor a comprobar

Por ejemplo, el seno de un valor '1' será '0.8414709848'.

Ejemplo:

```
sin(last(/host/key))
```

sinh(value)

El seno hiperbólico de un valor.

Parámetro:

- **value** - el valor a comprobar

Por ejemplo, el seno hiperbólico de un valor '1' será '1.17520119364'.

Ejemplo:

```
sinh(last(/host/key))
```

sqrt(value)

La raíz cuadrada de un valor.
 Esta función fallará con un valor negativo.

Parámetro:

- **value** - el valor a comprobar

Por ejemplo, la raíz cuadrada de un valor '3.5' será '1.87082869339'.

Ejemplo:

```
sqrt(last(/host/key))
```

sum(<valor1>,<valor2>,...)

La suma de los valores de los items referenciados.

Parámetro:

- **valorX**: el valor devuelto por otra función que trabaja con el histórico del item.

Ejemplo:

```
sum(avg(/host/key,1h),avg(/host2/key2,1h))
```

tan(value)

La tangente de un valor.

Parámetro:

- **value** - el valor a comprobar

Por ejemplo, la tangente de un valor '1' será '1.55740772465'.

Ejemplo:

```
tan(last(/host/key))
```

truncate(value,decimal places)

Trunca el valor a los decimales especificados.

Parámetros:

- **value** - el valor a comprobar;
- **decimal places** - especifica los decimales para truncar (0 también es posible).

Por ejemplo, un valor '2.5482' truncado a 2 decimales será '2.54'.

Ejemplo:

```
truncate(last(/host/key),2)
```

Consulte todas las funciones soportadas.

Funciones de operador

Todas las funciones enumeradas aquí son compatibles con:

- Expresiones de trigger
- Fórmulas de item calculado
- Macros de expresión

Las funciones se enumeran sin información adicional. Haga clic en una función para consultar todos los detalles.

Función	Descripción
between	Comprueba si el valor pertenece al rango indicado.
in	Comprueba si el valor es igual a al menos uno de los valores enumerados.

Detalles de la función

Algunas notas generales sobre los parámetros de la función:

- Los parámetros de la función están separados por una coma
- Se aceptan expresiones como parámetros

between(value,min,max)

Comprueba si el valor pertenece al rango dado.
 Tipos de valores soportados: *Integer*, *Float*.
 Devuelve: 1 - en rango; 0 - en caso contrario.

Parámetros:

- **value** - el valor a comprobar;
- **min** - el valor mínimo;
- **max** - el valor máximo.

Ejemplo:

```
between(last(/host/key),1,10)=1 #trigger si el valor está entre 1 y 10
```

in(value,value1,value2,...valueN)

Comprueba si el valor es igual a al menos uno de los valores listados.
 Tipos de valores soportados: *Integer*, *Float*, *Character*, *Text*, *Log*.
 Devuelve: 1 - si es igual; 0 - en caso contrario.

Parámetros:

- **value** - el valor a comprobar;
- **valueX** - valores listados (los valores de tipo string deben ir entre comillas dobles).

El valor se compara con los valores listados como números, si todos estos valores pueden convertirse a numéricos; de lo contrario, se comparan como cadenas.

Ejemplo:

```
in(last(/host/key),5,10)=1 #trigger si el último valor es igual a 5 o 10
in("text",last(/host/key),last(/host/key,#2))=1 #trigger si "text" es igual a cualquiera de los últimos 2
```

Consulte todas las funciones soportadas.

Funciones predictivas

Todas las funciones enumeradas aquí son compatibles en:

- Expresiones de trigger
- Fórmulas de item calculado
- Macros de expresión

Las funciones se enumeran sin información adicional. Haga clic en la función para ver los detalles completos.

Function	Description
forecast	El valor futuro, máximo, mínimo, delta o promedio del item.
timeleft	El tiempo en segundos necesario para que un item alcance el umbral especificado.

Parámetros comunes

- `/host/key` es el primer parámetro obligatorio común para las funciones que hacen referencia al historial del item del host.
- `(sec|#num)<:time shift>` es el segundo parámetro común para las funciones que hacen referencia al historial del item del host, donde:
  - **sec**: período máximo de evaluación en segundos (se pueden utilizar sufijos de tiempo), o
  - **#num**: rango máximo de evaluación en los últimos valores recopilados (si va precedido de un signo de almohadilla).
  - **time shift** (opcional): permite mover el punto de evaluación hacia atrás en el tiempo. Consulte más información sobre cómo especificar el desplazamiento temporal.

Detalles de la función

Algunas notas generales sobre los parámetros de la función:

- Los parámetros de la función están separados por una coma
- Los parámetros opcionales de la función (o partes del parámetro) se indican mediante < >
- Los parámetros específicos de la función se describen con cada función
- Los parámetros `/host/key` y `(sec|#num)<:desplazamiento de tiempo>` nunca deben ir entre comillas

forecast(/host/key,(sec|#num)<:time shift>,time,<fit>,<mode>)

El valor futuro, máximo, mínimo, delta o promedio del item.
 Tipos de valores compatibles: *Float*, *Integer*.

Parámetros:

- Consulte los parámetros comunes;
- **time** - el horizonte de predicción en segundos (por ejemplo, 1, 60) (se pueden utilizar sufijos de tiempo (por ejemplo, 30s, 1m)); se admiten valores negativos;
- **fit** (opcional; debe estar entre comillas dobles) - la función utilizada para ajustar los datos históricos. Ajustes compatibles:
linear - función lineal (predeterminada)
polynomialN - polinomio de grado N (1 <= N <= 6)
exponential - función exponencial
logarithmic - función logarítmica
power - función potencial
Tenga en cuenta que *polynomial1* equivale a *linear*;
- **mode** (opcional; debe estar entre comillas dobles) - el resultado solicitado. Modos compatibles:
value - valor (predeterminado)
max - máximo
min - mínimo
delta - *max-min*
avg - promedio
Tenga en cuenta que *value* estima el valor del item en el momento `now + time`; *max*, *min*, *delta* y *avg* analizan la estimación del valor del item en el intervalo entre `now` y `now + time`.

Comentarios:

- Si el valor que se devolverá es mayor que 1.7976931348623158E+308 o menor que -1.7976931348623158E+308, el valor devuelto se limita a 1.7976931348623158E+308 o -1.7976931348623158E+308, respectivamente;
- Solo pasa a ser no compatible si se utiliza incorrectamente en la expresión (tipo de item incorrecto, parámetros no válidos); de lo contrario, devuelve -1 en caso de errores;
- Consulte también información adicional sobre las funciones de trigger predictivas.

Ejemplos:

```
forecast(/host/key,#10,1h) #predice el valor del item dentro de una hora basándose en los últimos 10 valor
forecast(/host/key,1h,30m) #predice el valor del item dentro de 30 minutos basándose en los datos de la úl
forecast(/host/key,1h:now-1d,12h) #predice el valor del item dentro de 12 horas basándose en una hora de d
forecast(/host/key,1h,10m,"exponential") #predice el valor del item dentro de 10 minutos basándose en los
forecast(/host/key,1h,2h,"polynomial3","max") #predice el valor máximo que puede alcanzar el item en las p
forecast(/host/key,#2,-20m) #estima el valor del item de hace 20 minutos basándose en los dos últimos valo
```

timeleft(/host/key,(sec|#num)<:time shift>,threshold,<fit>)

El tiempo en segundos necesario para que un item alcance el umbral especificado.
 Tipos de valores compatibles: *Float*, *Integer*.

Parámetros:

- Consulte los parámetros comunes;
- **threshold**: el valor que se debe alcanzar (se pueden utilizar sufijos de unidades);
- **fit** (opcional; debe estar entre comillas dobles): consulte forecast().

Comentarios:

- Si el valor que se devolverá es mayor que 1.7976931348623158E+308, el valor devuelto se limita a 1.7976931348623158E+308;
- Devuelve 1.7976931348623158E+308 si no se puede alcanzar el umbral;
- Solo pasa a ser no compatible si se utiliza incorrectamente en la expresión (tipo de item incorrecto, parámetros no válidos); de lo contrario, devuelve -1 en caso de errores;
- Consulte también información adicional sobre las funciones de trigger predictivas.

Ejemplos:

```
timeleft(/host/key,#10,0) #el tiempo hasta que el valor del item llegue a cero según los últimos 10 valore
timeleft(/host/key,1h,100) #el tiempo hasta que el valor del item llegue a 100 según los datos de la últim
timeleft(/host/key,1h:now-1d,100) #el tiempo hasta que el valor del item llegue a 100 según los datos de u
timeleft(/host/key,1h,200,"polynomial2") #el tiempo hasta que el valor del item llegue a 200 según los dat
```

Consulte todas las funciones soportadas.

Funciones de trigger predictivas

Descripción general

A veces hay signos de un problema inminente. Estos signos pueden ser detectados para que se puedan tomar medidas con antelación para prevenir o al menos minimizar el impacto del problema.

Zabbix dispone de herramientas para predecir el comportamiento futuro del sistema monitorizado basándose en los datos históricos. Estas herramientas se implementan a través de funciones predictivas de trigger.

Funciones

Antes de configurar un trigger, es necesario definir qué es un estado de problema y cuánto tiempo se necesita para tomar medidas. Luego, hay dos formas de configurar un trigger que señale una posible situación no deseada. Primero: el trigger debe activarse cuando se espera que el sistema esté en un estado de problema después del "tiempo para actuar". Segundo: el trigger debe activarse cuando el sistema vaya a alcanzar el estado de problema en menos del "tiempo para actuar". Las funciones de trigger correspondientes a utilizar son **forecast** y **timeleft**. Tenga en cuenta que el análisis estadístico subyacente es básicamente idéntico para ambas funciones. Puede configurar un trigger de la forma que prefiera con resultados similares.

Parámetros

Ambas funciones utilizan casi el mismo conjunto de parámetros. Utilice la lista de funciones soportadas como referencia.

Intervalo de tiempo

En primer lugar, debe especificar el período histórico que Zabbix debe analizar para obtener la predicción. Lo hace de la manera habitual mediante el parámetro `período de tiempo` y el desplazamiento de tiempo opcional, como lo hace con las funciones **avg**, **count**, **delta**, **max**, **min** y **sum**.

Horizonte de previsión

(**forecast** solamente)
 El parámetro `time` especifica hasta qué punto en el futuro Zabbix debe extrapolar las dependencias que encuentra en los datos históricos. No importa si utiliza `time_shift` o no, `time` siempre se cuenta a partir del momento actual.

Umbral a alcanzar

(**timeleft** solo)
 El parámetro `threshold` especifica un valor que el item analizado debe alcanzar, sin importar si es desde arriba o desde abajo. Una vez que hemos determinado f(t) (ver más abajo), debemos resolver la ecuación f(t) = `threshold` y devolver la raíz que esté más cerca del momento actual y a la derecha del mismo, o 1.7976931348623158E+308 si no existe tal raíz.

Note:
Cuando los valores del item se acercan al umbral y luego lo cruzan, **timeleft** asume que la intersección ya está en el pasado y por lo tanto cambia a la siguiente intersección con el nivel `threshold`, si existe. La mejor práctica es usar las predicciones como complemento de los diagnósticos de problemas ordinarios, no como sustituto.[1]

Funciones de ajuste

La función `fit` por defecto es la función *lineal*. Pero si su sistema monitorizado es más complicado, tiene más opciones para elegir.

`fit`	x = f(t)
linear	$x = a + b*t$
polynomialN[2]	$x = a_0 + a_1*t + a_2*t^2 + ... + a_n*t^n$

`fit`	x = f(t)
exponential	x = a*exp(b*t)
logarithmic	x = a + b*log(t)
power	x = a*t^{b}

Modos

(**forecast** solamente)
 Cada vez que se evalúa una función de trigger, obtiene datos del período de historial especificado y ajusta una función especificada a los datos. Por lo tanto, si los datos son ligeramente diferentes, la función ajustada será ligeramente diferente. Si simplemente calculamos el valor de la función ajustada en un momento especificado en el futuro, no sabrá nada sobre cómo se espera que se comporte el item analizado entre ahora y ese momento en el futuro. Para algunas opciones de `fit` (como *polynomial*) un valor simple del futuro puede ser engañoso.

`mode`	**forecast** resultado
value	f(now + `time`)
max	$\max_{\text{now <= t <= now + time}}$ f(t)
min	$\min_{\text{now <= t <= now + time}}$ f(t)
delta	*max* - *min*
avg	promedio de f(t) (now <= t <= now + `time`) según la definición

Detalles

Para evitar cálculos con números enormes, consideramos la marca de tiempo del primer valor en el período especificado más 1 ns como un nuevo tiempo cero (el tiempo de época actual es del orden de 10^{9}, la época al cuadrado es 10^{18}, la precisión doble es de aproximadamente 10^{-16}). Se añade 1 ns para proporcionar todos los valores de tiempo positivos para los ajustes *logarítmicos* y *de potencia* que implican calcular log(t). El desplazamiento de tiempo no afecta a los ajustes *lineales*, *polinomiales*, *exponenciales* (aparte de cálculos más fáciles y precisos), pero cambia la forma de las funciones *logarítmicas* y *de potencia*.

Errores potenciales

Las funciones devuelven -1 en las siguientes situaciones:

- el período de evaluación especificado no contiene datos;
- el resultado de la operación matemática no está definido[3];
- complicaciones numéricas (desafortunadamente, para algunos conjuntos de datos de entrada, el rango y la precisión del formato de coma flotante de doble precisión se vuelven insuficientes)[4].

Note:
No se muestran advertencias ni errores si el ajuste elegido describe mal los datos proporcionados o si hay muy pocos datos para una predicción precisa.

Ejemplos y manejo de errores

Para recibir una advertencia cuando esté a punto de quedarse sin espacio libre en disco en su host, puede usar una expresión de trigger como esta:

```
timeleft(/host/vfs.fs.size[/,free],1h,0)<1h
```

Sin embargo, el código de error -1 puede entrar en juego y poner su trigger en estado de problema. Generalmente esto es bueno porque recibe una advertencia de que sus predicciones no funcionan correctamente y debería revisarlas más a fondo para averiguar por qué. Pero a veces es malo porque -1 simplemente puede significar que no se obtuvieron datos sobre el espacio libre en disco del host en la última hora. Si está recibiendo demasiadas alertas falsas positivas, considere usar una expresión de trigger más complicada [5]:

```
timeleft(/host/vfs.fs.size[/,free],1h,0)<1h and timeleft(/host/vfs.fs.size[/,free],1h,0)<>-1
```

La situación es un poco más difícil con **forecast**. En primer lugar, -1 puede o no poner el trigger en estado de problema dependiendo de si tiene una expresión como `forecast(/host/item,(...))<...` o como `forecast(/host/item,(...))>...`

Además, -1 puede ser una predicción válida si es normal que el valor del item sea negativo. Pero la probabilidad de esta situación en el mundo real es insignificante (vea cómo funciona el operador =). Así que agregue `... or forecast(/host/item,(...))=-1` o `... and forecast(/host/item,(...))<>-1` si desea o no desea tratar -1 como un problema respectivamente.

Notas al pie

[1] Por ejemplo, un trigger simple como `timeleft(/host/item,1h,X) < 1h` puede entrar en estado de problema cuando el valor del item se acerca a X y luego recuperarse repentinamente una vez que se alcanza el valor X. Si el problema es que el valor

del item está por debajo de X, use: `last(/host/item) < X or timeleft(/host/item,1h,X) < 1h` Si el problema es que el valor del item está por encima de X use: `last(/host/item) > X or timeleft(/host/item,1h,X) < 1h`

[2] El grado del polinomio puede ser de 1 a 6, *polynomial1* es equivalente a *linear*. Sin embargo, use polinomios de mayor grado con precaución. Si el período de evaluación contiene menos puntos de los necesarios para determinar los coeficientes del polinomio, el grado del polinomio se reducirá (por ejemplo, se solicita *polynomial5*, pero solo hay 4 puntos, por lo tanto se ajustará *polynomial3*).

[3] Por ejemplo, ajustar funciones *exponential* o *power* implica calcular log() de los valores del item. Si los datos contienen ceros o números negativos, obtendrá un error ya que log() solo está definido para valores positivos.

[4] Para los ajustes *linear*, *exponential*, *logarithmic* y *power* todos los cálculos necesarios pueden escribirse explícitamente. Para *polynomial* solo se puede calcular *value* sin pasos adicionales. Calcular *avg* implica calcular la antiderivada del polinomio (analíticamente). Calcular *max*, *min* y *delta* implica calcular la derivada del polinomio (analíticamente) y encontrar sus raíces (numéricamente). Resolver f(t) = 0 implica encontrar raíces del polinomio (numéricamente).

[5] Pero en este caso -1 puede hacer que su trigger se recupere del estado de problema. Para estar completamente protegido use: `timeleft(/host/vfs.fs.size[/,free],1h,0)<1h and ({TRIGGER.VALUE}=0 and timeleft(/host/vfs.fs.size[/,free], or {TRIGGER.VALUE}=1)`

Funciones de cadenas

Todas las funciones enumeradas aquí son compatibles con:

- Expresiones de trigger
- Fórmulas de item calculado
- Macros de expresión

Las funciones se enumeran sin información adicional. Haga clic en una función para consultar todos los detalles.

Function	Description
ascii	El código ASCII del carácter situado más a la izquierda del valor.
bitlength	La longitud del valor en bits.
bytelength	La longitud del valor en bytes.
char	Devuelve el carácter interpretando el valor como un código ASCII.
concat	La cadena resultante de concatenar los valores de los item referenciados o los valores constantes.
contains	Devuelve 1 si el valor contiene la subcadena especificada por `search value`; de lo contrario, devuelve 0.
insert	Inserta los caracteres o espacios especificados en la cadena de caracteres, comenzando en la posición especificada de la cadena.
jsonpath	Devuelve el resultado de JSONPath.
left	Devuelve los caracteres situados más a la izquierda del valor.
length	La longitud del valor en caracteres.
ltrim	Elimina los caracteres especificados del principio de la cadena.
mid	Devuelve una subcadena de N caracteres que comienza en la posición de carácter especificada por 'start'. Esta función es un alias de la función `substring()`.
repeat	Repite una cadena.
replace	Busca el patrón en el valor y lo reemplaza por el texto de reemplazo.
right	Devuelve los caracteres situados más a la derecha del valor.
rtrim	Elimina los caracteres especificados del final de la cadena.
substring	Devuelve una subcadena de N caracteres que comienza en la posición de carácter especificada por 'start'.
trim	Elimina los caracteres especificados del principio y del final de la cadena.
xmlxpath	Devuelve el resultado de XML XPath.

Detalles de la función

Algunas notas generales sobre los parámetros de la función:

- Los parámetros de la función están separados por una coma
- Se aceptan expresiones como parámetros
- Los parámetros de tipo cadena deben ir entre comillas dobles; de lo contrario, podrían ser malinterpretados
- Los parámetros opcionales de la función (o partes del parámetro) se indican con < >

ascii(value)

El código ASCII del carácter más a la izquierda del valor.
 Tipos de valores soportados: *String*, *Text*, *Log*.

Parámetro:

- **value** - el valor a comprobar

Por ejemplo, un valor como 'Abc' devolverá '65' (código ASCII de 'A').

Ejemplo:

```
ascii(last(/host/key))
```

bitlength(value)

La longitud del valor en bits.
 Tipos de valores soportados: *String*, *Text*, *Log*, *Integer*.

Parámetro:

- **value** - el valor a comprobar

Ejemplo:

```
bitlength(last(/host/key))
```

bytelength(value)

La longitud del valor en bytes.
 Tipos de valores soportados: *String*, *Text*, *Log*, *Integer*.

Parámetro:

- **value** - el valor a comprobar

Ejemplo:

```
bytelength(last(/host/key))
```

char(value)

Devuelve el carácter interpretando el valor como código ASCII.
 Tipos de valores soportados: *Entero*.

Parámetro:

- **value** - el valor a comprobar

El valor debe estar en el rango 0-255. Por ejemplo, un valor como '65' (interpretado como código ASCII) devolverá 'A'.

Ejemplo:

```
char(last(/host/key))
```

concat(<valor1>,<valor2>,...)

La cadena resultante de concatenar los valores de los items referenciados o valores constantes.
 Tipos de valores soportados: *String*, *Text*, *Log*, *Float*, *Integer*.

Parámetro:

- **valorX**: el valor devuelto por una de las funciones de histórico o un valor constante (cadena, entero o número flotante). Debe contener al menos dos parámetros.

Por ejemplo, un valor como 'Zab' concatenado con 'bix' (la cadena constante) devolverá 'Zabbix'.

Ejemplos:

```
concat(last(/host/key),"bix")
concat("1 min: ",last(/host/system.cpu.load[all,avg1]),", 15 min: ",last(/host/system.cpu.load[all,avg15])
```

contains(value, search value)

Devuelve 1 si el valor contiene la subcadena especificada en `search value`; de lo contrario, devuelve 0.| Tipos de valores compatibles: *String*, *Text*, *Log*.

Parámetros:

- **value**: el valor que se comprobará;
- **search value**: el valor que se buscará.

Comentarios:

- La búsqueda distingue entre mayúsculas y minúsculas.
- Cuando el valor es válido y `search value` está vacío, la función devuelve 1.

Por ejemplo, la función devuelve 1 si el valor contiene «Zabbix».

Ejemplo:

```
contains(last(/host/key),"Zabbix")
```

insert(value,start,length,replacement)

Inserta los caracteres o espacios especificados en la cadena de caracteres comenzando en la posición especificada en la cadena.
 Tipos de valor soportados: *String*, *Text*, *Log*.

Parámetros:

- **value** - el valor a comprobar;
- **start** - posición de inicio;
- **length** - posiciones a reemplazar;
- **replacement** - cadena de reemplazo.

Por ejemplo, un valor como 'Zabbbix' será reemplazado por 'Zabbix' si 'bb' (posición inicial 3, posiciones a reemplazar 2) se reemplaza por 'b'.

Ejemplo:

```
insert(last(/host/key),3,2,"b")
```

jsonpath(value,path,<default>)

Devuelve el resultado de JSONPath.
 Tipos de valores soportados: *String*, *Text*, *Log*.

Parámetros:

- **value** - el valor a comprobar;
- **path** - la ruta (debe ir entre comillas);
- **default** - el valor opcional por defecto si la consulta JSONPath no devuelve datos. Tenga en cuenta que en otros errores se devuelve un fallo (por ejemplo, "unsupported construct").

Ejemplo:

```
jsonpath(last(/host/proc.get[zabbix_agentd,,,summary]),"$..size")
```

left(value,count)

Devuelve los caracteres más a la izquierda del valor.
 Tipos de valor soportados: *String*, *Text*, *Log*.

Parámetros:

- **value** - el valor a comprobar;
- **count** - el número de caracteres a devolver.

Por ejemplo, puede devolver 'Zab' de 'Zabbix' especificando 3 caracteres más a la izquierda para devolver. Vea también right().

Ejemplo:

```
left(last(/host/key),3) #devuelve los tres caracteres más a la izquierda
```

length(value)

La longitud del valor en caracteres.
 Tipos de valores soportados: *String*, *Text*, *Log*.

Parámetro:

- **value** - el valor a comprobar.

Ejemplos:

```
length(last(/host/key)) #la longitud del valor más reciente
length(last(/host/key,#3)) #la longitud del tercer valor más reciente
length(last(/host/key,#1:now-1d)) #la longitud del valor más reciente hace un día
```

ltrim(value,<chars>)

Elimina los caracteres especificados del principio de la cadena.
 Tipos de valor soportados: *String*, *Text*, *Log*.

Parámetros:

- **value** - el valor a comprobar;
- **chars** (opcional) - especifica los caracteres a eliminar.

Por defecto, se eliminan los espacios en blanco al principio (si no se especifican caracteres opcionales). Ver también: rtrim(), trim().

Ejemplos:

```
ltrim(last(/host/key)) #elimina los espacios en blanco del principio de la cadena
ltrim(last(/host/key),"Z") #elimina cualquier 'Z' del principio de la cadena
ltrim(last(/host/key)," Z") #elimina cualquier espacio y 'Z' del principio de la cadena
```

mid(value,start,length)

Devuelve una subcadena de N caracteres que comienza en la posición de carácter especificada por 'start'.
Esta función es un alias de la función `substring()`.
 Tipos de valores compatibles: *String*, *Text*, *Log*.

Parámetros:

- **value** - el valor que se comprobará;
- **start** - posición inicial de la subcadena;
- **length** - posiciones que se devolverán en la subcadena.

Por ejemplo, es posible devolver 'abbi' de un valor como 'Zabbix' si la posición inicial es 2 y el número de posiciones que se devolverán es 4.

Ejemplo:

```
mid(last(/host/key),2,4)="abbi"
```

repeat(value,count)

Repetir una cadena.
 Tipos de valores soportados: *Cadena*, *Texto*, *Log*.

Parámetros:

- **value** - el valor a comprobar;
- **count** - el número de veces a repetir.

Ejemplo:

```
repeat(last(/host/key),2) #repetir el valor dos veces
```

replace(value,pattern,replacement)

Busca el patrón en el valor y lo reemplaza por el valor de reemplazo. Todas las ocurrencias del patrón serán reemplazadas.
 Tipos de valores soportados: *String*, *Text*, *Log*.

Parámetros:

- **value** - el valor a comprobar;
- **pattern** - el patrón a buscar;
- **replacement** - la cadena por la que se reemplazará el patrón.

Ejemplo:

```
replace(last(/host/key),"ibb","abb") #reemplaza todos los 'ibb' por 'abb'
```

right(value,count)

Devuelve los caracteres más a la derecha del valor.
 Tipos de valores soportados: *String*, *Text*, *Log*.

Parámetros:

- **value** - el valor a comprobar;
- **count** - el número de caracteres a devolver.

Por ejemplo, puede devolver 'bix' de 'Zabbix' especificando 3 caracteres más a la derecha para devolver. Vea también left().

Ejemplo:

```
right(last(/host/key),3) #devuelve los tres caracteres más a la derecha
```

rtrim(value,<chars>)

Elimina los caracteres especificados del final de la cadena.
 Tipos de valores soportados: *String*, *Text*, *Log*.

Parámetros:

- **value** - el valor a comprobar;
- **chars** (opcional) - especifica los caracteres a eliminar.

Por defecto, se eliminan los espacios en blanco a la derecha (si no se especifican caracteres opcionales). Véase también: ltrim(), trim().

Ejemplos:

```
rtrim(last(/host/key)) #elimina los espacios en blanco del final de la cadena
rtrim(last(/host/key),"x") #elimina cualquier 'x' del final de la cadena
rtrim(last(/host/key),"x ") #elimina cualquier 'x' y espacio del final de la cadena
```

substring(value,start,length)

Devuelve una subcadena de N caracteres que comienza en la posición de carácter especificada por «start».
 Tipos de valor compatibles: *String*, *Text*, *Log*.

Parámetros:

- **value** - el valor que se debe comprobar;
- **start** - posición inicial de la subcadena;
- **length** - posiciones que se deben devolver en la subcadena.

Por ejemplo, es posible devolver «abbi» de un valor como «Zabbix» si la posición inicial es 2 y el número de posiciones que se deben devolver es 4.

Ejemplo:

```
substring(last(/host/key),2,4)="abbi"
```

trim(value,<chars>)

Elimina los caracteres especificados del principio y final de la cadena.
 Tipos de valor soportados: *String*, *Text*, *Log*.

Parámetros:

- **value** - el valor a comprobar;
- **chars** (opcional) - especifica los caracteres a eliminar.

Por defecto, se eliminan los espacios en blanco de ambos lados (si no se especifican caracteres opcionales). Véase también: ltrim(), rtrim().

Ejemplos:

```
trim(last(/host/key)) #elimina los espacios en blanco del principio y final de la cadena
trim(last(/host/key),"_") #elimina '_' del principio y final de la cadena
```

xmlxpath(value,path,<default>)

Devuelve el resultado de XML XPath.
 Tipos de valores soportados: *String*, *Text*, *Log*.

Parámetros:

- **value** - el valor a comprobar;
- **path** - la ruta (debe ir entre comillas);
- **default** - el valor opcional de reserva si la consulta XML XPath devuelve un conjunto de nodos vacío. No se devolverá si el resultado vacío no es un conjunto de nodos (es decir, cadena vacía). En otros errores se devuelve un fallo (por ejemplo, ”expresión no válida”).

Ejemplo:

```
xmlxpath(last(/host/xml_result),"/response/error/status")
```

Consulte todas las funciones soportadas.

Mejores prácticas para triggers

Los triggers son una herramienta poderosa, pero también pueden generar ruido de alertas no deseado. Para ver más señales reales y menos ruido, siga estos consejos:

1. Desensibilice los triggers. En lugar de alertar sobre el último valor (demasiado alto/demasiado bajo), analice el promedio durante un período prolongado usando funciones como **avg**, **min** y **max**.
2. Considere usar la función **percentile** (configúrela en 95% o 5%) en los triggers, si desea evitar alertas por picos y caídas aleatorios.

3. Use hysteresis para evitar el flapping de triggers: cambios frecuentes del estado del trigger (OK ↔ Problem). Se puede definir un continuo para el estado de problema agregando una expresión de recuperación (una condición separada para la resolución del problema).
4. Use las dependencias de triggers para evitar alertas que no estén relacionadas con la causa raíz.
5. Use la severidad de los triggers para alertar solo sobre los problemas más graves.
6. Defina ventanas de maintenance.

Histéresis

A veces se necesita un intervalo entre los estados de problema y recuperación, en lugar de un simple umbral. Por ejemplo, si queremos definir un trigger que informe de un problema cuando la temperatura de la sala de servidores supere los 20°C y queremos que permanezca en estado de problema hasta que la temperatura baje de los 15°C, un simple umbral de trigger en 20°C no será suficiente.

En su lugar, primero debemos definir una expresión de trigger para el evento de problema (temperatura superior a 20°C). Luego debemos definir una condición de recuperación adicional (temperatura inferior a 15°C). Esto se hace definiendo una *Expresión de recuperación* al configurar un trigger.

En este caso, la recuperación del problema se produce en dos pasos:

- Primero, la expresión de problema (temperatura superior a 20°C) debe evaluarse como FALSA
- Segundo, la expresión de recuperación (temperatura inferior a 15°C) debe evaluarse como VERDADERA

La expresión de recuperación se evalúa solo después de que se resuelve el evento de problema. ¡Que la expresión de recuperación sea VERDADERA por sí sola no resuelve un problema si la expresión de problema sigue siendo VERDADERA!

Ejemplo

La temperatura en la sala de servidores es demasiado alta.

Expresión de problema:

```
last(/server/temp)>20
```

Expresión de recuperación:

```
last(/server/temp)<=15
```

> **Note:**
> No es productivo utilizar la macro {TRIGGER.VALUE} en una expresión de recuperación porque esta expresión solo se evalúa cuando el trigger está en estado de ”Problema”. En consecuencia, {TRIGGER.VALUE} siempre se resolverá como ”1” (lo que indica un estado de ”Problema”) al evaluar la expresión.

Dependencias de triggers

Las dependencias de triggers se pueden utilizar para evitar alertas que no estén relacionadas con la causa raíz.

Consulte todas las mejores prácticas.

Descripción general

A veces, la disponibilidad de un host depende de otro. Un server que se encuentra detrás de un router dejará de estar disponible si el router deja de funcionar. Si se configuran triggers para ambos, es posible que reciba notificaciones sobre dos hosts caídos, cuando en realidad el único responsable era el router.

En este caso, puede resultar útil establecer una dependencia entre los hosts. Una vez configurada la dependencia, las notificaciones de los elementos dependientes pueden retenerse y enviarse únicamente la notificación sobre el problema raíz.

Aunque Zabbix no admite dependencias directas entre hosts, estas pueden definirse mediante un método más flexible: las dependencias entre triggers. Un trigger puede depender de uno o varios triggers.

En nuestro sencillo ejemplo, abrimos el formulario de configuración del trigger del server y establecemos que depende del trigger correspondiente del router. Con esta dependencia, el trigger del server no cambiará de estado mientras el trigger del que depende se encuentre en estado ’PROBLEM’; por lo tanto, no se ejecutarán acciones dependientes ni se enviarán notificaciones.

Si tanto el server como el router están caídos y se establece una dependencia entre el trigger ”server caído” y el trigger ”router caído”, Zabbix no ejecutará acciones para el trigger dependiente.

Mientras el trigger principal se encuentre en estado PROBLEM, es posible que sus elementos dependientes informen de valores que no sean fiables. Por lo tanto, los triggers dependientes no se volverán a evaluar hasta que el trigger principal (el router en el ejemplo anterior):

- vuelva del estado 'PROBLEM' al estado 'OK';
- cambie su estado de 'PROBLEM' a 'UNKNOWN';
- se cierre manualmente, mediante correlación o con ayuda de las funciones date and time y/o nodata();
- se resuelva mediante el valor de un item que no intervenga en el trigger dependiente;
- se deshabilite, tenga un item deshabilitado o un host con un item deshabilitado

En todos los casos mencionados anteriormente, el trigger dependiente (el server) se volverá a evaluar únicamente cuando se reciba una nueva métrica para él. Esto significa que el trigger dependiente podría no actualizarse inmediatamente.

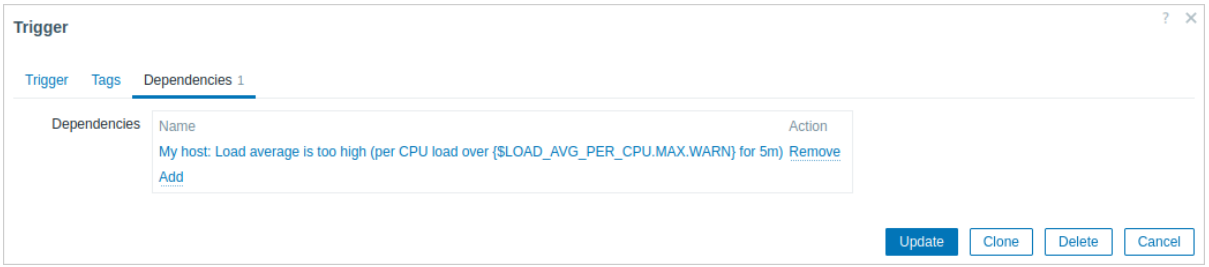
Además:

- Se puede añadir una dependencia de trigger desde cualquier trigger de host a cualquier otro trigger de host, siempre que no se genere una dependencia circular.
- Se puede añadir una dependencia de trigger de un template a otro. Si algún trigger del template A depende de algún trigger del template B, el template A solo podrá vincularse a un host (o a otro template) junto con el template B, mientras que el template B podrá vincularse por separado a un host (o a otro template).
- Se puede añadir una dependencia de trigger desde un trigger de template a un trigger de host. En este caso, vincular dicho template a un host creará un trigger de host que dependerá del mismo trigger de template del que dependía el trigger original. Esto permite, por ejemplo, tener un template en el que algunos triggers dependan de los triggers del router (host). Todos los hosts vinculados a este template dependerán de ese router específico.
- No se puede añadir una dependencia de trigger desde un trigger de host a un trigger de template.
- Se puede añadir una dependencia de trigger desde un prototipo de trigger a otro prototipo de trigger (dentro de la misma regla de descubrimiento de bajo nivel) o a un trigger real. Un prototipo de trigger no puede depender de un prototipo de trigger de una regla LLD diferente ni de un trigger creado a partir de un prototipo de trigger. Un prototipo de trigger de host no puede depender de un trigger de un template.

Configuración

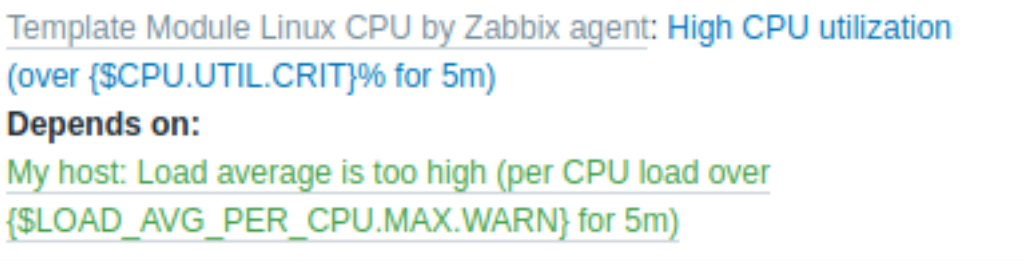
Para definir una dependencia:

1. Abra la pestaña *Dependencias* en el formulario de configuración del trigger.
2. Haga clic en *Añadir* en la sección *Dependencias* y seleccione uno o varios triggers de los que dependerá el trigger.



3. Haga clic en *Actualizar*.

Ahora el trigger muestra la indicación de su dependencia en la lista.



Ejemplo de varias dependencias

Por ejemplo, el Host está detrás de Router2 y Router2 está detrás de Router1.

```
Zabbix - Router1 - Router2 - Host
```

Si Router1 está caído, entonces, obviamente, Host y Router2 también son inalcanzables, pero recibir tres notificaciones sobre que Host, Router1 y Router2 están caídos es excesivo.

Así que, en este caso, definimos dos dependencias:

```
el trigger 'Host is down' depende del trigger 'Router2 is down'
el trigger 'Router2 is down' depende del trigger 'Router1 is down'
```

Antes de cambiar el estado del trigger 'Host is down', Zabbix comprobará las dependencias correspondientes del trigger. Si se encuentran y uno de esos triggers está en el estado `Problem`, entonces el estado del trigger no se cambiará, no se ejecutarán las acciones y no se enviarán notificaciones.

Zabbix realiza esta comprobación de forma recursiva. Si Router1 o Router2 es inalcanzable, el trigger de Host no se actualizará.

Gravedad del trigger

La gravedad del trigger puede utilizarse para evitar el ruido de alertas en problemas menos graves.

Consulte todas las mejores prácticas.

Descripción general

La gravedad representa la importancia relativa de un trigger.

Asignar una gravedad no solo es útil para la representación visual, sino también para evitar notificaciones sobre problemas triviales.

Las gravedades de los triggers se utilizan para:

- Representación visual (color distinto para cada problema creado, según la gravedad original del trigger)
- Limitar las notificaciones según la gravedad (consulte condiciones de acción)
- Medios de usuario: diferentes medios (canal de notificación) para diferentes gravedades (por ejemplo, SMS para triggers de gravedad *Alta* y *Desastre*, y correo electrónico para triggers de otras gravedades)
- Audio en alarmas globales: diferentes sonidos para diferentes gravedades

De forma predeterminada, se admiten las siguientes gravedades de trigger:

Gravedad	Color	Descripción
No clasificado	Gris	Puede utilizarse cuando se desconoce el nivel de gravedad de un evento, no se ha determinado, no forma parte del alcance de monitorización habitual, etc., por ejemplo, durante la configuración inicial, como marcador de posición para una evaluación futura o como parte de un proceso de integración.
Información	Azul claro	Puede utilizarse para eventos informativos que no requieren atención inmediata, pero que aún pueden proporcionar información valiosa.
Advertencia	Amarillo	Puede utilizarse para indicar un posible problema que podría requerir investigación o acción, pero que no es crítico.
Media	Naranja	Puede utilizarse para indicar un problema significativo que debe abordarse relativamente pronto para evitar problemas mayores.
Alta	Rojo claro	Puede utilizarse para indicar problemas críticos que requieren atención inmediata para evitar interrupciones importantes.
Desastre	Rojo	Puede utilizarse para indicar un incidente grave que requiere acción inmediata para evitar, por ejemplo, caídas del sistema o pérdida de datos.

Los nombres y colores de las gravedades de los triggers pueden **personalizarse** en *Administración* > *General* > *Opciones de visualización de triggers*. Los colores se comparten entre todos los temas del frontend.

Traducir nombres personalizados de severidad

> **Attention:**
> Si se utilizan traducciones del frontend de Zabbix, los nombres personalizados de severidad sobrescribirán los nombres traducidos por defecto.

Los nombres de severidad de trigger por defecto están disponibles para su traducción en todos los idiomas. Si se cambia un nombre de severidad, se utiliza un nombre personalizado en todos los idiomas y se necesita una traducción manual adicional.

Procedimiento para traducir nombres personalizados de severidad:

- establezca el nombre personalizado de severidad requerido, por ejemplo, 'Importante'
- edite <frontend_dir>/locale/<required_locale>/LC_MESSAGES/frontend.po
- añada 2 líneas:

```
msgid "Importante"
msgstr "<cadena de traducción>"
```

y guarde el archivo.

- cree los archivos .mo como se describe en <frontend_dir>/locale/README

Aquí **msgid** debe coincidir con el nuevo nombre personalizado de severidad y **msgstr** debe ser la traducción para él en el idioma específico.

Este procedimiento debe realizarse después de cada cambio de nombre de severidad.

Visualización

Resumen

La visualización ofrece diferentes formas de mostrar los datos de monitorización recopilados por Zabbix. Los gráficos muestran tendencias y métricas, los mapas de red muestran la topología de la infraestructura y los paneles combinan widgets personalizables para proporcionar vistas de monitorización completas.

Herramientas de visualización

- Gráficos - vea datos de item como gráficos simples generados automáticamente, cree gráficos personalizados o compare varios items en gráficos ad hoc.
- Mapas de red - cree representaciones visuales de la infraestructura de red con elementos e indicadores personalizables.
- Paneles - personalice diseños combinando una variedad de widgets de panel que visualizan datos en diferentes formatos.

Gráficos

Descripción general

Con una gran cantidad de datos que llegan a Zabbix, resulta mucho más fácil para los usuarios consultar una representación visual de lo que está ocurriendo en lugar de ver únicamente números.

Aquí es donde entran en juego los gráficos. Los gráficos permiten comprender el flujo de datos de un vistazo, correlacionar problemas, descubrir cuándo comenzó algo o ilustrar cuándo algo podría convertirse en un problema.

Zabbix proporciona a los usuarios:

- Gráficos simples integrados de los datos de un item.
- La posibilidad de crear gráficos personalizados más complejos.
- Acceso rápido a una comparación de varios items en gráficos ad hoc.
- Gráficos vectoriales modernos y personalizables, y gráficos circulares.

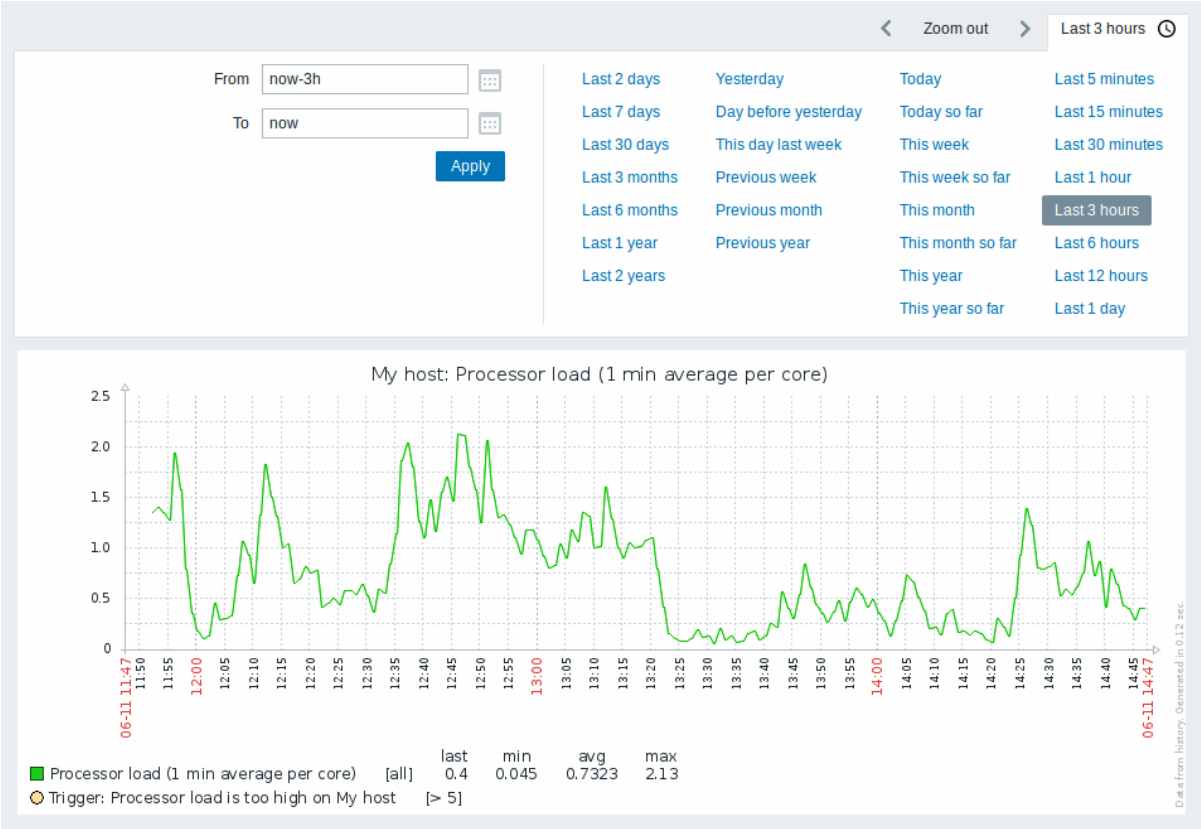
Gráficos simples

Descripción general

Los gráficos simples se proporcionan para visualizar los datos recopilados por los item.

No se requiere ninguna configuración por parte del usuario para ver los gráficos simples. Zabbix los pone a disposición libremente.

Solo tiene que ir a *Monitoring* > *Latest data* y hacer clic en el enlace *Graph* del item correspondiente para que se muestre un gráfico.



Note:
Los gráficos simples se proporcionan para todos los item numéricos. Para los item de texto, hay un enlace a *History* disponible en *Monitoring* > *Latest data*.

Selector de período de tiempo

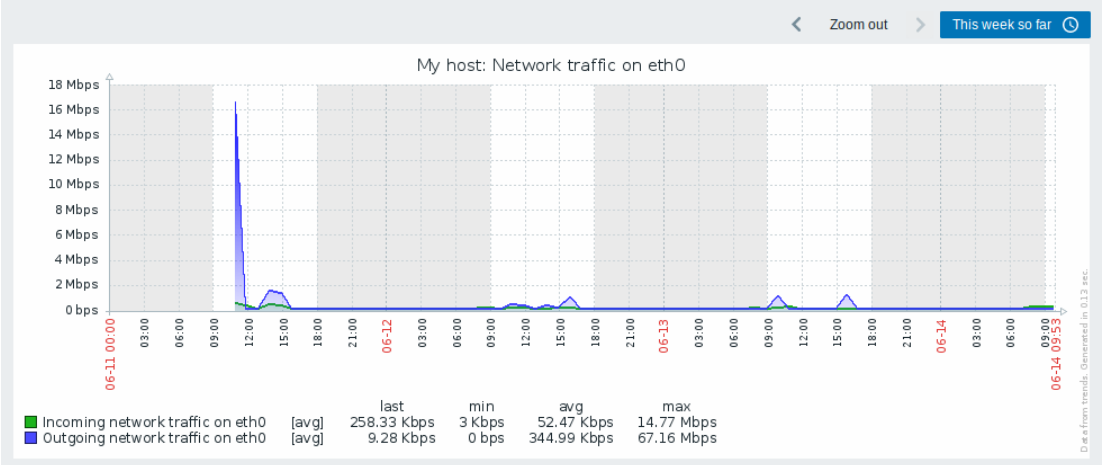
El selector de período de tiempo situado encima del gráfico permite seleccionar períodos utilizados con frecuencia con un solo clic del ratón.
Para obtener más información, consulte Selectores de período de tiempo y host.

Datos recientes frente a períodos más largos

Para datos muy recientes, se dibuja una **única** línea que conecta cada valor recibido. La línea única se dibuja siempre que haya al menos un píxel horizontal disponible para un valor.

Para los datos que muestran un período más largo, se dibujan **tres líneas**: una verde oscura muestra el promedio, mientras que una línea rosa clara y una verde clara muestran los valores máximo y mínimo en ese momento. El espacio entre los valores altos y bajos se rellena con un fondo amarillo.

El tiempo de trabajo (días laborables) se muestra en los gráficos con un fondo blanco, mientras que el tiempo no laborable se muestra en gris (con el tema frontend predeterminado *Original blue*).



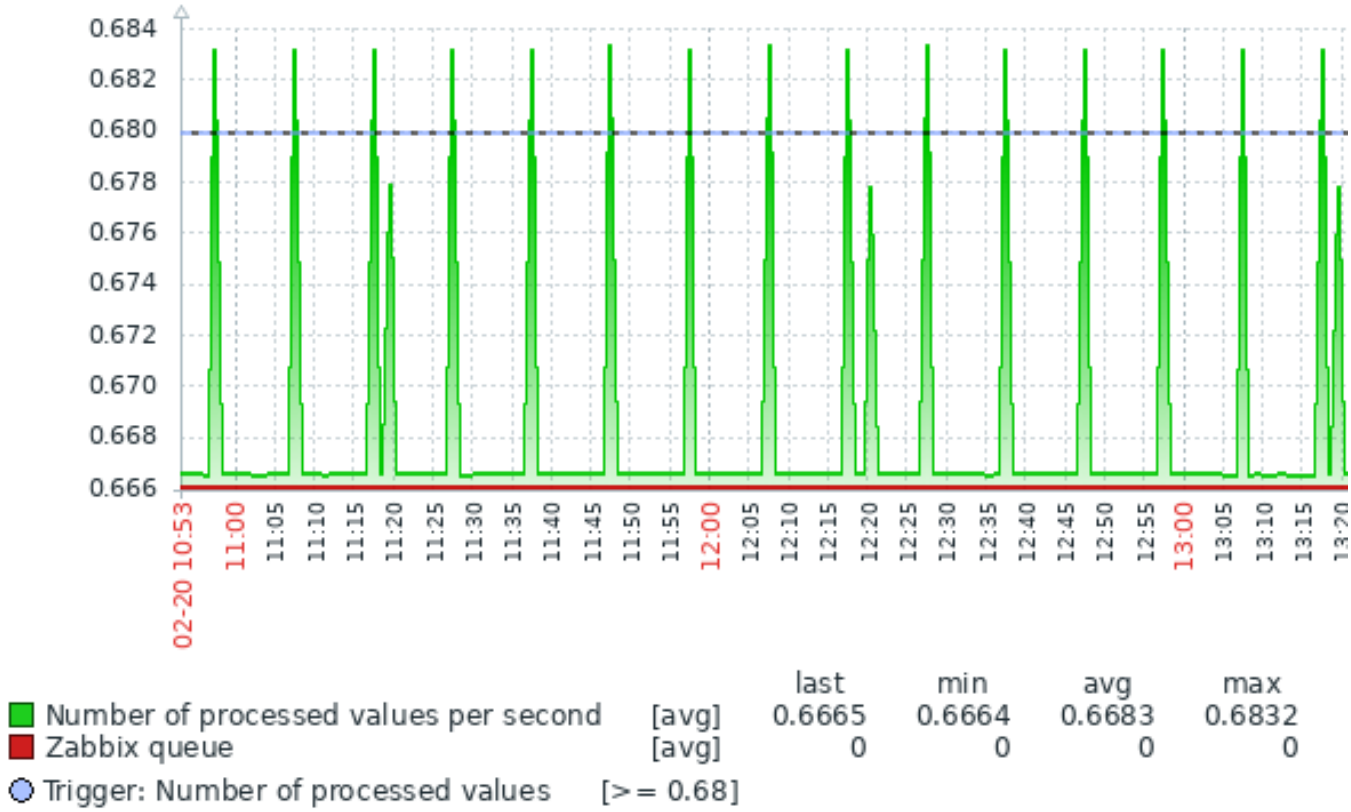
El tiempo de trabajo siempre se muestra en los gráficos simples, mientras que su visualización en los gráficos personalizados es

una preferencia del usuario.

El tiempo de trabajo no se muestra si el gráfico abarca más de 3 meses.

Líneas de trigger

Los triggers simples se muestran como líneas con guiones negros sobre el color de severidad del trigger; observe la línea azul en el gráfico y la información del trigger mostrada en la leyenda. Se pueden mostrar hasta 3 líneas de trigger en el gráfico; si hay más triggers, se priorizan los triggers con menor severidad. Los triggers siempre se muestran en los gráficos simples, mientras que su visualización en los gráficos personalizados es una preferencia del usuario.



Generación a partir del historial/tendencias

Los gráficos pueden dibujarse basándose en el historial o las tendencias de un item.

Para los usuarios que tienen activado el modo de depuración del frontend, se muestra un texto vertical gris en la esquina inferior derecha del gráfico, que indica de dónde proceden los datos.

Varios factores influyen en si se utiliza el historial o las tendencias:

- Antigüedad del historial del item. Por ejemplo, el historial de un item puede conservarse durante 14 días. En ese caso, todos los datos con una antigüedad superior a catorce días procederán de las tendencias.
- Congestión de datos en el gráfico. Si la cantidad de segundos que se muestran en un píxel horizontal del gráfico supera 3600/16, se muestran los datos de tendencias (aunque el historial del item siga estando disponible para el mismo período).
- Si las tendencias están deshabilitadas, se utiliza el historial del item para generar el gráfico, si está disponible para ese período.

Ausencia de datos

Para los items con un intervalo de actualización regular, no se muestra nada en el gráfico si no se recopilan datos del item.

Sin embargo, para los items trapper y los items con un intervalo de actualización programado (y el intervalo de actualización regular establecido en 0), se dibuja una línea recta que conduce hasta el primer valor recopilado y desde el último valor recopilado hasta el final del gráfico; la línea está al nivel del primer/último valor respectivamente.

Cambiar a valores sin procesar

Un menú desplegable en la parte superior derecha permite cambiar del gráfico simple a las listas de *Valores/500 valores más recientes*. Esto puede resultar útil para consultar los valores numéricos que componen el gráfico.

Los valores que se muestran aquí son sin procesar, es decir, no se utilizan unidades ni posprocesamiento de valores. Sin embargo, se aplica la asignación de valores.

Problemas conocidos

Consulte los problemas conocidos para los gráficos.

Gráficos personalizados

Descripción general

Los gráficos personalizados, como su nombre indica, ofrecen capacidades de personalización.

Si bien los gráficos simples son buenos para ver los datos de un solo elemento, no ofrecen capacidades de configuración.

Por lo tanto, si desea cambiar el estilo del gráfico o la forma en que se muestran las líneas o comparar varios elementos, por ejemplo, el tráfico entrante y saliente en un solo gráfico, necesita un gráfico personalizado.

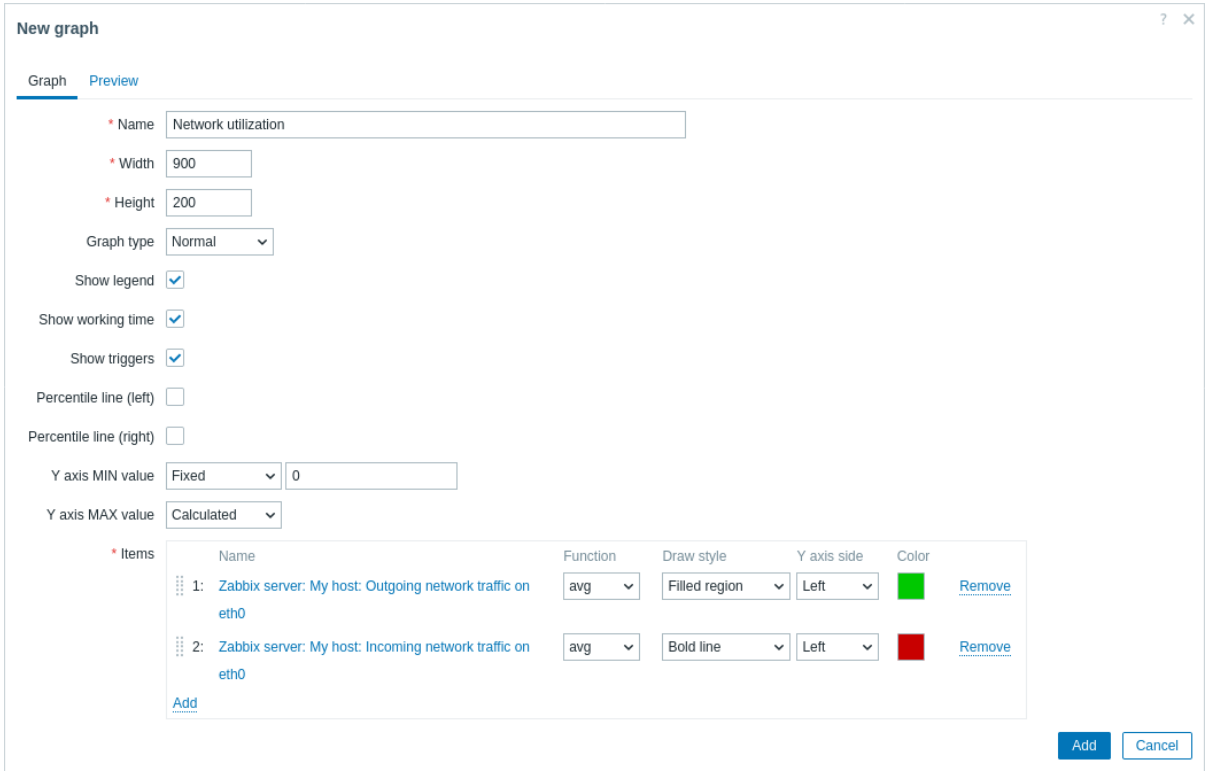
Los gráficos personalizados se configuran manualmente.

Pueden crearse para un equipo o varios equipos o para una sola plantilla.

Configuración de gráficos personalizados

Para crear un gráfico personalizado, haga lo siguiente:

1. Vaya a *Data collection* > *Hosts* (o *Templates*).
2. Haga clic en *Graphs* en la fila junto al host o template deseado.
3. En la pantalla de Graphs, haga clic en *Create graph*.
4. Edite los atributos del gráfico.



Todos los campos obligatorios están marcados con un asterisco rojo.

Atributos del gráfico:

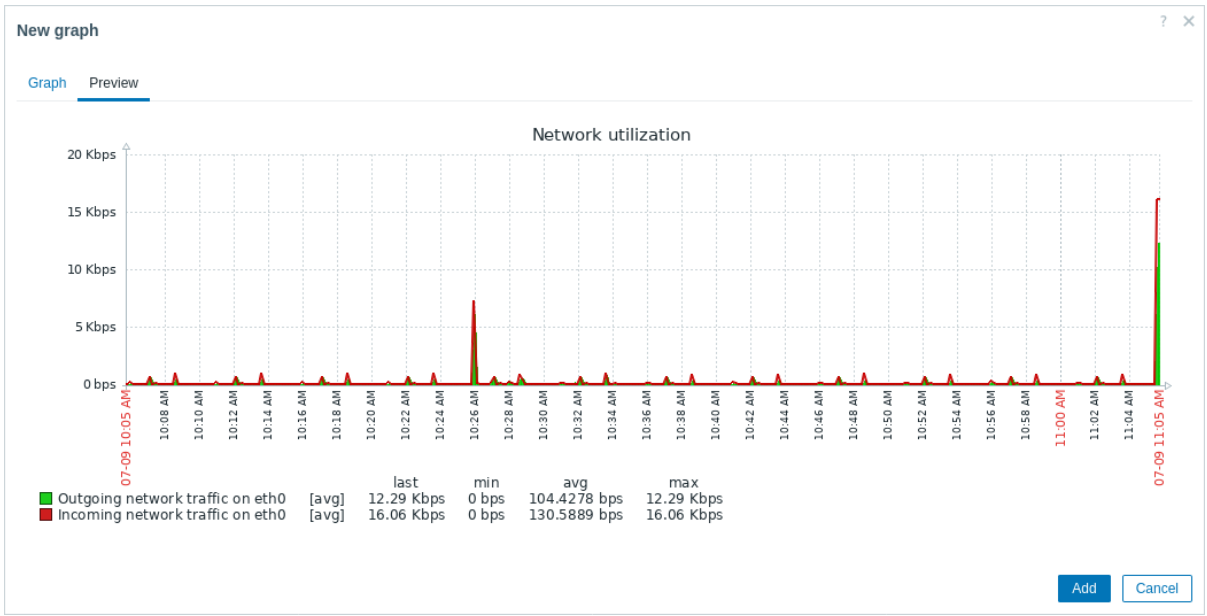
Parameter	Description
Name	Nombre único del gráfico. Se admiten expresiones macros en este campo, pero solo con las funciones `avg`, `last`, `min` y `max`, con el tiempo como parámetro (por ejemplo, `{?avg(/host/key,1h)}`). Se admiten macros {HOST.HOST<1-9>} para su uso dentro de esta macro, haciendo referencia al primer, segundo, tercer, etc. host del gráfico, por ejemplo, `{?avg(/{HOST.HOST2}/key,1h)}`. Tenga en cuenta que hacer referencia al primer host con esta macro es redundante, ya que el primer host puede referenciarse implícitamente, por ejemplo, `{?avg(//key,1h)}`.
Width	Ancho del gráfico en píxeles (solo para vista previa y gráficos de tipo pie/exploded).
Height	Altura del gráfico en píxeles.

Parameter	Description
Graph type	Tipo de gráfico: **Normal** - gráfico normal, los valores se muestran como líneas **Stacked** - gráfico apilado, se muestran áreas rellenas **Pie** - gráfico circular **Exploded** - gráfico circular ”desglosado”, las porciones se muestran como ”separadas” del gráfico circular
Show legend	Al marcar esta casilla se mostrará la leyenda del gráfico.
Show working time	Si se selecciona, las horas no laborables se mostrarán con un fondo gris. Este parámetro no está disponible para gráficos pie y exploded pie.
Show triggers	Si se selecciona, los simple triggers se mostrarán como líneas con guiones negros sobre el color de severidad del trigger. Este parámetro no está disponible para gráficos pie y exploded pie.
Percentile line (left)	Mostrar percentil para el eje Y izquierdo. Si, por ejemplo, se establece el percentil del 95%, la línea de percentil estará en el nivel por debajo del cual cae el 95 por ciento de los valores. Se muestra como una línea verde brillante. Solo disponible para gráficos normales.
Percentile line (right)	Mostrar percentil para el eje Y derecho. Si, por ejemplo, se establece el percentil del 95%, la línea de percentil estará en el nivel por debajo del cual cae el 95 por ciento de los valores. Se muestra como una línea roja brillante. Solo disponible para gráficos normales.
Y axis MIN value	Valor mínimo del eje Y: **Calculated** - el valor mínimo del eje Y se calculará automáticamente. **Fixed** - valor mínimo fijo para el eje Y. **Item** - el último valor del item seleccionado será el valor mínimo. Este parámetro no está disponible para gráficos pie y exploded pie.
Y axis MAX value	Valor máximo del eje Y: **Calculated** - el valor máximo del eje Y se calculará automáticamente. **Fixed** - valor máximo fijo para el eje Y. **Item** - el último valor del item seleccionado será el valor máximo Este parámetro no está disponible para gráficos pie y exploded pie.
3D view	Habilitar estilo 3D. Solo para gráficos pie y exploded pie.
Items	Items cuyos datos se mostrarán en este gráfico. Haga clic en *Add* para seleccionar items. También puede seleccionar varias opciones de visualización (función, estilo de dibujo, visualización en el eje izquierdo/derecho, color).
Sort order (0→100)	Orden de dibujo. El 0 se procesará primero. Puede usarse para dibujar líneas o regiones detrás de otra (o delante de otra). Puede arrastrar y soltar items usando el icono al principio de una línea para establecer el orden de clasificación o qué item se muestra delante del otro.
Name	El nombre del item seleccionado se muestra como un enlace. Al hacer clic en el enlace se abre la lista de otros items disponibles.
Type	Tipo (solo disponible para gráficos pie y exploded pie): **Simple** - el valor del item se representa proporcionalmente en el gráfico circular **Graph sum** - el valor del item representa el gráfico circular completo Tenga en cuenta que el color del item ”graph sum” solo será visible en la medida en que no sea ocupado por items ”proportional”.

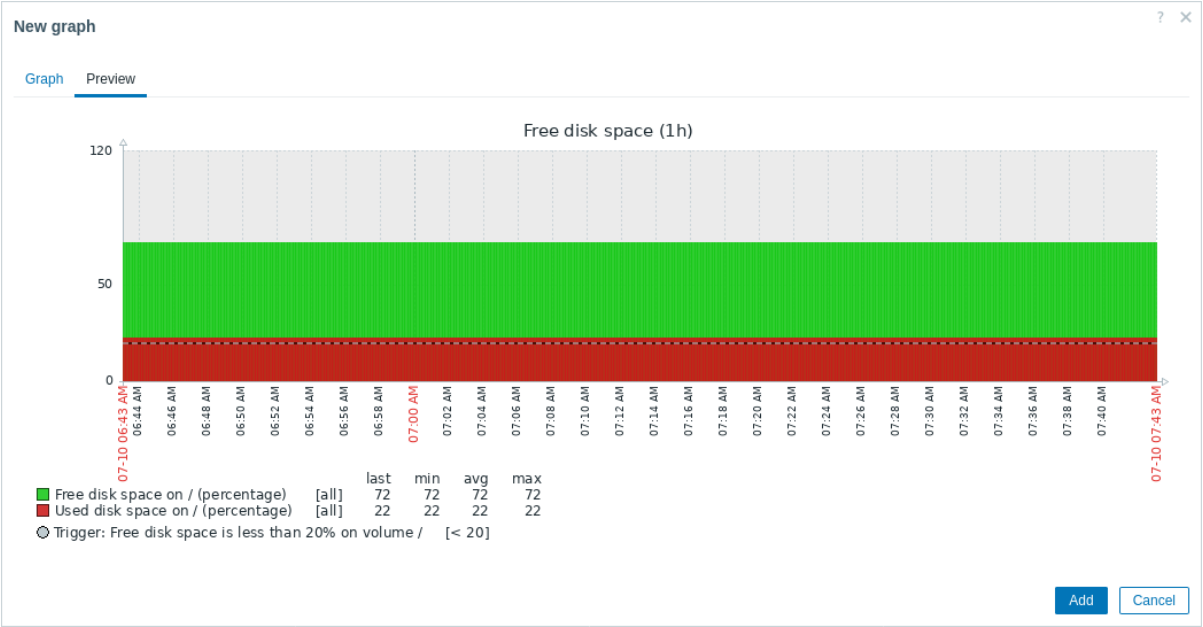
Parameter	Description
Function	Seleccione qué valores se mostrarán cuando exista más de un valor por píxel vertical del gráfico para un item: **all** - muestra todos los valores posibles (mínimo, máximo, promedio) en el gráfico. Tenga en cuenta que para períodos más cortos esta configuración no tiene efecto; solo para períodos más largos, cuando la congestión de datos en un píxel vertical del gráfico aumenta, 'all' comienza a mostrar los valores mínimo, máximo y promedio. Esta función solo está disponible para el tipo de gráfico *Normal*. Consulte también: Generación de gráficos from history/trends. **avg** - muestra los valores promedio **last** - muestra los valores más recientes. Esta función solo está disponible si se selecciona *Pie/Exploded pie* como tipo de gráfico. **max** - muestra los valores máximos **min** - muestra los valores mínimos
Draw style	Seleccione el estilo de dibujo (solo disponible para gráficos normales; para gráficos apilados siempre se usa una región rellena) que se aplicará a los datos del item - *Line*, *Bold line*, *Filled region*, *Dot*, *Dashed line*, *Gradient line*.
Y axis side	Seleccione el lado del eje Y para mostrar los datos del item - *Left*, *Right*.
Color	Seleccione el color que se aplicará a los datos del item.

Vista previa del gráfico

En la pestaña *Vista previa*, se muestra una vista previa del gráfico para que pueda ver inmediatamente lo que está creando.



Tenga en cuenta que la vista previa no mostrará ningún dato para los elementos de plantilla.



En este ejemplo, preste atención a la línea discontinua en negrita que muestra el nivel del disparador y la información del disparador que se muestra en la leyenda.

Note:
No se pueden mostrar más de 3 líneas de disparador. Si hay más disparadores, se priorizan para mostrar los disparadores con menor gravedad.

 Si la altura del gráfico se establece en menos de 120 píxeles, no se mostrará ningún disparador en la leyenda.

Gráficos ad hoc

Descripción general

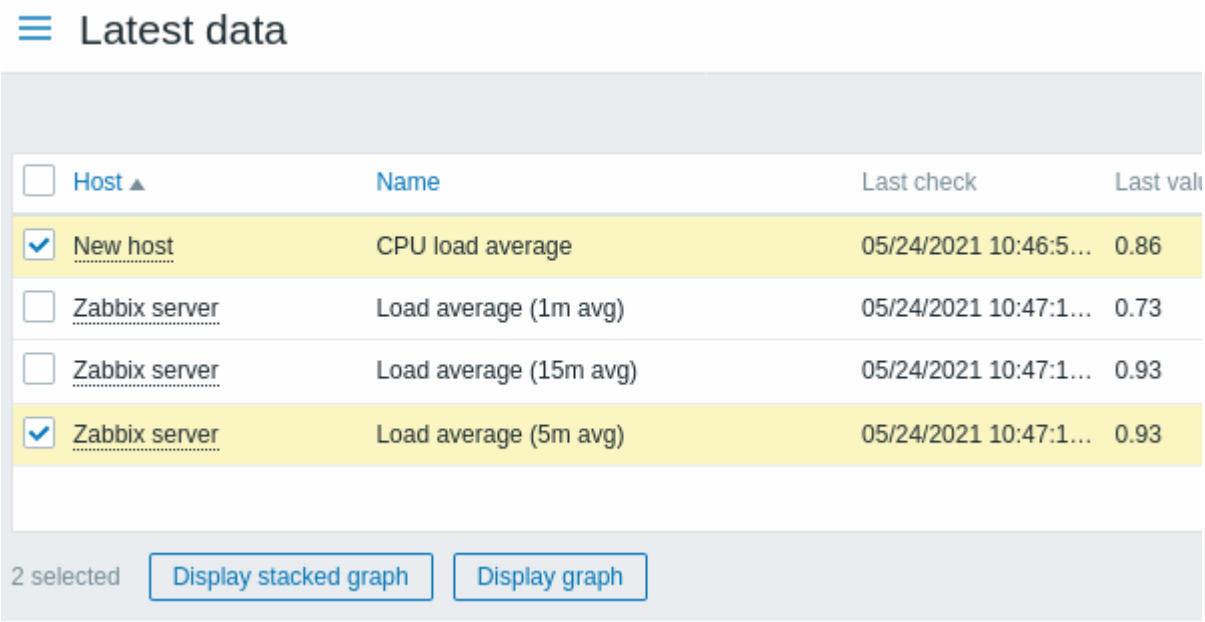
Mientras que un gráfico simple resulta útil para consultar los datos de un único item y los gráficos personalizados ofrecen opciones de personalización, ninguno de los dos permite crear rápidamente un gráfico comparativo para varios items con un esfuerzo y mantenimiento mínimos.

Para abordar este problema, es posible crear gráficos ad hoc para varios items de forma muy rápida.

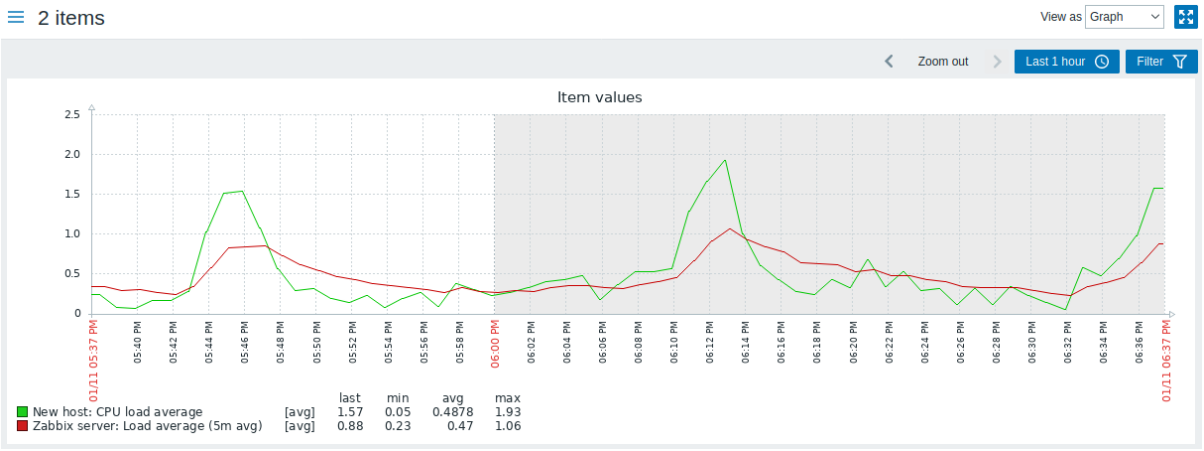
Configuración

Para crear un gráfico ad hoc, haga lo siguiente:

1. Vaya a *Monitoring* > *Latest data*.
2. Use el filtro para mostrar los items que desea.
3. Marque las casillas de verificación de los items que desea graficar.
4. Haga clic en los botones *Display stacked graph* o *Display graph*.

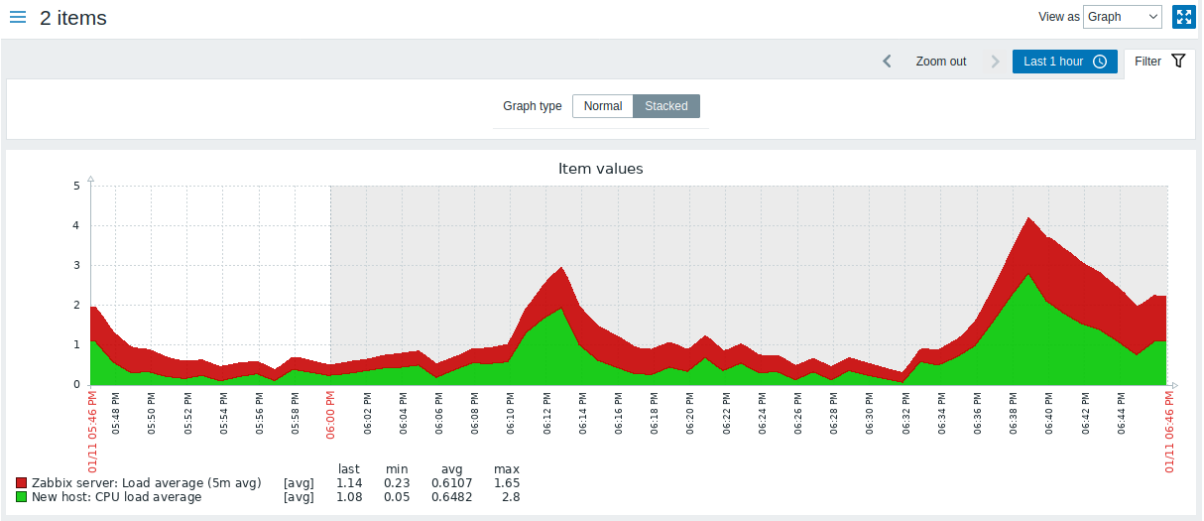


Su gráfico se crea al instante:



Tenga en cuenta que, para evitar mostrar demasiadas líneas en el gráfico, solo se muestra el valor promedio de cada item (no se muestran las líneas de valor mínimo/máximo). Los triggers y la información de los triggers no se muestran en el gráfico.

En la ventana del gráfico creado, dispone del selector de período de tiempo y de la posibilidad de cambiar del gráfico de líneas "normal" a uno apilado (y viceversa).



Mapas de red

Resumen

Si tiene una red que administrar, es posible que desee tener una visión general de su infraestructura en algún lugar. Para ese propósito, puede crear mapas en Zabbix, tanto de redes como de cualquier cosa que desee.

Todos los usuarios pueden crear mapas de red. Los mapas pueden ser públicos (disponibles para todos los usuarios) o privados (disponibles para usuarios seleccionados).

Proceda a configurar un mapa de red.

Configuración de un mapa de red

Resumen

Configurar un map en Zabbix requiere que primero cree un map definiendo sus parámetros generales y, a continuación, empiece a rellenar el map real con elementos y sus enlaces.

Puede poblar el map con elementos que sean un host, un grupo de hosts, un trigger, una imagen o otro map.

Los iconos se utilizan para representar los elementos del map. Puede definir la información que se mostrará con los iconos y establecer que los problemas recientes se muestren de una forma especial. Puede enlazar los iconos y definir la información que se mostrará en los enlaces.

Puede añadir URL personalizadas para que se pueda acceder a ellas al hacer clic en los iconos. De este modo, puede vincular un icono de host a las propiedades del host o un icono de map a otro map.

El recuento de problemas en los maps se muestra solo para los problemas causales.

Los maps se administran en *Monitoring* > *Maps*, donde se pueden configurar, administrar y ver. En la vista de monitorización, puede hacer clic en los iconos y aprovechar los enlaces a algunos scripts y URL.

Los mapas de red se basan en gráficos vectoriales (SVG).

Mapas públicos y privados

Todos los usuarios en Zabbix (incluidos los usuarios no administradores) pueden crear mapas de red. Los mapas tienen un propietario: el usuario que los creó. Los mapas pueden hacerse públicos o privados.

- Los mapas *públicos* son visibles para todos los usuarios, aunque para verlos, el usuario debe tener acceso de solo lectura a al menos un elemento del mapa. Los mapas públicos pueden editarse en caso de que un usuario/grupo de usuarios tenga permisos de lectura-escritura para este mapa y al menos permisos de solo lectura para todos los elementos del mapa correspondiente, incluidos los triggers en los enlaces.
- Los mapas *privados* son visibles solo para su propietario y los usuarios/grupos de usuarios con los que el propietario comparta el mapa. Los usuarios normales (no Super administradores) solo pueden compartir con los grupos a los que pertenecen y con los usuarios que pertenecen a esos grupos. Los usuarios con nivel de administrador pueden ver los mapas privados independientemente de ser el propietario o de pertenecer a la lista de usuarios compartidos. Los mapas privados pueden ser editados por el propietario del mapa y en caso de que un usuario/grupo de usuarios tenga permisos de lectura-escritura para este mapa y al menos permisos de solo lectura para todos los elementos del mapa correspondiente, incluidos los triggers en los enlaces.

Los elementos del mapa para los que el usuario no tiene permiso de solo lectura se muestran con un icono atenuado y toda la información textual sobre el elemento está oculta. Sin embargo, la etiqueta del trigger es visible incluso si el usuario no tiene permiso para el trigger.

Para agregar un elemento al mapa, el usuario también debe tener al menos permiso de solo lectura para el elemento.

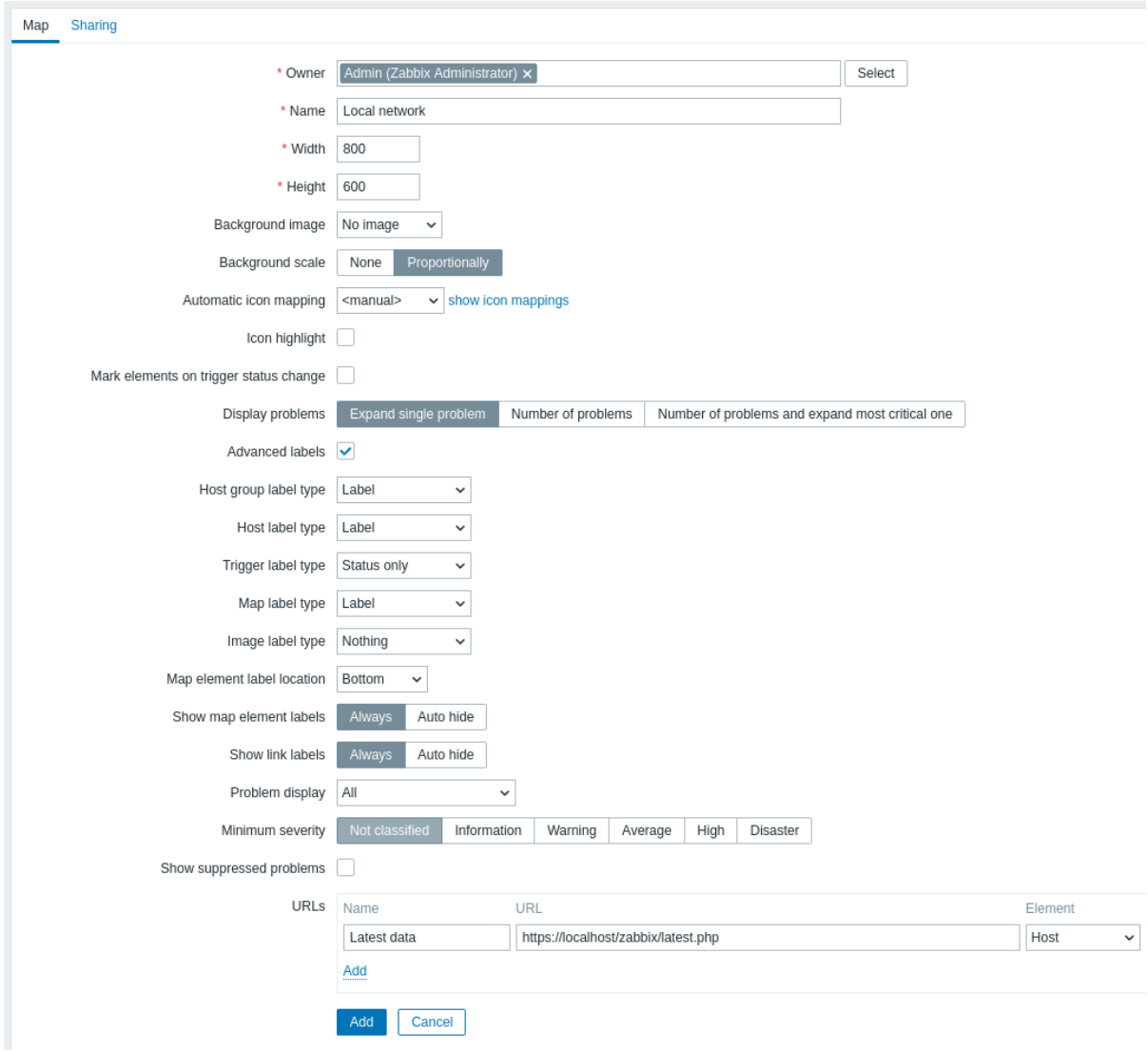
Creación de un mapa

Para crear un mapa, haga lo siguiente:

1. Vaya a *Monitoring* > *Maps*.
2. Vaya a la vista con todos los mapas.
3. Haga clic en *Create map*.
4. Configure los detalles del mapa.

También puede utilizar el botón *Clone* en el formulario de configuración de un mapa existente para crear un mapa nuevo. Este mapa tendrá todas las propiedades del mapa existente, incluidos los atributos generales de diseño, así como los elementos del mapa existente.

La pestaña **Map** contiene los atributos generales del mapa:



Todos los campos de entrada obligatorios están marcados con un asterisco rojo.

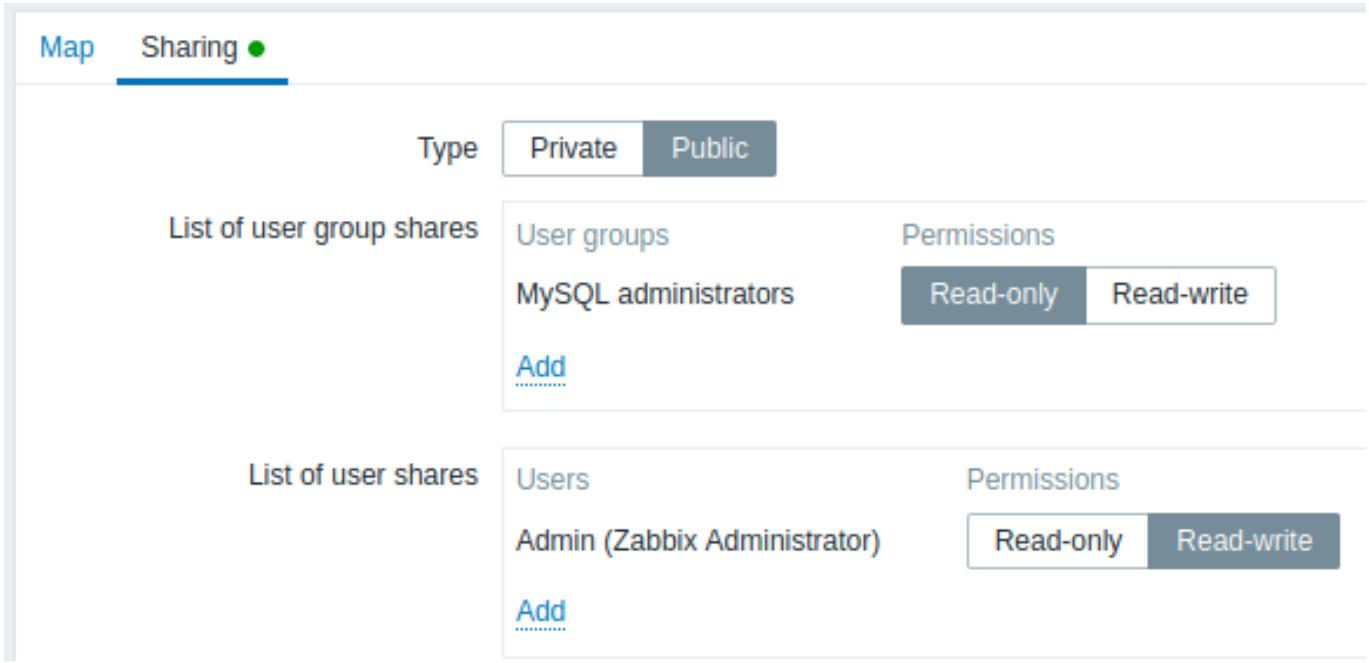
Atributos generales del mapa:

Parámetro	Descripción
Owner	Nombre del propietario del mapa.
Name	Nombre único del mapa.
Width	Ancho del mapa en píxeles.
Height	Alto del mapa en píxeles.
Background image	Utilizar una imagen de fondo: **No image**: no se utiliza ninguna imagen de fondo (fondo blanco) **Image**: imagen seleccionada que se utilizará como imagen de fondo. Puede utilizar un mapa geográfico o cualquier otra imagen para mejorar el mapa.
Background scale	Utilizar el escalado del fondo: **None**: sin escalado; **Proportionally**: cubrir todo el fondo del mapa, manteniendo las proporciones de la imagen (opción predeterminada).
Automatic icon mapping	Puede configurar el uso de una asignación automática de iconos, configurada en *Administration > General > Icon mapping*. La asignación de iconos permite asociar determinados iconos con determinados campos del inventario del host.
Icon highlight	Si marca esta casilla, los elementos del mapa se resaltarán. Los elementos con un trigger activo recibirán un fondo circular del mismo color que el trigger de mayor severidad. Además, se mostrará una línea verde gruesa alrededor del círculo si todos los problemas están reconocidos. Los elementos con estado ”disabled” o ”in maintenance” tendrán un fondo cuadrado, gris y naranja respectivamente. Consulte también: Visualización de mapas

Parámetro	Descripción
Mark elements on trigger status change	Un cambio reciente del estado del trigger (problema reciente o resolución) se resaltará con marcadores (triángulos rojos orientados hacia dentro) en los tres lados del icono del elemento que estén libres de la etiqueta. Los marcadores se muestran durante 30 minutos.
Display problems	Seleccione cómo se muestran los problemas con un elemento del mapa: **Expand single problem**: si solo hay un problema, se muestra el nombre del problema. De lo contrario, se muestra el número total de problemas. **Number of problems**: se muestra el número total de problemas. **Number of problems and expand most critical one**: se muestran el nombre del problema más crítico y el número total de problemas. El problema ”más crítico” se determina según la severidad del problema y, si es igual, según el ID del evento del problema (el ID más alto o el problema más reciente se muestra primero). Para un *elemento de mapa de trigger*, se determina según la severidad del problema y, si es igual, según la posición del trigger en la lista de triggers. En caso de varios problemas del mismo trigger, se mostrará el más reciente.
Advanced labels	Si marca esta casilla, podrá definir tipos de etiqueta independientes para distintos tipos de elementos.
Host group label type	Tipo de etiqueta utilizado para el elemento del mapa: **Label**: etiqueta del elemento del mapa **IP address**: dirección IP **Element name**: nombre del elemento (por ejemplo, nombre del host) **Status only**: solo el estado (OK o PROBLEM) **Nothing**: no se muestran etiquetas
Host label type *Trigger label type* *Map label type* *Image label type*	
Map element label location	Ubicación de la etiqueta en relación con el elemento del mapa: **Bottom**: debajo del elemento del mapa **Left**: a la izquierda **Right**: a la derecha **Top**: encima del elemento del mapa
Show map element labels	Seleccione cómo mostrar las etiquetas de los elementos del mapa: **Always**: siempre se muestran (opción predeterminada); **Auto hide**: ocultar la etiqueta si no se coloca el cursor sobre ella ni se selecciona.
Show link labels	Seleccione cómo mostrar las etiquetas de los enlaces: **Always**: siempre se muestran (opción predeterminada); **Auto hide**: ocultar la etiqueta si no se coloca el cursor sobre ella ni se selecciona.
Problem display	Mostrar el recuento de problemas como: **All**: se muestra el recuento total de problemas **Separated**: el recuento de problemas no reconocidos se muestra por separado como parte del recuento total de problemas **Unacknowledged only**: solo se muestra el recuento de problemas no reconocidos
Minimum severity	Los problemas por debajo del nivel mínimo de severidad seleccionado no se mostrarán en el mapa. Por ejemplo, si selecciona *Warning*, los cambios con triggers de nivel *Information* y *Not classified* no se reflejarán en el mapa.
Show suppressed problems	Marque la casilla para mostrar los problemas que, de otro modo, se suprimirían (no se mostrarían) debido al mantenimiento del host.
URLs	Aquí se pueden definir las URL para cada tipo de elemento, donde: **Name**: una etiqueta para la URL; **URL**: una URL (de hasta 2048 caracteres); **Element**: el tipo de elemento (por ejemplo, *Host*, *Host group*, etc.). Se mostrarán como enlaces cuando un usuario haga clic en el elemento en el modo de visualización del mapa. Se pueden utilizar macros en los nombres y valores de las URL del mapa. Para obtener la lista completa, consulte macros compatibles y busque ’map URL names and values’.

Compartir

La pestaña **Sharing** contiene el tipo de mapa, así como las opciones de compartición (grupos de usuarios, usuarios) para mapas privados:



Parameter	Description
Type	Seleccione el tipo de mapa: **Private** - el mapa es visible solo para los grupos de usuarios y usuarios seleccionados **Public** - el mapa es visible para todos
List of user group shares	Seleccione los grupos de usuarios a los que el mapa es accesible. Puede permitir acceso de solo lectura o de lectura y escritura.
List of user shares	Seleccione los usuarios a los que el mapa es accesible. Puede permitir acceso de solo lectura o de lectura y escritura.

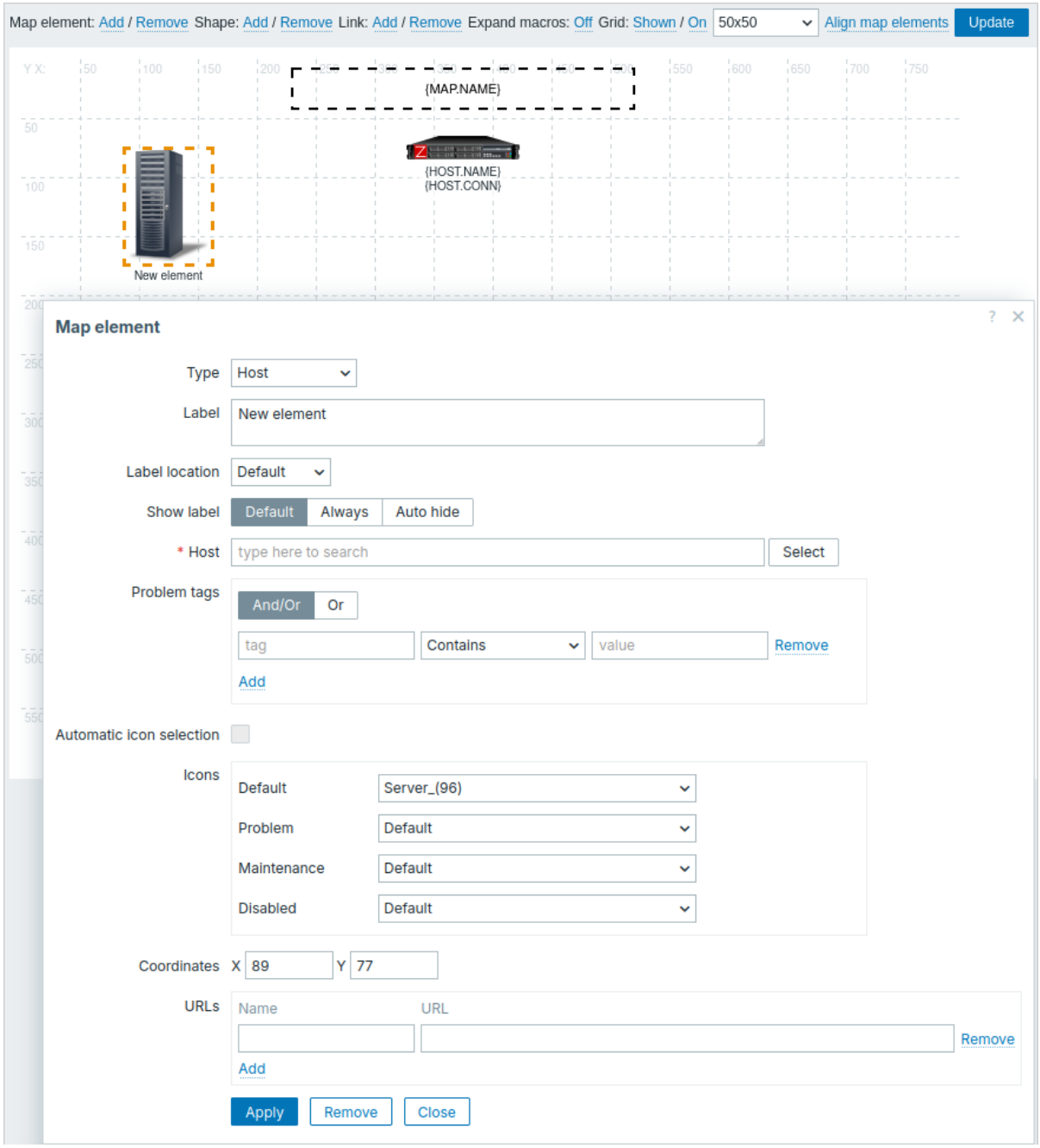
Cuando haga clic en *Add* para guardar este mapa, habrá creado un mapa vacío con un nombre, dimensiones y ciertas preferencias. Ahora necesita añadir algunos elementos. Para ello, haga clic en *Edit* en la lista de mapas para abrir el área editable.

Adición de elementos

Para añadir un elemento, haga clic en *Add* junto a `Map element`. El nuevo elemento aparecerá en la esquina superior izquierda del mapa. Arrástrelo y suéltelo donde quiera.

Tenga en cuenta que, con la opción Grid en "On", los elementos siempre se alinearán con la cuadrícula (puede elegir varios tamaños de cuadrícula en la lista desplegable, y también ocultar/mostrar la cuadrícula). Si desea colocar elementos en cualquier lugar sin alineación, cambie la opción a "Off". (Puede alinear elementos aleatorios con la cuadrícula más tarde, haciendo clic en *Align map elements*.)

Ahora que ya tiene algunos elementos colocados, quizá quiera empezar a diferenciarlos asignándoles nombres, etc. Al hacer clic en el elemento, se muestra un formulario en el que puede establecer el tipo de elemento, asignar un nombre, elegir un icono diferente, etc.



Atributos del elemento del mapa:

Parameter	Description
Type	Tipo del elemento: **Host** - icono que representa el estado de todos los triggers del host seleccionado **Map** - icono que representa el estado de todos los elementos de un mapa **Trigger** - icono que representa el estado de uno o más triggers **Host group** - icono que representa el estado de todos los triggers de todos los hosts pertenecientes al grupo seleccionado **Image** - un icono, no vinculado a ningún recurso
Label	Etiqueta del icono, cualquier cadena. Se pueden usar macros y cadenas multilínea. En este campo se admiten macros de expresión macros, pero solo con las funciones `avg`, `last`, `min` y `max`, con tiempo como parámetro (por ejemplo, `{?avg(/host/key,1h)}`). Para ver la lista completa de macros admitidas, consulte supported macros y busque 'map element labels'.

Parameter	Description
Label location	Ubicación de la etiqueta en relación con el icono: **Default** - ubicación predeterminada de la etiqueta del mapa; **Bottom** - debajo del icono; **Left** - a la izquierda; **Right** - a la derecha; **Top** - encima del icono.
Show label	Seleccione cómo mostrar la etiqueta del elemento: **Default** - usar la configuración *Show map element labels* de la configuración del mapa (predeterminado); **Always** - siempre mostrada (predeterminado); **Auto hide** - ocultar la etiqueta si no se pasa el cursor sobre ella o no está seleccionada.
Host	Seleccione un host; alternativamente, introduzca el nombre del host. Este campo admite autocompletado, por lo que al empezar a escribir el nombre de un host se mostrará una lista desplegable con los hosts coincidentes. Desplácese hacia abajo para seleccionar uno. Haga clic en el icono de eliminar (x) para quitar la selección. Este campo solo está disponible para elementos de tipo *Host*.
Map	Seleccione un mapa; alternativamente, introduzca el nombre del mapa. Este campo admite autocompletado, por lo que al empezar a escribir el nombre de un mapa se mostrará una lista desplegable con los mapas coincidentes. Desplácese hacia abajo para seleccionar uno. Haga clic en el icono de eliminar (x) para quitar la selección. Este campo solo está disponible para elementos de tipo *Map*.
Triggers	Seleccione uno o más triggers en el campo *New triggers* de abajo y haga clic en *Add*.Alternativamente, empiece a escribir el nombre de un trigger y seleccione uno de la lista desplegable de triggers coincidentes. Haga clic en el icono de eliminar (x) para quitar la selección. Este campo solo está disponible para elementos de tipo *Trigger*. El orden de los triggers seleccionados puede cambiarse, pero solo dentro de la misma severidad de triggers. La selección múltiple de triggers también afecta a la resolución de macros {HOST.*} tanto en el modo de edición como en el de visualización. En el modo de edición, las primeras macros {HOST.*} mostradas se resolverán en función del primer trigger de la lista (según la severidad del trigger). El modo de visualización depende del parámetro Display problems en los atributos generales del mapa: - Si se elige el modo *Expand single problem*, las primeras macros {HOST.*} mostradas se resolverán en función del último trigger de problema detectado (sin importar la severidad) o del primer trigger de la lista (en caso de que no se detecte ningún problema). - Si se elige el modo *Number of problems and expand most critical one*, las primeras macros {HOST.*} mostradas se resolverán en función de la severidad del trigger.
Host group	Seleccione un host group; alternativamente, introduzca el host group. Este campo admite autocompletado, por lo que al empezar a escribir el nombre de un host group se mostrará una lista desplegable con los host groups coincidentes. Desplácese hacia abajo para seleccionar uno. Haga clic en el icono de eliminar (x) para quitar la selección. Este campo solo está disponible para elementos de tipo *Host group*.

Parameter	Description
Problem tags	Especifique etiquetas para limitar el número de problemas mostrados en el widget. Es posible incluir y excluir etiquetas y valores de etiqueta específicos. Se pueden establecer varias condiciones. La coincidencia del nombre de la etiqueta siempre distingue entre mayúsculas y minúsculas. Hay varios operadores disponibles para cada condición: **Exists** - incluir los nombres de etiqueta especificados **Equals** - incluir los nombres y valores de etiqueta especificados (distingue mayúsculas y minúsculas) **Contains** - incluir los nombres de etiqueta especificados cuyos valores contengan la cadena introducida (coincidencia de subcadena, sin distinguir mayúsculas y minúsculas) **Does not exist** - excluir los nombres de etiqueta especificados **Does not equal** - excluir los nombres y valores de etiqueta especificados (distingue mayúsculas y minúsculas) **Does not contain** - excluir los nombres de etiqueta especificados cuyos valores contengan la cadena introducida (coincidencia de subcadena, sin distinguir mayúsculas y minúsculas) Hay dos tipos de cálculo para las condiciones: **And/Or** - deben cumplirse todas las condiciones; las condiciones que tengan el mismo nombre de etiqueta se agruparán mediante la condición Or **Or** - basta con que se cumpla una condición Este campo solo está disponible para elementos de tipo *Host* y *Host group*.
Automatic icon selection	Cuando la casilla está marcada, se utilizará la asignación de iconos para determinar qué icono mostrar.
Icons	Puede elegir mostrar distintos iconos para el elemento en estos casos: *Default*, *Problem*, *Maintenance* y *Disabled*. Este campo solo está disponible para elementos de tipo *Host*.
Coordinates	Coordenadas **X** e **Y** del elemento del mapa.
URLs	Aquí se pueden definir URLs específicas del elemento, donde: **Name** - una etiqueta para la URL; **URL** - una URL (hasta 2048 caracteres). Estas se mostrarán como enlaces cuando un usuario haga clic en el elemento en el modo de visualización del mapa. Si el elemento tiene sus propias URLs y existen URLs a nivel de mapa definidas para su tipo, se combinarán en el mismo menú. Se pueden usar macros en los nombres y valores de los elementos del mapa. Para ver la lista completa, consulte supported macros y busque 'map URL names and values'.

Attention:
Los elementos añadidos no se guardan automáticamente. Si sale de la página, todos los cambios pueden perderse.
Por lo tanto, es recomendable hacer clic en el botón **Update** en la esquina superior derecha. Una vez hecho clic, los cambios se guardan independientemente de lo que elija en la ventana emergente siguiente.
Las opciones de cuadrícula seleccionadas también se guardan con cada mapa.

Selección de elementos

Para seleccionar elementos, seleccione uno y luego mantenga presionada la tecla *Ctrl* para seleccionar los demás.

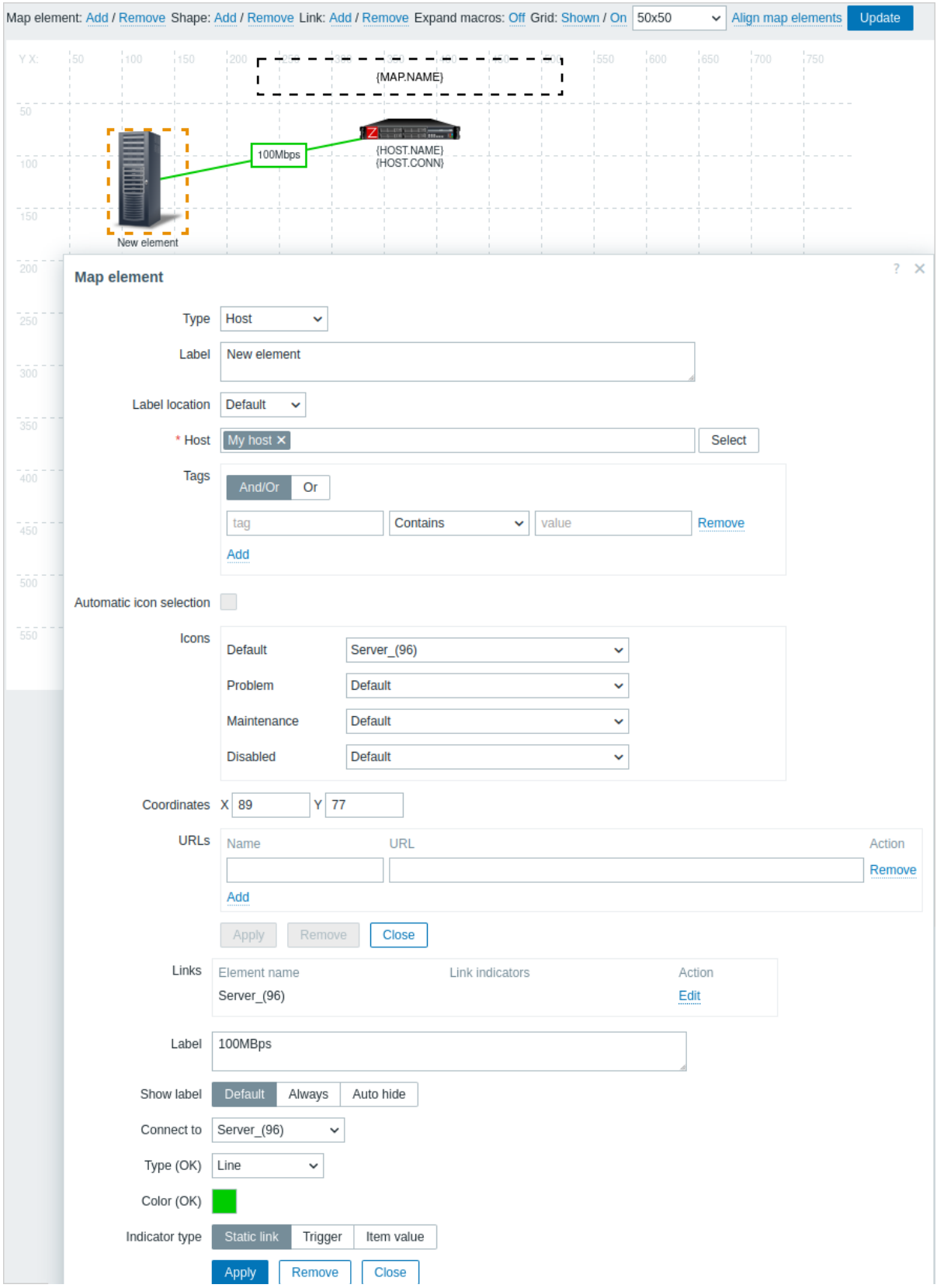
También puede seleccionar varios elementos arrastrando un rectángulo en el área editable y seleccionando todos los elementos dentro de él.

Una vez que seleccione más de un elemento, el formulario de propiedades del elemento cambia al modo de actualización masiva para que pueda cambiar los atributos de los elementos seleccionados de una sola vez. Para hacerlo, marque el atributo usando la casilla de verificación e ingrese un nuevo valor para él. Puede usar macros aquí (por ejemplo, {HOST.NAME} para la etiqueta del elemento).

Vinculación de elementos

Una vez que haya colocado algunos elementos en el mapa, es momento de empezar a vincularlos. Para vincular dos elementos, primero debe seleccionarlos. Con los elementos seleccionados, haga clic en *Add* junto a Link.

Una vez creado un enlace, el formulario de elemento individual ahora contiene una sección adicional *Links*. Haga clic en *Edit* para editar los atributos del enlace.



Atributos del enlace:

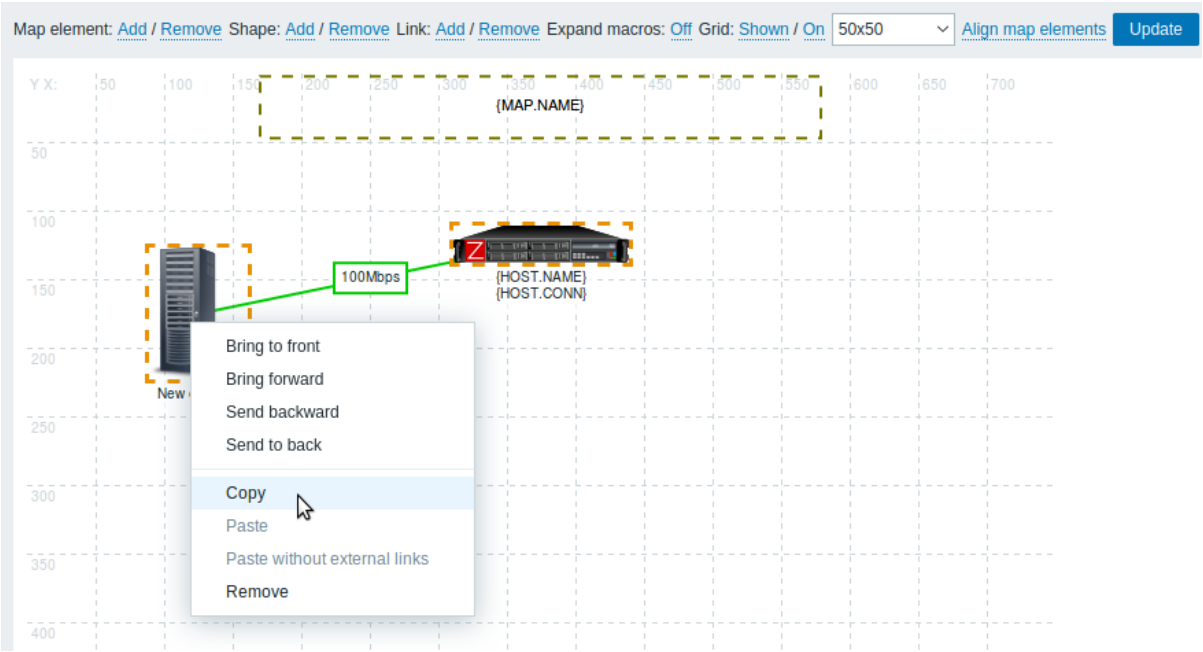
Parameter	Description
Label	Etiqueta que se mostrará sobre el enlace. Se admiten expresiones macros en este campo, pero solo con funciones `avg`, `last`, `min` y `max`, con tiempo como parámetro (por ejemplo, `{?avg(/host/key,1h)}`).

Parameter	Description
Show label	Seleccione cómo mostrar la etiqueta del enlace: **Default** - usa la configuración *Show map element labels* de la configuración del mapa (predeterminado); **Always** - se muestra siempre (predeterminado); **Auto hide** - oculta la etiqueta si no está bajo el cursor o seleccionada.
Connect to	El elemento al que se conecta el enlace.
Type (OK)	Estilo predeterminado del enlace. Seleccione el estilo: *Line*, *Bold line*, *Dot* o *Dashed line*.
Color (OK)	Color predeterminado del enlace. Haga clic en la muestra de color para seleccionar otro color.
Indicator type	Seleccione el tipo de indicador del enlace: **Static link** - sin indicadores; **Trigger** - permite indicadores de enlace basados en trigger; **Item value** - permite indicadores de enlace basados en el valor de item.
Item	Seleccione el item. El valor de este item afectará al estilo del enlace. Este campo está disponible cuando se selecciona *Item value* para *Indicator type*.
Indicators	La lista de indicadores de enlace. Este campo está disponible cuando *Indicator type* está configurado como *Trigger* o *Item value*. Si se configura como *Trigger*, se pueden asignar triggers como indicadores de enlace. Cuando un trigger está en estado *Problem*, se aplicará al enlace su estilo (color y tipo de línea seleccionados). Si se configura como *Item value* y se selecciona un item en el campo *Item*, se pueden asignar valores de item como indicadores de enlace. Cuando el valor del item alcanza el umbral especificado (para un item numérico) o coincide con el patrón (para un item de texto), se aplica al enlace su estilo (color y tipo de línea seleccionados).

Mover y copiar y pegar elementos

Varios elementos seleccionados se pueden **mover** a otro lugar en el mapa haciendo clic en uno de los elementos seleccionados, manteniendo pulsado el botón del ratón y moviendo el cursor a la ubicación deseada.

Uno o más elementos se pueden **copiar** seleccionando los elementos y, a continuación, haciendo clic con el botón derecho del ratón en un elemento seleccionado y eligiendo *Copy* en el menú.

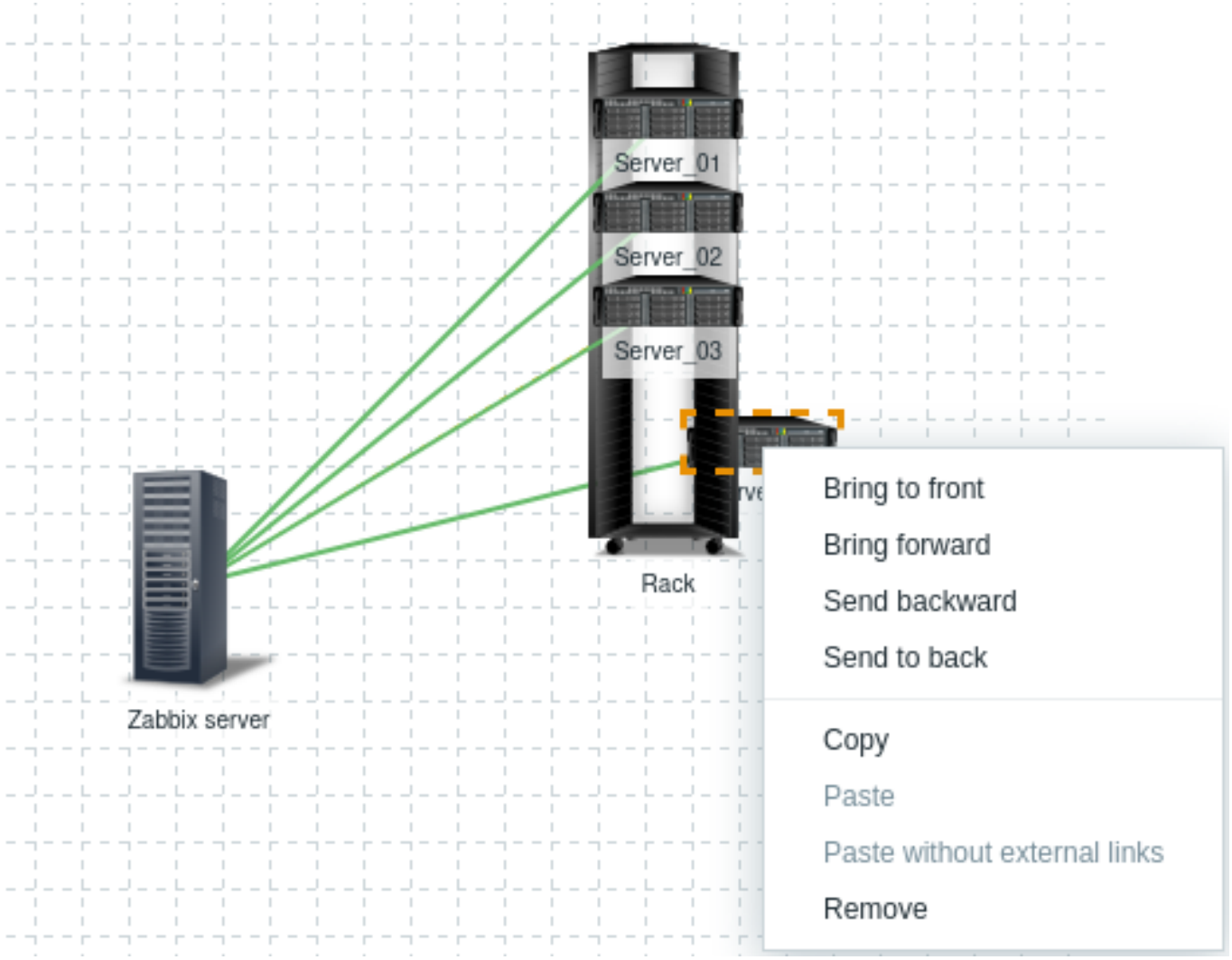


Para pegar los elementos, haga clic con el botón derecho del ratón en un área del mapa y seleccione *Paste* en el menú. La opción *Paste without external links* pegará los elementos conservando solo los enlaces que existen entre los elementos seleccionados.

La función de copiar y pegar funciona dentro de la misma ventana del navegador. No se admiten atajos de teclado.

Ordenación de elementos

Para colocar un elemento delante de otro (o viceversa), haga clic con el botón derecho del ratón en el elemento y seleccione *Bring forward*/*Bring to front* o *Send backward*/*Send to back*.

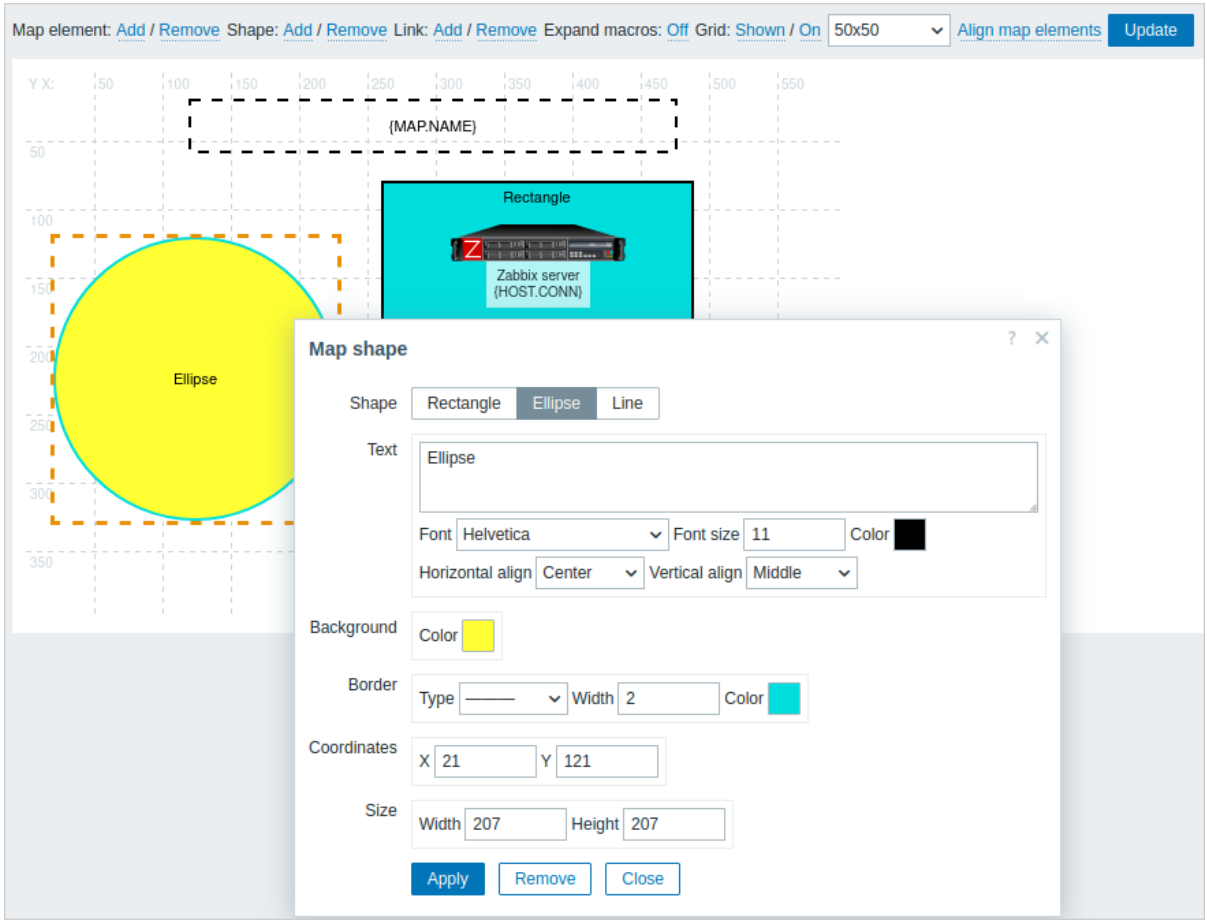


Añadir formas

Además de los elementos del mapa, también es posible añadir algunas formas. Las formas no son elementos del mapa; son solo una representación visual. Por ejemplo, una forma de rectángulo puede usarse como fondo para agrupar algunos hosts. Se pueden añadir formas de rectángulo y elipse.

Para añadir una forma, haga clic en *Add* junto a Shape. La nueva forma aparecerá en la esquina superior izquierda del mapa. Arrástrela y suéltela donde quiera.

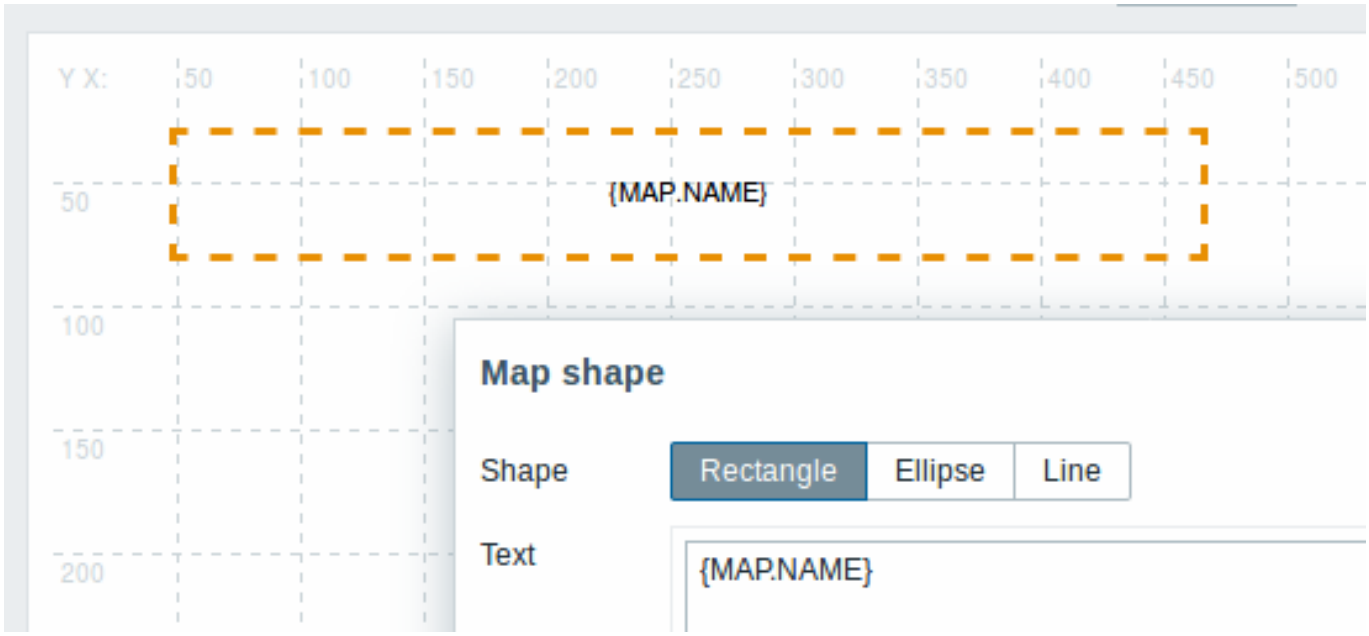
Se añade una nueva forma con los colores predeterminados. Al hacer clic en la forma, se muestra un formulario y puede personalizar el aspecto de la forma, añadir texto, etc.



Para seleccionar formas, seleccione una y luego mantenga pulsado *Ctrl* para seleccionar las demás. Con varias formas seleccionadas, las propiedades comunes se pueden actualizar en masa, de forma similar a como ocurre con los elementos.

Se puede añadir texto en las formas. Las macros de expresión son compatibles en el texto, pero solo con las funciones `avg`, `last`, `min` y `max`, con tiempo como parámetro (por ejemplo, `{?avg(/host/key,1h)}`).

Para mostrar solo texto, la forma puede hacerse invisible eliminando el borde de la forma (seleccione 'None' en el campo *Border*). Por ejemplo, observe cómo la macro {MAP.NAME}, visible en la captura de pantalla anterior, es en realidad una forma de rectángulo con texto, que puede verse al hacer clic en la macro:



{MAP.NAME} se resuelve en el nombre del mapa configurado al visualizar el mapa.

Si se usan hipervínculos en el texto, se vuelven clicables al visualizar el mapa.

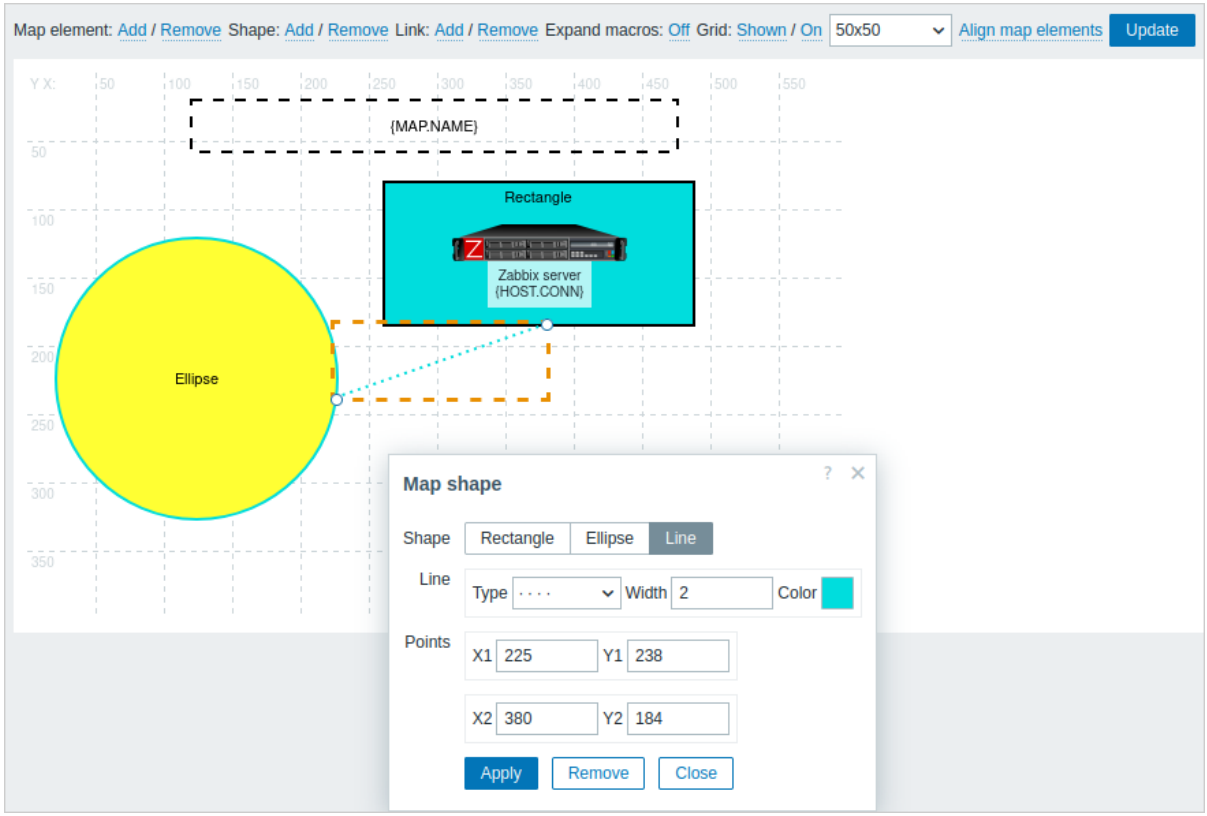
El ajuste de línea del texto siempre está "activado" dentro de las formas. Sin embargo, dentro de una elipse, las líneas se ajustan como si la elipse fuera un rectángulo. No se implementa el ajuste por palabras, por lo que las palabras largas (palabras que no

caben en la forma) no se ajustan, sino que se enmascaran (en la página de edición del mapa) o se recortan (en otras páginas con mapas).

Añadir líneas

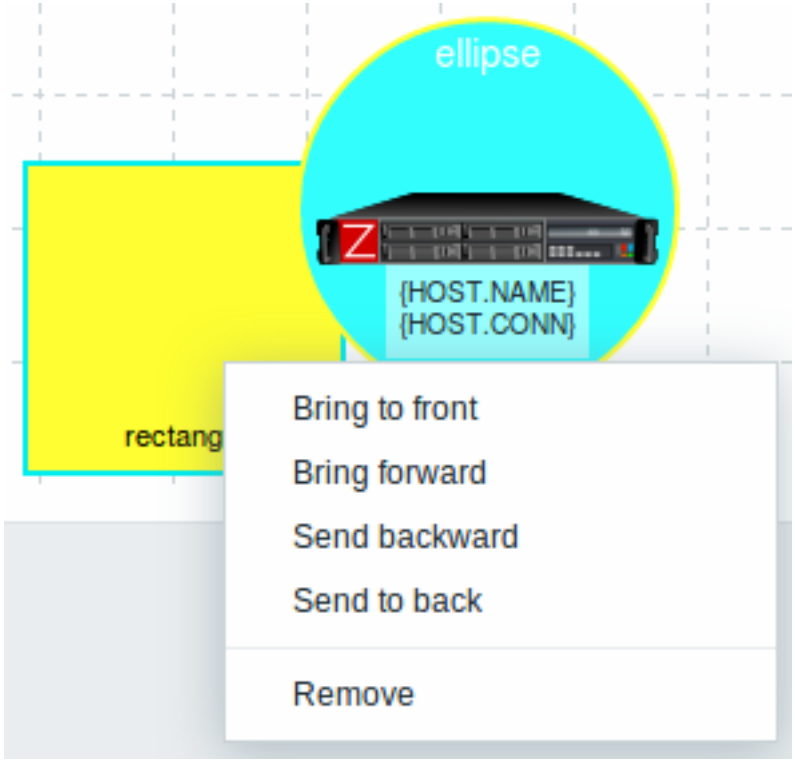
Además de formas, también es posible añadir algunas líneas. Las líneas se pueden usar para vincular elementos o formas en un mapa.

Para añadir una línea, haga clic en *Add* junto a Shape. Aparecerá una nueva forma en la esquina superior izquierda del mapa. Selecciónela y haga clic en *Line* en el formulario de edición para convertir la forma en una línea. Después, ajuste las propiedades de la línea, como el tipo de línea, el ancho, el color, etc.



Ordenación de formas y líneas

Para colocar una forma delante de otra (o viceversa), haga clic con el botón derecho del ratón sobre la forma para abrir el menú de formas del mapa.

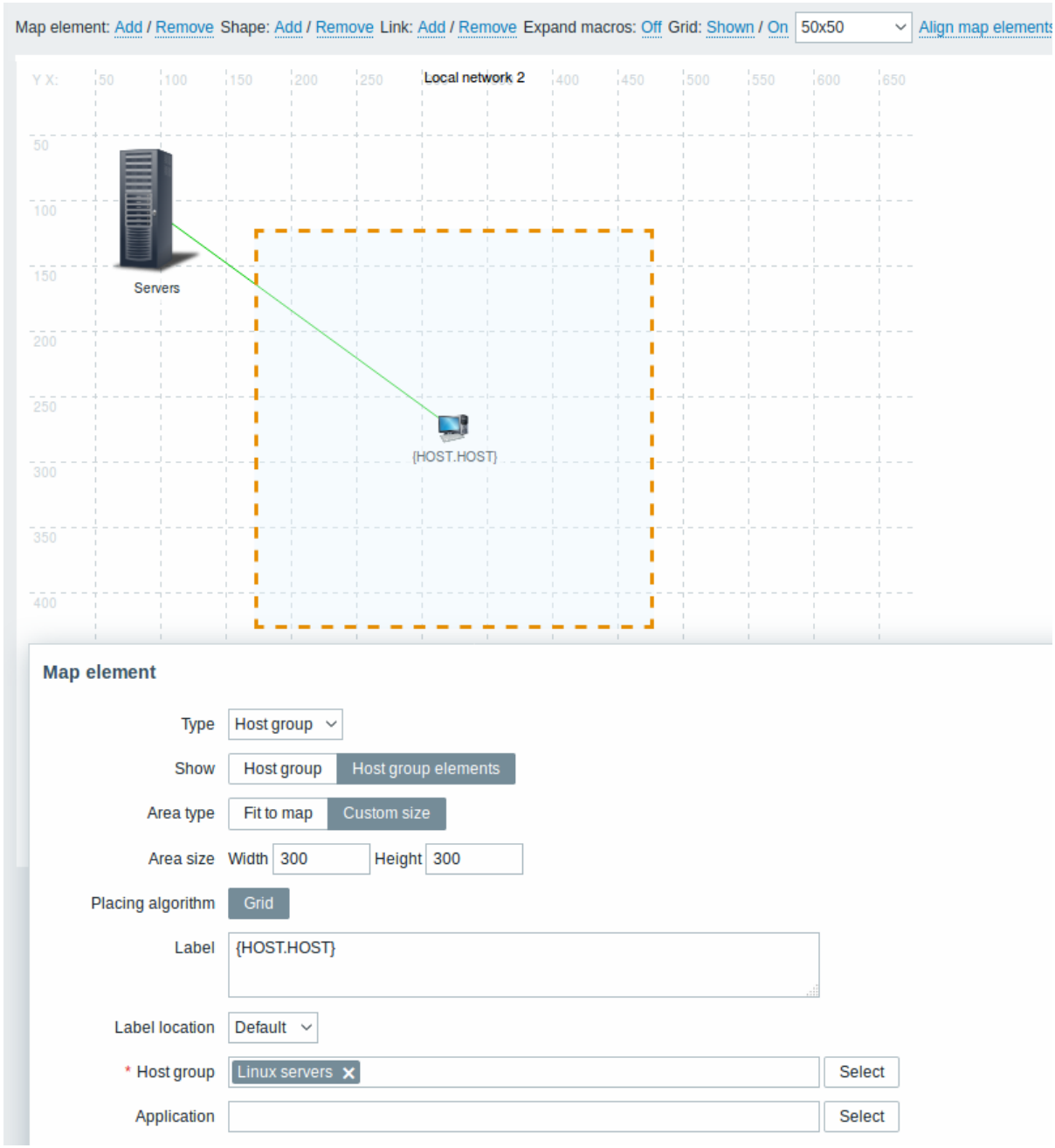


Elementos del grupo de hosts

Descripción general

Esta sección explica cómo añadir un elemento de tipo *Grupo de hosts* al configurar un mapa de red.

Configuración



Todos los campos de entrada obligatorios están marcados con un asterisco rojo.

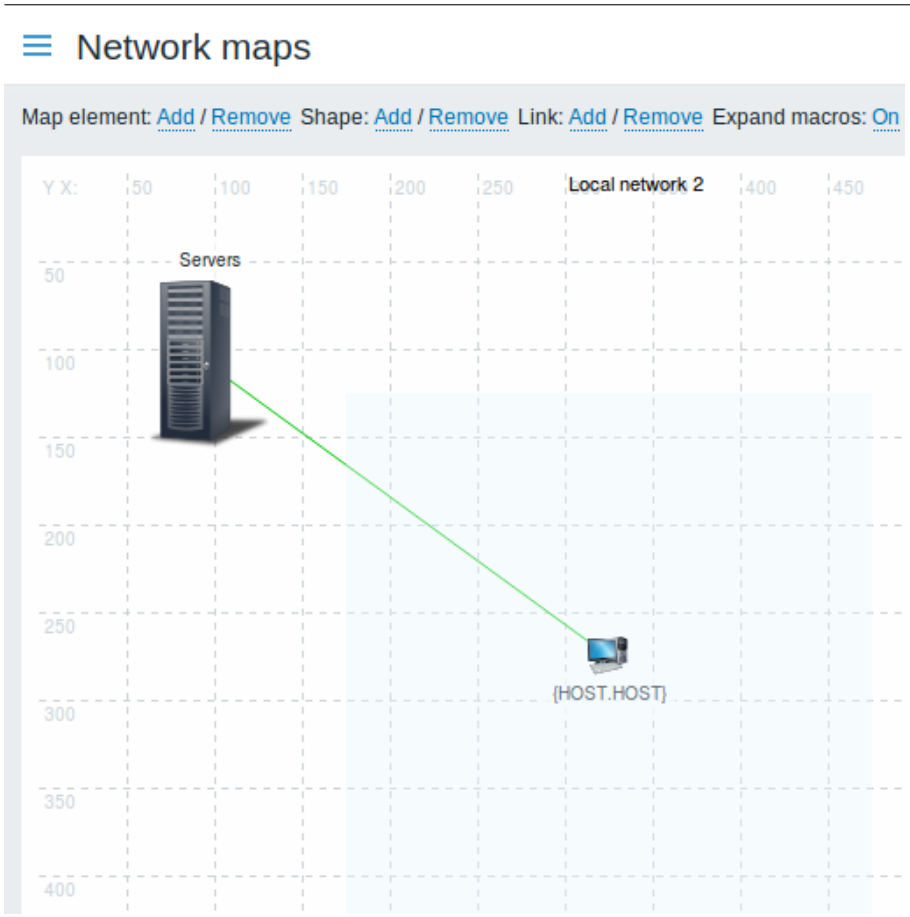
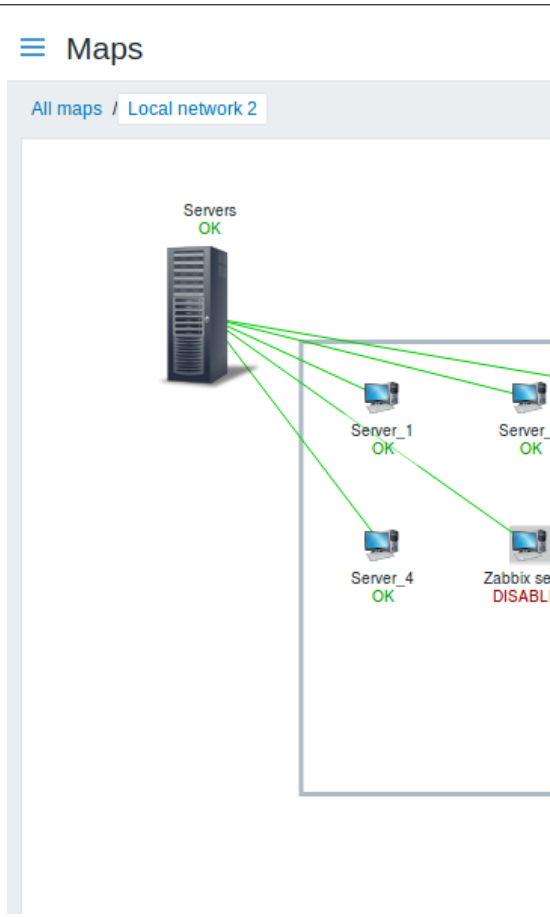
Esta tabla consta de parámetros típicos del tipo de elemento *Host group*:

Parámetro	Descripción
Type	Seleccione el tipo de elemento: **Host group**: icono que representa el estado de todos los triggers de todos los hosts pertenecientes al grupo seleccionado.
Show	Opciones de visualización: **Host group**: al seleccionar esta opción, se mostrará un único icono con la información correspondiente sobre el host group determinado. **Host group elements**: al seleccionar esta opción, se mostrarán varios iconos con la información correspondiente sobre cada elemento (host) del host group determinado.
Area type	Esta configuración está disponible si *Show* está establecido en *Host group elements*: **Fit to map**: todos los elementos del host group se colocan de manera uniforme dentro del mapa. **Custom size**: configuración manual del área del mapa para todos los elementos del host group que se mostrarán.
Area size	Esta configuración está disponible si *Area type* está establecido en *Custom size*: **Width**: valor en píxeles que especifica el ancho del área del mapa. **Height**: valor en píxeles que especifica la altura del área del mapa.
Placing algorithm	**Grid**: única opción disponible para mostrar todos los elementos del host group.

Parámetro	Descripción
Label	Etiqueta del icono, cualquier cadena. En las etiquetas se pueden utilizar macros y cadenas multilínea. Si el tipo de elemento del mapa es *Host group*, especificar determinadas macros afecta a la vista del mapa, que mostrará la información correspondiente sobre cada host. Por ejemplo, si se utiliza la macro `{HOST.IP}`, la vista de edición del mapa solo mostrará la propia macro `{HOST.IP}`, mientras que la vista del mapa incluirá y mostrará la dirección IP única de cada host.

Visualización de los elementos del grupo de hosts

Esta opción está disponible si se selecciona la opción de visualización *Elementos del grupo de hosts*.
Al seleccionar *Elementos del grupo de hosts* como opción de *visualización*, inicialmente solo verá un icono para el grupo de hosts. Sin embargo, al guardar el mapa y acceder a la vista del mapa, verá que el mapa incluye todos los elementos (hosts) del grupo de hosts correspondiente:

Vista de edición del mapa	Vista del mapa
	

Observe cómo se utiliza la macro `{HOST.HOST}`.
En la edición del mapa, el nombre de la macro no se resuelve, mientras que en la vista del mapa se muestran los nombres únicos de todos los hosts.

Indicadores de enlace

Descripción general

Puede asignar indicadores a un enlace entre elementos en un mapa de red.

Los indicadores pueden basarse en triggers o en valores de item. Es posible mostrar un estilo y color de enlace diferentes:

- cuando los triggers pasan a un estado de problema;
- cuando el valor del item:
  - alcanza un umbral (para items numéricos);
  - coincide con una expresión regular (para items de texto).

Cuando configura un enlace, establece el tipo y color predeterminados del enlace. Al asignar indicadores a un enlace, es posible hacer que el estilo y el color del enlace dependan del estado del trigger o del valor del item.

Por ejemplo, si cualquiera de los triggers asignados pasa a un estado de problema, el estilo y el color del enlace cambiarán para reflejarlo. Así, quizá su enlace predeterminado era una línea verde. Ahora, con el trigger en estado de problema, su enlace puede volverse rojo y en negrita (si así lo ha definido).

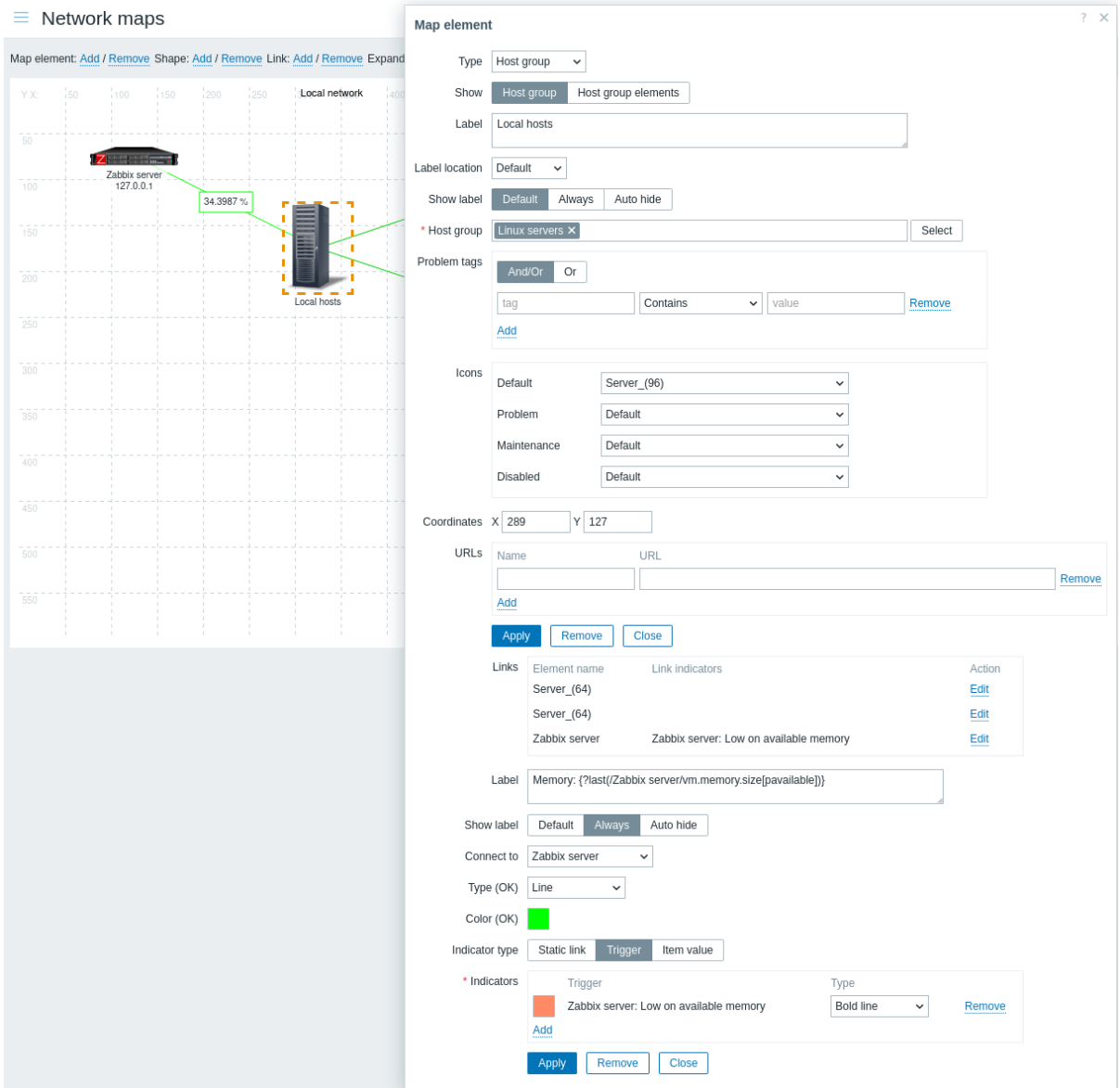
De forma similar, si el valor de un item alcanza un umbral especificado o coincide con una expresión regular especificada, el estilo del enlace puede reflejarlo.

Configuración

Triggers

Para asignar triggers como indicadores de enlace, haga lo siguiente:

- seleccione un elemento del mapa
- haga clic en *Editar* en la sección *Enlaces* para el enlace correspondiente
- seleccione *Trigger* como tipo de indicador
- haga clic en *Añadir* en el bloque *Indicadores* y seleccione uno o más triggers

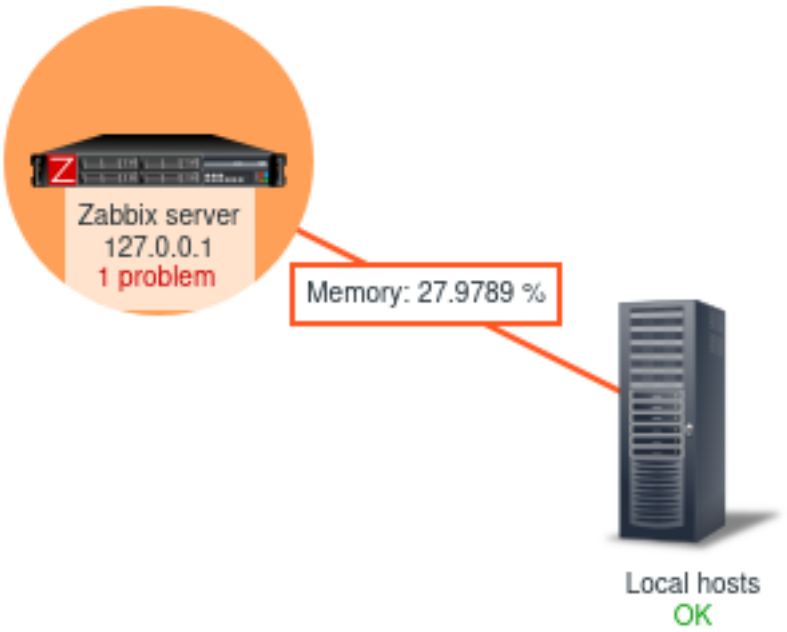


Todos los campos de entrada obligatorios están marcados con un asterisco rojo.

Los triggers añadidos se pueden ver en la lista *Indicadores*.

Puede establecer el tipo de enlace y el color para cada indicador directamente desde la lista. Cuando termine, haga clic en *Aplicar*, cierre el formulario y haga clic en *Actualizar* para guardar los cambios del mapa.

En *Monitoring* > *Maps* se mostrará el tipo de enlace y el color correspondientes si el **trigger** entra en un estado de problema.



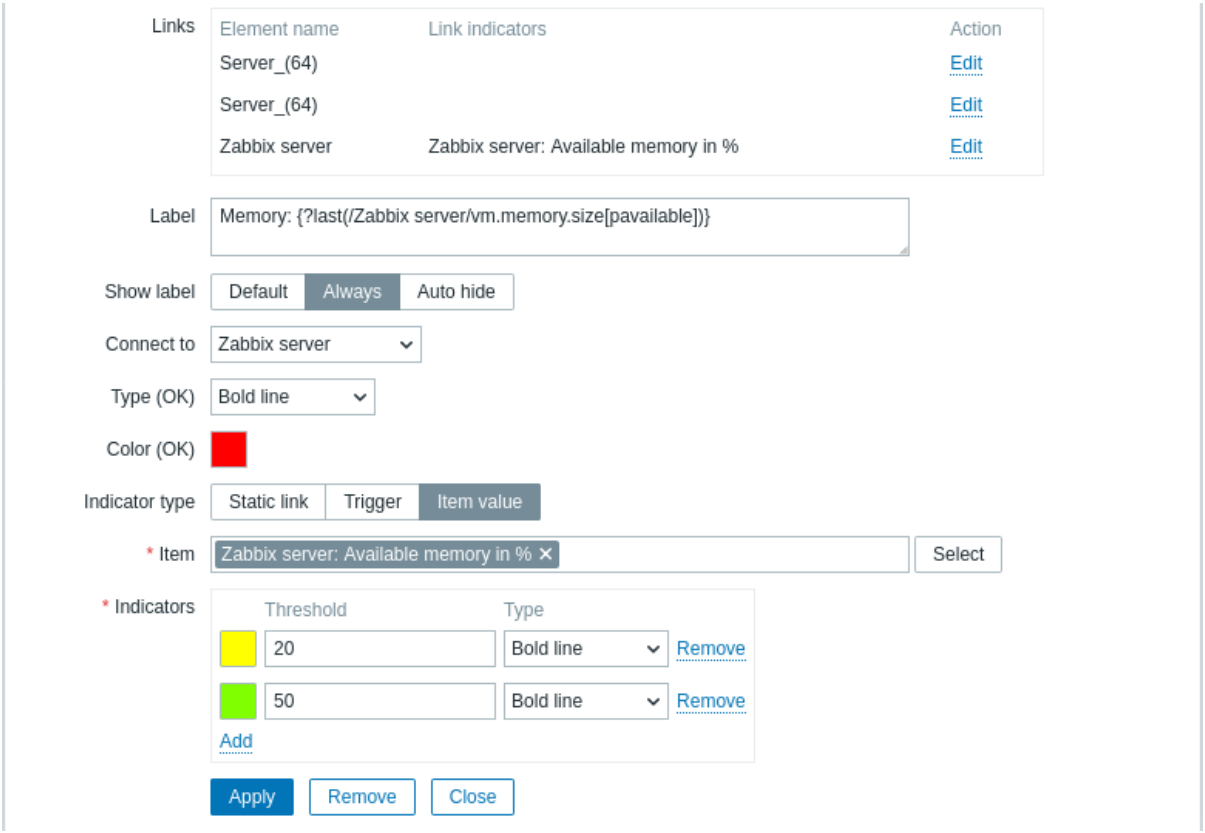
Note:
Si varios triggers entran en un estado de problema, el problema con la severidad más alta determinará el estilo y el color del enlace. Si varios triggers con la misma severidad están asignados al mismo enlace del mapa, tendrá prioridad el que tenga el ID más bajo. Tenga en cuenta también que:

1. Los ajustes de *Severidad mínima del trigger* y *Mostrar problema suprimido* de la configuración del mapa afectan a qué problemas se tienen en cuenta.
2. En el caso de triggers con varios problemas (generación de varios problemas), cada problema puede tener una severidad distinta de la severidad del trigger (cambiada manualmente), puede tener etiquetas diferentes (debido a macros) y puede estar suprimido.

Valores de item

Para asignar valores de item como indicadores de enlace, haga lo siguiente:

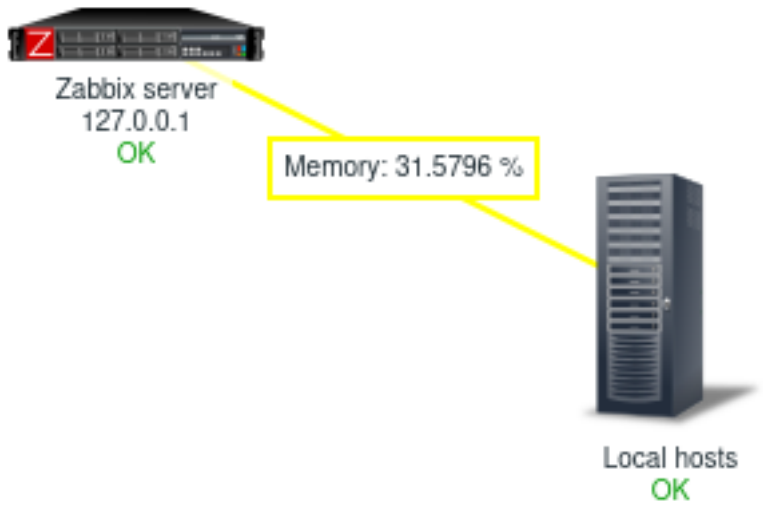
- seleccione un elemento del mapa
- haga clic en *Editar* en la sección *Enlaces* para el enlace correspondiente
- seleccione *Valor de item* como tipo de indicador
- seleccione el item
- añada uno o más umbrales o patrones de valor de item en el bloque *Indicadores*



Los umbrales/patrones de item añadidos se pueden ver en la lista *Indicadores*.

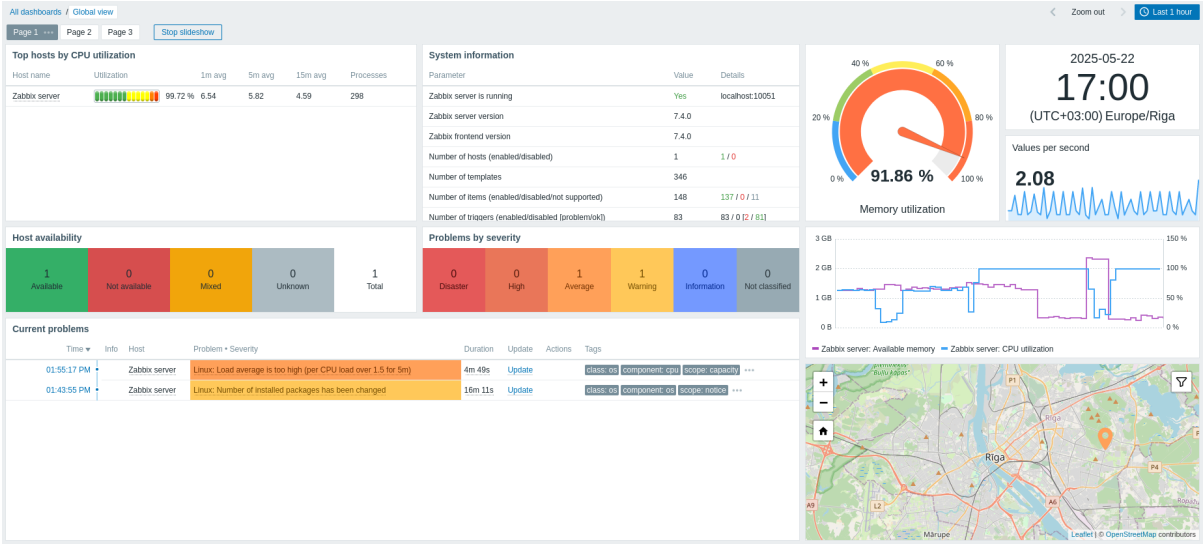
Puede establecer el tipo de enlace y el color para cada indicador directamente desde la lista. Cuando termine, haga clic en *Aplicar*, cierre el formulario y haga clic en *Actualizar* para guardar los cambios del mapa.

En *Monitoring* > *Maps* se mostrará el tipo de enlace y el color correspondientes si el **valor de item** alcanza el umbral establecido (para tipos de datos numéricos) o coincide con el patrón de expresión regular (para tipos de datos de texto).



Paneles

Los paneles, tanto los paneles globales como los paneles de host, proporcionan una potente plataforma de visualización con widgets y herramientas como gráficos modernos, mapas, presentaciones de diapositivas y más.



Templates y grupos de templates

Descripción general

El uso de plantillas es una excelente manera de reducir la carga de trabajo y agilizar la configuración de Zabbix. Una plantilla es un conjunto de entidades que se pueden aplicar cómodamente a múltiples equipos.

Las entidades pueden ser:

- métricas
- disparadores
- gráficos
- paneles
- reglas de descubrimiento de bajo nivel
- escenarios web

Como muchos equipos en la vida real son idénticos o bastante similares, es natural que el conjunto de entidades (métricas, disparadores, gráficos,...) que ha creado para un equipo, pueda ser útil para muchos. Por supuesto, podría copiarlas a cada nuevo equipo, pero eso sería mucho trabajo manual. En su lugar, con las plantillas puede copiarlas a una plantilla y luego aplicar la plantilla a tantos equipos como sea necesario.

Cuando una plantilla se vincula a un equipo, todas las entidades (métricas, disparadores, gráficos,...) de la plantilla se agregan al equipo. Las plantillas se asignan directamente a cada equipo individual (y no a un grupo de equipos).

Las plantillas se utilizan a menudo para agrupar entidades para servicios o aplicaciones particulares (como Apache, MySQL, PostgreSQL, Postfix...) y luego se aplican a los equipos que ejecutan esos servicios.

Otro beneficio de usar plantillas es cuando se debe cambiar algo para todos los equipos. Cambiar algo a nivel de plantilla una vez propagará el cambio a todos los equipos vinculados.

Las plantillas se organizan en grupos de plantillas.

Proceda a crear y configurar una plantilla.

Templates listos para usar

Descripción general

Zabbix proporciona un conjunto cada vez mayor de templates preconfigurados para simplificar y agilizar la configuración de los objetivos de monitorización.

Todos los templates disponibles de forma predeterminada se encuentran en *Recopilación de datos* > *Templates*.

Al actualizar Zabbix, los templates no se actualizan automáticamente. Esto evita sobrescribir cualquier cambio personalizado que haya realizado. Para actualizar o añadir templates manualmente, consulte la sección *Actualización de templates* a continuación.

Utilice la barra lateral para acceder a información sobre tipos específicos de templates y requisitos de funcionamiento.

Los dashboards globales diseñados para funcionar con los templates existentes de Zabbix se pueden descargar desde el repositorio de dashboards e importar en su entorno de Zabbix. En el archivo `README` del dashboard se encuentra disponible una descripción detallada de cada dashboard, incluidos sus requisitos previos.

Consulte también:

- Vinculación de un template
- Problemas conocidos de los templates

Actualización de templates

Para actualizar o añadir un **único template**:

1. Abra el directorio `templates` en el repositorio Git de Zabbix.
2. Elija la rama (en la etiqueta *Source*) que corresponda a su versión de Zabbix (por ejemplo, *master* para la versión de desarrollo).
3. Busque el template que desea añadir o actualizar (por ejemplo, MySQL by Zabbix agent).
4. Copie el contenido del archivo del template (por ejemplo, `template_db_mysql_agent.yaml`) y guárdelo en un archivo local.
5. En la interfaz de usuario de Zabbix, vaya a *Recopilación de datos* > *Templates* e importe el archivo local.

> **Note:**
> Los tipos de medio webhook se pueden actualizar de forma similar: obténgalos del directorio `media` e impórtelos en *Alertas* > *Tipos de medio*.

Para actualizar o añadir **varios templates**:

1. Descargue el archivo del paquete de templates según su web server:
   - Nginx - `templates_bundles-8.0-1M-nginx.zip`
   - Apache - `templates_bundles-8.0-2M-apache.zip`
2. Extraiga el archivo. Contiene archivos YAML, cada uno con varios templates, y un archivo de índice que indica qué templates contiene cada archivo YAML.
3. En la interfaz de usuario de Zabbix, vaya a *Recopilación de datos* > *Templates* e importe los archivos YAML uno por uno.

> **Note:**
> Los tipos de medio webhook se pueden actualizar de forma similar: tome el archivo `media_bundle-8.0-000.yaml` del archivo del paquete de templates e impórtelo en *Alertas* > *Tipos de medio*.

Para actualizar o añadir **todos los templates** a la vez, puede utilizar el script `import_templates.sh` incluido en el archivo del paquete de templates. El script requiere las utilidades `curl` y `jq`, y utiliza el método de la API de Zabbix `configuration.import`. Al ejecutar el script, proporcione el directorio que contiene los archivos YAML, así como la URL de la API de Zabbix y el token de autenticación.

> **Warning:**
> El script `import_templates.sh` sobrescribe todos los templates existentes, sin ofrecer ninguna opción para revisar ni deshacer los cambios en los templates.

Ejemplo:

```
/tmp/templates_bundles-8.0-2M-apache/import_templates.sh \
  /tmp/templates_bundles-8.0-2M-apache \
  https://example.com/zabbix/api_jsonrpc.php \
  d8d6b5c78ee2a8333db11bfcd3e448c49c239d8ce00702759e2b56f42e24dbe4
```

Compatibilidad de los templates con el Host Wizard

El Host Wizard es compatible con todos los templates incluidos de forma predeterminada.

También puede hacer que sus templates personalizados sean compatibles con el Host Wizard:

1. En el frontend de Zabbix, vaya a *Recopilación de datos* > *Templates* y exporte el template personalizado.
2. Edite el archivo exportado:
   - Añada el elemento de template `wizard_ready` para indicar la compatibilidad con el Host Wizard.
   - Opcionalmente, añada el elemento de template `readme` o `config` según sea necesario.
3. En el frontend de Zabbix, vaya a *Recopilación de datos* > *Templates* e importe el archivo de template actualizado.

Operación del template de Zabbix agent

Pasos para garantizar el funcionamiento correcto de los templates que recopilan métricas con Zabbix agent:

1. Asegúrese de que Zabbix agent esté instalado en el host. Para las comprobaciones activas, asegúrese también de que la dirección del server/proxy de Zabbix esté añadida al parámetro 'ServerActive' del archivo de configuración del agent.
2. Vincule el template a un host de destino (si el template no está disponible en su instalación de Zabbix, es posible que primero deba importarlo).
3. Si es necesario, ajuste los valores de las macros del template.
4. Configure la instancia supervisada para permitir el intercambio de datos con Zabbix.

En el archivo README del template (al que se puede acceder haciendo clic en el nombre del template) encontrará una descripción detallada del template, incluida la lista completa de macros, items y triggers.

Los siguientes templates están disponibles:

- Apache by Zabbix agent
- Apache by Zabbix agent active
- HAProxy by Zabbix agent
- IIS by Zabbix agent
- IIS by Zabbix agent active
- MariaDB by Zabbix agent
- MariaDB by Zabbix agent active
- Microsoft Exchange Server 2016 by Zabbix agent
- Microsoft Exchange Server 2016 by Zabbix agent active
- MySQL by Zabbix agent
- MySQL by Zabbix agent active
- Nginx by Zabbix agent
- Nginx by Zabbix agent active
- Percona by Zabbix agent
- Percona by Zabbix agent active
- PHP-FPM by Zabbix agent
- PHP-FPM by Zabbix agent active
- Podman by Zabbix agent
- Podman by Zabbix agent active
- PostgreSQL by Zabbix agent
- PostgreSQL by Zabbix agent active
- RabbitMQ cluster by Zabbix agent

Operación de template de Zabbix agent 2

Pasos para garantizar el funcionamiento correcto de los templates que recopilan métricas con Zabbix agent 2:

1. Asegúrese de que agent 2 esté instalado en el host y de que la versión instalada contenga el plugin requerido. En algunos casos, es posible que primero deba actualizar agent 2.
2. Vincule el template al host de destino (si el template no está disponible en su instalación de Zabbix, es posible que primero deba importar el template).
3. Si es necesario, ajuste los valores de las macros del template. Tenga en cuenta que las macros de usuario pueden utilizarse para sobrescribir parámetros de configuración.
4. Configure la instancia supervisada para permitir el intercambio de datos con Zabbix.

> **Attention:**
> Los templates de Zabbix agent 2 funcionan junto con los plugins. Aunque la configuración básica puede realizarse simplemente ajustando las macros de usuario, la personalización más profunda puede lograrse configurando el plugin en sí. Por ejemplo, si un plugin admite sesiones con nombre, es posible supervisar varias entidades del mismo tipo (por ejemplo, MySQL1 y MySQL2) especificando una sesión con nombre con su propia URI, nombre de usuario y contraseña para cada entidad en el archivo de configuración.

En el archivo README del template (accesible al hacer clic en el nombre de un template) se encuentra una descripción detallada del template, incluida la lista completa de macros, items y triggers.

Los siguientes templates están disponibles:

- Ceph by Zabbix agent 2
- Docker

- Memcached
- MongoDB cluster by Zabbix agent 2
- MongoDB node by Zabbix agent 2
- MySQL by Zabbix agent 2
- MySQL by Zabbix agent 2 active
- Oracle by Zabbix agent 2
- PostgreSQL by Zabbix agent 2
- PostgreSQL by Zabbix agent 2 active
- Redis
- SMART by Zabbix agent 2
- SMART by Zabbix agent 2 active
- Systemd by Zabbix agent 2
- Website certificate by Zabbix agent 2

Operación de templates HTTP

Pasos para garantizar el funcionamiento correcto de los templates que recopilan métricas con el HTTP agent:

1. Cree un host en Zabbix que corresponda al objetivo de monitorización. Especifique la dirección IP o el nombre DNS del objetivo de monitorización como interfaz principal del host. Esto es necesario para que la macro `{HOST.CONN}` se resuelva correctamente en los items del template.
2. Vincule el template al host creado en el paso 1 (si el template no está disponible en su instalación de Zabbix, es posible que primero deba importarlo).
3. Si es necesario, ajuste los valores de las macros del nivel del template.
4. Configure la instancia monitorizada para permitir el intercambio de datos con Zabbix.

En el archivo `README` del template (al que se puede acceder haciendo clic en el nombre de un template) hay disponible una descripción detallada del template, incluida la lista completa de macros, items y triggers.

Están disponibles los siguientes templates:

- Acronis Cyber Protect Cloud by HTTP
- Apache by HTTP
- Asterisk by HTTP
- AWS by HTTP
- AWS Cost Explorer by HTTP
- AWS EC2 by HTTP
- AWS ECS Cluster by HTTP
- AWS ECS Serverless Cluster by HTTP
- AWS ELB Application Load Balancer by HTTP
- AWS ELB Network Load Balancer by HTTP
- AWS Lambda by HTTP
- AWS RDS instance by HTTP
- AWS S3 bucket by HTTP
- Azure by HTTP
- Cisco Meraki organization by HTTP
- Cisco SD-WAN by HTTP
- Cisco Secure Firewall Threat Defense by HTTP
- Claude API by HTTP
- ClickHouse by HTTP
- Cloudflare by HTTP
- CockroachDB by HTTP
- Control-M enterprise manager by HTTP
- Control-M server by HTTP
- Cradlepoint NCM v2 by HTTP
- DELL PowerEdge R720 by HTTP
- DELL PowerEdge R740 by HTTP
- DELL PowerEdge R820 by HTTP
- DELL PowerEdge R840 by HTTP
- Domain RDAP by HTTP
- Elasticsearch Cluster by HTTP
- Envoy Proxy by HTTP
- Etcd by HTTP
- FortiGate by HTTP

- GitHub repository by HTTP
- GitLab by HTTP
- GLPI by HTTP
- Google Cloud Platform (GCP) by HTTP
- Hadoop by HTTP
- HAProxy by HTTP
- HashiCorp Consul Cluster by HTTP
- HashiCorp Consul Node by HTTP
- HashiCorp Nomad by HTTP
- HashiCorp Vault by HTTP
- Hikvision camera by HTTP
- HPE iLO by HTTP
- HPE MSA 2040 Storage by HTTP
- HPE MSA 2060 Storage by HTTP
- HPE Primera by HTTP
- HPE Synergy by HTTP
- InfluxDB by HTTP
- Jenkins by HTTP
- Kubernetes API server by HTTP, Kubernetes cluster state by HTTP, Kubernetes Controller manager by HTTP, Kubernetes kubelet by HTTP, Kubernetes nodes by HTTP, Kubernetes Scheduler by HTTP están obsoletos y han sido reemplazados por los templates *Kubernetes Cluster by HTTP* y *Kubernetes Node by HTTP*.
- Kubernetes Cluster by HTTP
- Kubernetes Node by HTTP
- MantisBT by HTTP
- Microsoft 365 reports by HTTP
- Microsoft SharePoint by HTTP
- NetApp AFF A700 by HTTP
- Nextcloud by HTTP
- NGINX by HTTP
- NGINX Plus by HTTP
- Nutanix Prism Element by HTTP
- OpenAI Platform by HTTP
- OpenStack by HTTP
- OpenWeatherMap by HTTP
- Oracle Cloud by HTTP
- Palo Alto PA-440 by HTTP
- PHP-FPM by HTTP
- Podman by HTTP
- Proxmox VE by HTTP
- Pure Storage FlashArray v1 and v2 by HTTP
- RabbitMQ cluster by HTTP
- Ribbon SBC Edge by HTTP
- Ribbon SBC SWe CE by HTTP
- Ribbon SBC SWe Core by HTTP
- TiDB by HTTP
- TiDB PD by HTTP
- TiDB TiKV by HTTP
- Travis CI by HTTP
- Veeam Backup Enterprise Manager by HTTP
- Veeam Backup and Replication by HTTP
- VeloCloud SD-WAN by HTTP
- VeloCloud SD-WAN Edge by HTTP
- YugabyteDB by HTTP
- ZooKeeper by HTTP

Operación de template IPMI

Los templates IPMI no requieren ninguna configuración específica. Para comenzar a supervisar, vincule el template a un host de destino (si el template no está disponible en su instalación de Zabbix, es posible que primero deba importar el template).

Una descripción detallada de un template, incluida la lista completa de macros, items y triggers, está disponible en el archivo README del template (accesible al hacer clic en el nombre de un template).

Template disponible:

- Chassis by IPMI

Operación de template JMX

Pasos para garantizar el funcionamiento correcto de los templates que recopilan métricas mediante JMX:

1. Asegúrese de que Zabbix Java gateway esté instalado y configurado correctamente.
2. Vincule el template al host de destino. El host debe tener configurada la interfaz JMX. Si el template no está disponible en su instalación de Zabbix, es posible que primero deba importar el template.
3. Si es necesario, ajuste los valores de las macros del template.
4. Configure la instancia supervisada para permitir el intercambio de datos con Zabbix.

En el archivo README del template, al que se puede acceder haciendo clic en el nombre del template, encontrará una descripción detallada del template, incluida la lista completa de macros, items y triggers.

Los siguientes templates están disponibles:

- Apache ActiveMQ by JMX
- Apache Cassandra by JMX
- Apache Kafka by JMX
- Apache Tomcat by JMX
- GridGain by JMX
- Ignite by JMX
- Jira Data Center by JMX
- WildFly Domain by JMX
- WildFly Server by JMX

Operación de template ODBC

Pasos para garantizar el funcionamiento correcto de los templates que recopilan métricas mediante monitorización ODBC:

1. Asegúrese de que el controlador ODBC requerido esté instalado en el server o proxy de Zabbix.
2. Vincule el template al host de destino (si el template no está disponible en su instalación de Zabbix, es posible que primero deba importar el template).
3. Si es necesario, ajuste los valores de las macros del template.
4. Configure la instancia supervisada para permitir el intercambio de datos con Zabbix.

En el archivo README del template (accesible al hacer clic en el nombre de un template) se encuentra una descripción detallada del template, incluida la lista completa de macros, items y triggers.

Los siguientes templates están disponibles:

- MariaDB by ODBC
- MSSQL by ODBC
- MySQL by ODBC
- Oracle by ODBC
- Percona by ODBC
- PostgreSQL by ODBC

Plantillas estandarizadas para dispositivos de red

Descripción general

Para proporcionar monitorización de dispositivos de red como switches y routers, hemos creado dos llamados modelos: uno para el propio dispositivo de red (básicamente su chasis) y otro para la interfaz de red.

Se proporcionan templates para muchas familias de dispositivos de red. Todos los templates cubren (cuando es posible obtener estos elementos del dispositivo):

- Monitorización de fallos del chasis (fuentes de alimentación, ventiladores y temperatura, estado general)
- Monitorización del rendimiento del chasis (items de CPU y memoria)
- Recopilación de inventario del chasis (números de serie, nombre del modelo, versión del firmware)
- Monitorización de la interfaz de red con IF-MIB y EtherLike-MIB (estado de la interfaz, carga de tráfico de la interfaz, estado dúplex para Ethernet)

Si el template no está disponible en su instalación de Zabbix, es posible que primero deba importarlo.

Si está importando los nuevos templates predeterminados, también puede querer actualizar la expresión regular global `@Network interfaces for discovery` a:

```
No coincide con la expresión regular: ^Software Loopback Interface
No coincide con la expresión regular: ^(In)?[lL]oop[bB]ack[0-9._]*$
No coincide con la expresión regular: ^NULL[0-9.]*$
No coincide con la expresión regular: ^[lL]o[0-9.]*$
No coincide con la expresión regular: ^[sS]ystem$
No coincide con la expresión regular: ^Nu[0-9.]*$
```

para filtrar las interfaces loopback y null en la mayoría de los sistemas.

Dispositivos

Lista de familias de dispositivos para las que hay templates disponibles:

Nombre del template	Proveedor	Familia de dispositivos	Modelos conocidos	SO	MIB utilizados	**Etiquetas**
Alcatel Timetra TiMOS SNMP	Alcatel	Alcatel Timetra	ALCATEL SR 7750	TiMOS	TIMETRA-SYSTEM-MIB,TIMETRA-CHASSIS-MIB	Certificado
Aruba CX 8300s by SNMP	HPE Aruba Networking	Serie de switches Aruba CX	Aruba JL636A 8325, Aruba JL717A 8360	Aruba AOS-CX	ARUBAWIRED-FAN-MIB, ARUBAWIRED-POWERSUPPLY-MIB, ARUBAWIRED-SYSTEMINFO-MIB, ARUBAWIRED-TEMPSENSOR-MIB, OSPF-MIB	Certificado
Brocade FC SNMP	Brocade	Switches FC de Brocade	Brocade 300 SAN Switch-	-	SW-MIB,ENTITY-MIB	Rendimiento Fallos
Brocade_Foundry Stackable SNMP	Brocade	Brocade ICX	Brocade ICX6610, Brocade ICX7250-48, Brocade ICX7450-48F		FOUNDRY-SN-AGENT-MIB, FOUNDRY-SN-STACKING-MIB	Certificado
Brocade_Foundry Nonstackable SNMP	Brocade, Foundry	Brocade MLX, Foundry	Brocade MLXe, Foundry FLS648, Foundry FWSX424		FOUNDRY-SN-AGENT-MIB	Rendimiento Fallos
Check Point Next Generation Firewall by SNMP	Check Point	Firewall de próxima generación	-	Gaia	HOST-RESOURCES-MIB, CHECKPOINT-MIB, UCD-SNMP-MIB, SNMPv2-MIB, IF-MIB	Certificado
Ciena 3906 by SNMP	Ciena	CPE de Ciena	Ciena 3906	SAOS	WWP-LEOS-BLADE-MIB, WWP-LEOS-CHASSIS-MIB, WWP-LEOS-SYSTEM-CONFIG-MIB	Certificado

Nombre del template	Proveedor	Familia de dispositivos	Modelos conocidos	SO	MIB utilizados	**Etiquetas**
Cisco Catalyst 3750<device model> SNMP	Cisco	Cisco Catalyst 3750	Cisco Catalyst 3750V2-24FS, Cisco Catalyst 3750V2-24PS, Cisco Catalyst 3750V2-24TS, Cisco Catalyst SNMP, Cisco Catalyst SNMP		CISCO-MEMORY-POOL-MIB, IF-MIB, EtherLike-MIB, SNMPv2-MIB, CISCO-PROCESS-MIB, CISCO-ENVMON-MIB, ENTITY-MIB	Certificado
Cisco IOS SNMP	Cisco	Cisco IOS ver > 12.2 3.5	Cisco C2950	IOS	CISCO-PROCESS-MIB,CISCO-MEMORY-POOL-MIB,CISCO-ENVMON-MIB	Certificado
Cisco IOS versions 12.0_3_T-12.2_3.5 SNMP	Cisco	Cisco IOS > 12.0 3 T y 12.2 3.5	-	IOS	CISCO-PROCESS-MIB,CISCO-MEMORY-POOL-MIB,CISCO-ENVMON-MIB	Certificado
Cisco IOS prior to 12.0_3_T SNMP	Cisco	Cisco IOS 12.0 3 T	-	IOS	OLD-CISCO-CPU-MIB,CISCO-MEMORY-POOL-MIB	Certificado
D-Link DES_DGS Switch SNMP	D-Link	Switches DES/DGX	D-Link DES-xxxx/DGS-xxxx,DLINK DGS-3420-26SC	-	DLINK-AGENT-MIB,EQUIPMENT-MIB,ENTITY-MIB	Certificado
D-Link DES 7200 SNMP	D-Link	DES-7xxx	D-Link DES 7206	-	ENTITY-MIB,MY-SYSTEM-MIB,MY-PROCESS-MIB,MY-MEMORY-MIB	Rendimiento Fallos Interfaces
Dell Force S-Series SNMP	Dell	Dell Force S-Series	S4810		F10-S-SERIES-CHASSIS-MIB	Certificado
Extreme Exos SNMP	Extreme	Extreme EXOS	X670V-48x	EXOS	EXTREME-SYSTEM-MIB,EXTREME-SOFTWARE-MONITOR-MIB	Certificado
FortiGate by SNMP	Fortinet	FortiGate (NGFW)	-	FortiOS	HOST-RESOURCES-MIB FORTINET-FORTIGATE-MIB FORTINET-CORE-MIB SNMPv2-MIB IF-MIB ENTITY-MIB	Rendimiento Inventario
HP Comware HH3C SNMP	HP	HP (H3C) Comware	HP A5500-24G-4SFP HI Switch		HH3C-ENTITY-EXT-MIB,ENTITY-MIB	Certificado
HP Enterprise Switch SNMP	HP	HP Enterprise Switch	HP ProCurve J4900B Switch 2626, HP J9728A 2920-48G Switch		STATISTICS-MIB,NETSWITCH-MIB,HP-ICF-CHASSIS,ENTITY-MIB,SEMI-MIB	Certificado

Nombre del template	Proveedor	Familia de dispositivos	Modelos conocidos	SO	MIB utilizados	**Etiquetas**
Huawei AR600 by SNMP	Huawei	Routers Huawei NetEngine serie AR600	Huawei AR600/AR6100/AR6200/AR6300 series routers with VRP software (V300R019/V300R021/V300R023 or later)		HUAWEI-CBQOS-MIB, HUAWEI-NQA-MIB, ENTITY-MIB, IF-MIB, HOST-RESOURCES-MIB, SNMPv2-MIB	Certificado
Huawei OceanStor 5300 V5 by SNMP	Huawei	Huawei OceanStor Dorado	Huawei OceanStor Dorado with V5 software		HUAWEI-STORAGE-HARDWARE-MIB, HUAWEI-STORAGE-SPACE-MIB, ISM-PERFORMANCE-MIB, HOST-RESOURCES-MIB, SNMPv2-MIB	Certificado
Huawei OceanStor Dorado by SNMP	Huawei	Huawei OceanStor, Huawei OceanStor Dorado	Huawei OceanStor Dorado 3000/5000/6000/8000/18000/5300V5/5500/5600/6800/18500/18800 with V6/V700 software		HUAWEI-STORAGE-HARDWARE-MIB, HUAWEI-STORAGE-SPACE-MIB, ISM-PERFORMANCE-MIB, HOST-RESOURCES-MIB, SNMPv2-MIB	Certificado
Huawei VRP by SNMP	Huawei	Huawei VRP	S2352P-EI		ENTITY-MIB,HUAWEI-ENTITY-EXTENT-MIB	Certificado
Intel_Qlogic Infiniband SNMP	Intel/QLogic	Dispositivos Infiniband de Intel/QLogic	Infiniband 12300		ICS-CHASSIS-MIB	Fallos Inventario
Juniper SNMP	Juniper	Modelos MX,SRX,EX	Juniper MX240, Juniper EX4200-24F	JunOS	JUNIPER-MIB	Certificado
Juniper MX NETCONF	Juniper	Modelos MX	Juniper MX204 Edge Router	JunOS 24.2R1-S1.10	*No aplicable (usa NETCONF sobre SSH)*	Certificado
Juniper MX SNMP	Juniper	Modelos MX	Juniper MX204 Edge Router	JunOS 24.2R1-S1.10	OSPF-MIB, JUNIPER-DOM-MIB, JUNIPER-MIB, BGP4-V2-MIB-JUNIPER, OSPFV3-MIB-JUNIPER	Certificado
Mellanox SNMP	Mellanox	Dispositivos Infiniband de Mellanox	SX1036	MLNX-OS	HOST-RESOURCES-MIB,ENTITY-MIB,ENTITY-SENSOR-MIB,MELLANOX-MIB	Certificado

Nombre del template	Proveedor	Familia de dispositivos	Modelos conocidos	SO	MIB utilizados	**Etiquetas**
MikroTik CCR<device model> SNMP	MikroTik	Routers MikroTik Cloud Core (serie CCR)	Hay templates dedicados independientes disponibles para MikroTik CCR1009-7G-1C-1S+, MikroTik CCR1009-7G-1C-1S+PC, MikroTik CCR1009-7G-1C-PC, MikroTik CCR1016-12G, MikroTik CCR1016-12S-1S+, MikroTik CCR1036-12G-4S-EM, MikroTik CCR1036-12G-4S, MikroTik CCR1036-8G-2S+, MikroTik CCR1036-8G-2S+EM, MikroTik CCR1072-1G-8S+, MikroTik CCR2004-16G-2S+, MikroTik CCR2004-1G-12S+2XS	RouterOS	MIKROTIK-MIB,HOST-RESOURCES-MIB	Certificado

Nombre del template	Proveedor	Familia de dispositivos	Modelos conocidos	SO	MIB utilizados	**Etiquetas**
MikroTik CRS<device model> SNMP	MikroTik	MikroTik Cloud Router Switches (serie CRS)	Hay templates dedicados independientes disponibles para MikroTik CRS106-1C-5S, MikroTik CRS109-8G-1S-2HnD-IN, MikroTik CRS112-8G-4S-IN, MikroTik CRS112-8P-4S-IN, MikroTik CRS125-24G-1S-2HnD-IN, MikroTik CRS212-1G-10S-1S+IN, MikroTik CRS305-1G-4S+IN, MikroTik CRS309-1G-8S+IN, MikroTik CRS312-4C+8XG-RM, MikroTik CRS317-1G-16S+RM, MikroTik CRS326-24G-2S+IN, MikroTik CRS326-24G-2S+RM, MikroTik CRS326-24S+2Q+RM, MikroTik CRS328-24P-4S+RM, MikroTik CRS328-4C-20S-4S+RM, MikroTik CRS354-48G-4S+2Q+RM, MikroTik CRS354-48P-4S+2Q+RM	RouterOS/SwitchOS	MIKROTIK-MIB,HOST-RESOURCES-MIB	Certificado
MikroTik CSS<device model> SNMP	MikroTik	MikroTik Cloud Smart Switches (serie CSS)	Hay templates dedicados independientes disponibles para MikroTik CSS326-24G-2S+RM, MikroTik CSS610-8G-2S+IN	RouterOS	MIKROTIK-MIB,HOST-RESOURCES-MIB	Certificado
MikroTik FiberBox SNMP	MikroTik	MikroTik FiberBox	MikroTik FiberBox	RouterOS	MIKROTIK-MIB,HOST-RESOURCES-MIB	Certificado
MikroTik hEX <device model> SNMP	MikroTik	MikroTik hEX	Hay templates dedicados independientes disponibles para MikroTik hEX, MikroTik hEX lite, MikroTik hEX PoE, MikroTik hEX PoE lite, MikroTik hEX S	RouterOS	MIKROTIK-MIB,HOST-RESOURCES-MIB	Certificado

Nombre del template	Proveedor	Familia de dispositivos	Modelos conocidos	SO	MIB utilizados	**Etiquetas**
MikroTik netPower <device model> SNMP	MikroTik	MikroTik netPower	Hay templates dedicados independientes disponibles para MikroTik netPower 15FR, MikroTik netPower 16P SNMP, MikroTik netPower Lite 7R	RouterOS/SwitchOS, SwitchOS Lite	MIKROTIK-MIB,HOST-RESOURCES-MIB	Certificado
MikroTik PowerBox <device model> SNMP	MikroTik	MikroTik PowerBox	Hay templates dedicados independientes disponibles para MikroTik PowerBox, MikroTik PowerBox Pro	RouterOS	MIKROTIK-MIB,HOST-RESOURCES-MIB	Certificado
MikroTik RB<device model> SNMP	MikroTik	Routers MikroTik serie RB	Hay templates dedicados independientes disponibles para MikroTik RB1100AHx4, MikroTik RB1100AHx4 Dude Edition, MikroTik RB2011iL-IN, MikroTik RB2011iL-RM, MikroTik RB2011iLS-IN, MikroTik RB2011UiAS-IN, MikroTik RB2011UiAS-RM, MikroTik RB260GS, MikroTik RB3011UiAS-RM, MikroTik RB4011iGS+RM, MikroTik RB5009UG+S+IN	RouterOS	MIKROTIK-MIB,HOST-RESOURCES-MIB	Certificado
MikroTik SNMP	MikroTik	Dispositivos MikroTik RouterOS	MikroTik CCR1016-12G, MikroTik RB2011UAS-2HnD, MikroTik 912UAG-5HPnD, MikroTik 941-2nD, MikroTik 951G-2HnD, MikroTik 1100AHx2	RouterOS	MIKROTIK-MIB,HOST-RESOURCES-MIB	Certificado

Nombre del template	Proveedor	Familia de dispositivos	Modelos conocidos	SO	MIB utilizados	**Etiquetas**
Netgear Fastpath SNMP	Netgear	Netgear Fastpath	M5300-28G		FASTPATH-SWITCHING-MIB,FASTPATH-BOXSERVICES-PRIVATE-MIB	Fallos Inventario
QTech QSW SNMP	QTech	Dispositivos Qtech	Qtech QSW-2800-28T	-	QTECH-MIB,ENTITY-MIB	Rendimiento Inventario
Stormshield SNS by SNMP	Stormshield	Firewalls Stormshield Network Security (SNS)	SN3100	Stormshield SNS	HOST-RESOURCES-MIB, UCD-SNMP-MIB, STORMSHIELD-ASQ-STATS-MIB, STORMSHIELD-AUTOUPDATE-MIB, STORMSHIELD-HA-MIB, STORMSHIELD-PROPERTY-MIB, STORMSHIELD-HEALTH-MONITOR-MIB, STORMSHIELD-IF-MIB, STORMSHIELD-SYSTEM-MONITOR-MIB, STORMSHIELD-IPSEC-STATS-MIB	Certificado
TP-LINK SNMP	TP-LINK	TP-LINK	T2600G-28TS v2.0		TPLINK-SYSMONITOR-MIB,TPLINK-SYSINFO-MIB	Rendimiento Inventario
Ubiquiti AirOS SNMP	Ubiquiti	Dispositivos inalámbricos Ubiquiti AirOS	NanoBridge,NanoStation,Unifi	AirOS	FROGFOOT-RESOURCES-MIB,IEEE802dot11-MIB	Rendimiento
Vyatta Virtual Router by SNMP	Ciena	Vyatta	Vyatta Virtual Router 1908e	Vyatta 1908e	SNMPv2-MIB, HOST-RESOURCES-MIB, UCD-SNMP-MIB, IF-MIB, DISMAN-EVENT-MIB	Rendimiento Inventario

Diseño de la plantilla

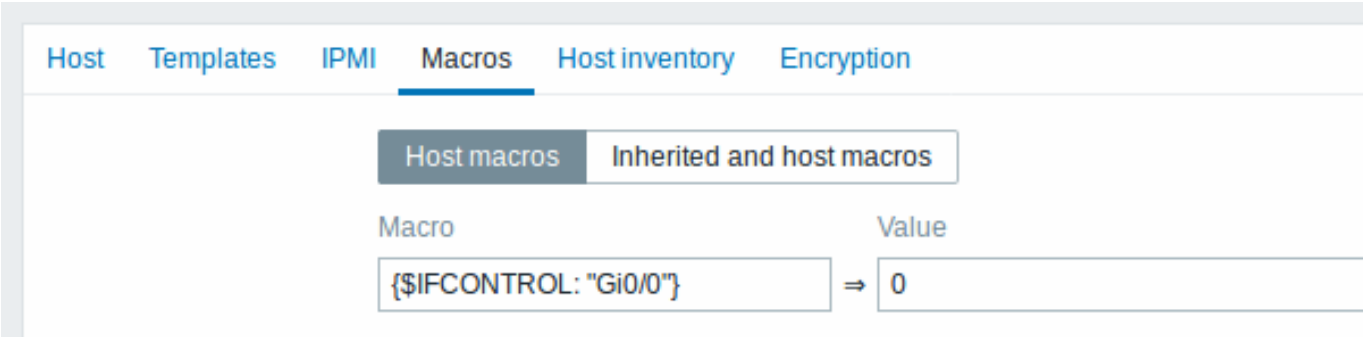
Las plantillas se diseñaron teniendo en cuenta lo siguiente:

- Se utilizan macros de usuario tanto como sea posible para que los disparadores puedan ser ajustados por el usuario;
- Se utiliza el descubrimiento de bajo nivel tanto como sea posible para minimizar el número de métricas no soportadas;
- Todas las plantillas dependen de la plantilla ICMP Ping, por lo que todos los dispositivos también se comprueban mediante ICMP;
- Las métricas no utilizan ningún MIB: se utilizan OIDs SNMP en las métricas y descubrimientos de bajo nivel. Por lo tanto, no es necesario cargar ningún MIB en Zabbix para que las plantillas funcionen;
- Las interfaces de red de loopback se filtran durante el descubrimiento, así como las interfaces con ifAdminStatus = down(2)
- Se utilizan contadores de 64 bits de IF-MIB::ifXTable cuando es posible. Si no se soportan, se utilizan en su lugar los contadores de 32 bits por defecto.

Todas las interfaces de red descubiertas tienen un disparador que monitoriza su estado operativo (enlace), por ejemplo:

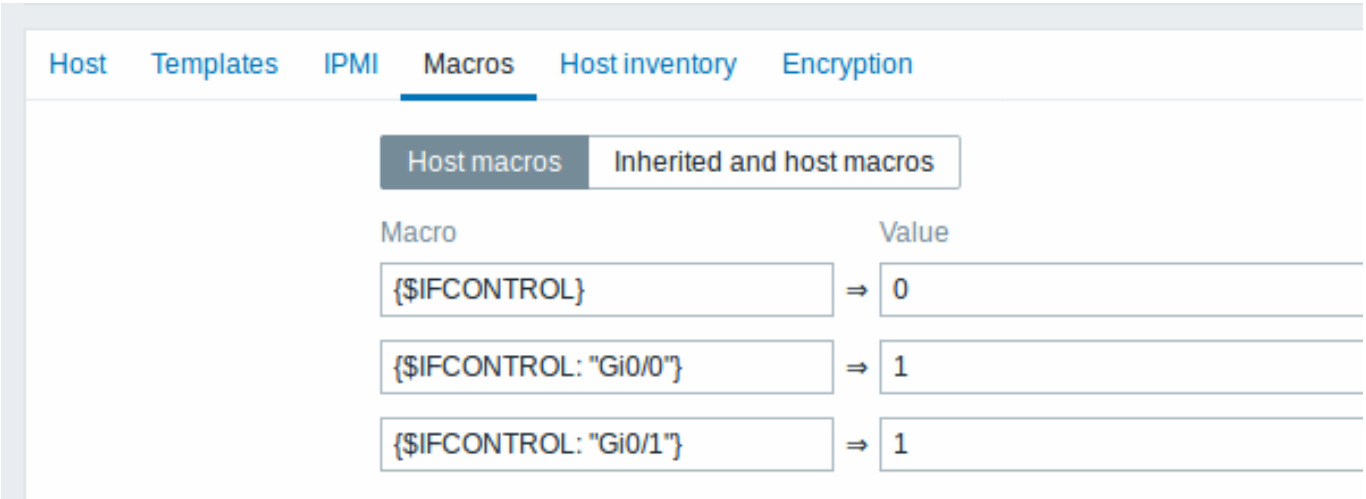
```
{$IFCONTROL:"{#IFNAME}"}=1 and last(/Alcatel Timetra TiMOS SNMP/net.if.status[ifOperStatus.{#SNMPINDEX}])
```

- Si no desea monitorizar esta condición para una interfaz específica, cree una macro de usuario con contexto con el valor 0. Por ejemplo:



donde Gi0/0 es {#IFNAME}. De esta manera, el disparador ya no se utiliza para esta interfaz específica.

- También puede cambiar el comportamiento por defecto para que todos los disparadores no se activen y activar este disparador solo para un número limitado de interfaces, como enlaces ascendentes:



Etiquetas

- Rendimiento: los MIB de la familia de dispositivos proporcionan una forma de monitorizar items de CPU y memoria;
- Fallos: los MIB de la familia de dispositivos proporcionan una forma de monitorizar al menos un sensor de temperatura;
- Inventario: los MIB de la familia de dispositivos proporcionan una forma de recopilar al menos el número de serie y el nombre del modelo del dispositivo;
- Certificado: se cubren las tres categorías principales anteriores.

Operación de template de VMware

Descripción general

Zabbix incluye una variedad de plantillas listas para usar para la monitorización de máquinas virtuales, diseñadas para los hipervisores VMware vCenter y ESXi. Las plantillas disponibles se encuentran en dos conjuntos de plantillas separados:

- VMware: utiliza datos UUID para las macros correspondientes
- VMware FQDN: utiliza datos FQDN para las macros correspondientes

Las plantillas diseñadas para monitorizar VMware vCenter o hipervisores ESXi contienen reglas de descubrimiento de bajo nivel preconfiguradas junto con varias comprobaciones integradas para monitorizar instalaciones virtuales.

Note:
Para el correcto funcionamiento de la plantilla *VMware FQDN*, cada máquina virtual monitorizada debe tener un nombre de SO único que cumpla con las reglas FQDN. Además, VMware Tools/Open Virtual Machine tools deben estar instaladas en cada máquina. Si se cumplen estos requisitos previos, se recomienda utilizar la plantilla *VMware FQDN*. La plantilla *VMware FQDN* está disponible desde Zabbix 5.2 con la introducción de la capacidad de crear equipos con interfaces personalizadas.

 También está disponible una plantilla clásica *VMware* que puede utilizarse si no se cumplen los requisitos FQDN. Sin embargo, la plantilla *VMware* tiene un problema conocido. Los equipos para las máquinas virtuales descubiertas se crean con los nombres que se guardan en vCenter (por ejemplo, ”VM1”, ”VM2”, etc.). Si el agente Zabbix está instalado en estos equipos y el autoregistro activo del agente Zabbix está habilitado, el proceso de autoregistro leerá los nombres de los equipos tal como se registraron durante el inicio (por ejemplo, ”vm1.example.com”, ”vm2.example.com”, etc.). Esto puede llevar a la creación de nuevos equipos para máquinas virtuales existentes (ya que no se han encontrado coincidencias de nombres), lo que resulta en equipos duplicados con diferentes nombres.

Tenga en cuenta:

- La plantilla *VMware Hypervisor* puede vincularse manualmente a un equipo, así como utilizarse en el descubrimiento;
- La plantilla *VMware Guest* no debe vincularse manualmente a un equipo y solo puede utilizarse en el descubrimiento.

Configuración de macros de equipo

Para utilizar comprobaciones simples de VMware, el equipo debe tener definidas las siguientes macros de usuario:

- {$VMWARE.URL} - URL del SDK del servicio VMware (vCenter o hipervisor ESXi) (https://servername/sdk)
- {$VMWARE.USERNAME} - Nombre de usuario del servicio VMware
- {$VMWARE.PASSWORD} - Contraseña del usuario {$VMWARE.USERNAME} del servicio VMware

Operación del template OTLP

Zabbix proporciona una variedad de templates listos para usar para recopilar y supervisar trazas, métricas y registros de OpenTelemetry mediante items de *consulta de telemetría*.

Pasos para garantizar el funcionamiento correcto de los templates:

1. Consulte el archivo `README` del template para conocer los requisitos de configuración específicos del template.
2. Cree un host en Zabbix que corresponda al objetivo de supervisión.
3. Asegúrese de que el parámetro `TelemetryProvider` esté configurado en el archivo de configuración del server de Zabbix o del proxy de Zabbix.
4. Vincule el template al host creado (si el template no está disponible en su instalación de Zabbix, es posible que primero deba importarlo).
5. Si es necesario, ajuste los valores de las macros a nivel del template.

Encontrará una descripción detallada de los templates, incluida la lista completa de macros, items y triggers, en el archivo `README` del template, al que puede acceder haciendo clic en el nombre del template.

Están disponibles los siguientes templates:

- Envoy Proxy by OTLP
- Generic OpenTelemetry by OTLP
- OpenRouter traces by OTLP
- Proxmox VE by OTLP

Notificaciones al producirse eventos

Descripción general

Suponiendo que hemos configurado algunas métricas y disparadores y ahora estamos recibiendo algunos eventos como resultado de que los disparadores cambian de estado, es momento de considerar algunas acciones.

Para empezar, no querríamos estar mirando la lista de disparadores o eventos todo el tiempo. Sería mucho mejor recibir una notificación si ocurre algo significativo (como un problema). Además, cuando ocurren problemas, nos gustaría asegurarnos de que todas las personas involucradas estén informadas.

Por eso, el envío de notificaciones es una de las acciones principales que ofrece Zabbix. Se puede definir quién y cuándo debe ser notificado ante un determinado evento.

Para poder enviar y recibir notificaciones desde Zabbix debe:

- definir algún medio
- configurar una acción que envíe un mensaje a uno de los medios definidos

Las acciones constan de *condiciones* y *operaciones*. Básicamente, cuando se cumplen las condiciones, se llevan a cabo las operaciones. Las dos operaciones principales son enviar un mensaje (notificación) y ejecutar un comando remoto.

Para los eventos creados por descubrimiento y autorregistro, hay disponibles algunas operaciones adicionales. Estas incluyen agregar o eliminar un equipo, vincular una plantilla, etc.

Tipos de medios

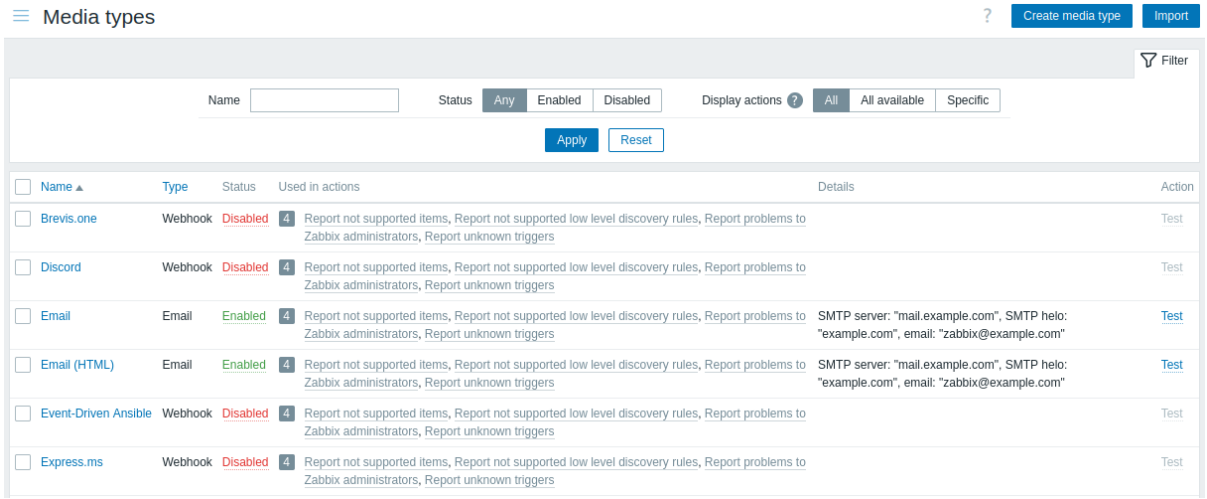
Descripción general

Los tipos de medios son los canales de entrega utilizados para enviar notificaciones y alertas desde Zabbix.

Los tipos de medios admiten los siguientes métodos de entrega:

- Correo electrónico
- Notificación push
- SMS
- Script personalizado
- Webhook

Los tipos de medios se administran en *Alertas* > *Tipos de medios*. Algunos tipos de medios vienen predefinidos en el conjunto de datos predeterminado. Solo es necesario ajustar sus parámetros para que funcionen.



Para ver cómo encajan los tipos de medios en el proceso de alertas, revisemos los tres requisitos para entregar notificaciones de Zabbix a los usuarios finales:

1. Debe definirse una operación de acción que envíe notificaciones.
2. Debe definirse un tipo de medio funcional, como *Correo electrónico*, que envíe alertas mediante SMTP.
3. Los detalles de entrega a nivel de usuario, como direcciones de correo electrónico, números de teléfono, etc., deben definirse en medios del usuario.

Pruebas del tipo de medio

Para comprobar si un tipo de medio configurado funciona, haga clic en *Probar* en la lista de tipos de medios.

La solicitud de prueba se enviará al Zabbix server. El Zabbix server intentará enviar una alerta utilizando el tipo de medio especificado y devolverá el resultado al frontend. El frontend esperará a que el server devuelva los resultados. Las pruebas de tipos de medios tienen un tiempo de espera de 65 segundos de forma predeterminada (configurable en *Administración* > *General* > *Tiempos de espera*).

Consulte también los detalles de las pruebas para:

- Correo electrónico
- Notificación push
- Webhook
- Script

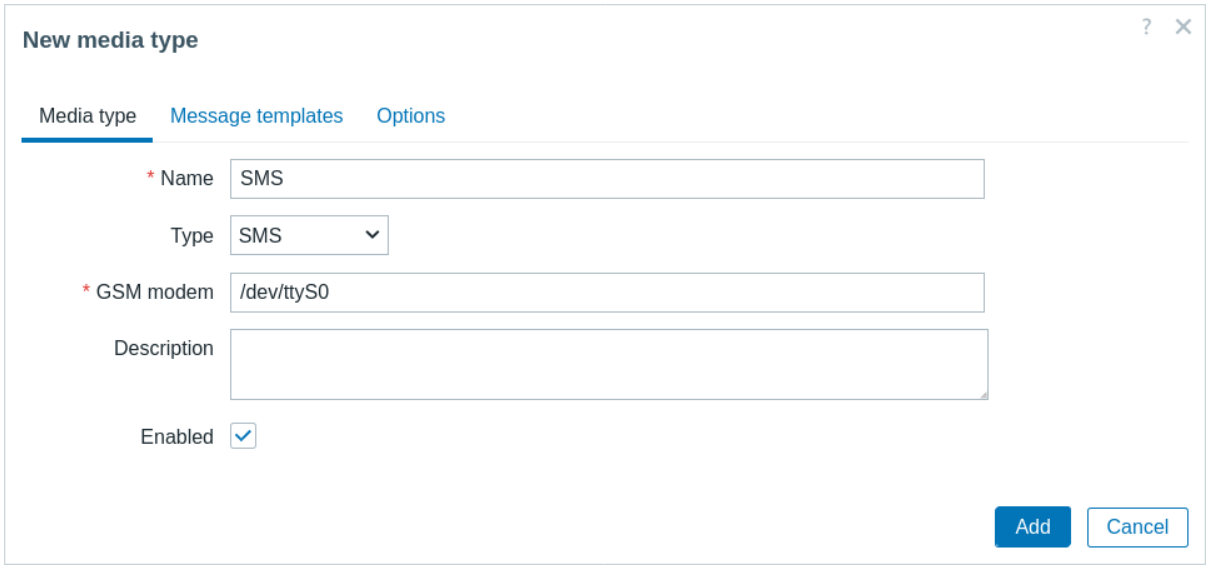
Configuración

Para crear un tipo de medio en el frontend de Zabbix:

1. Vaya a *Alertas* > *Tipos de medios*.
2. Haga clic en *Crear tipo de medio*.
3. Introduzca los parámetros del tipo de medio en el formulario.

Algunos parámetros son **comunes** a todos los métodos de entrega.

Note:
Si se especifican uno o más métodos de entrega en el indicador de funcionalidad `$ZBX_FEATURE_FLAGS['media_type_denylist']` del archivo de configuración del frontend (`zabbix.conf.php`), esos métodos de entrega no estarán disponibles en la lista *Tipo*.

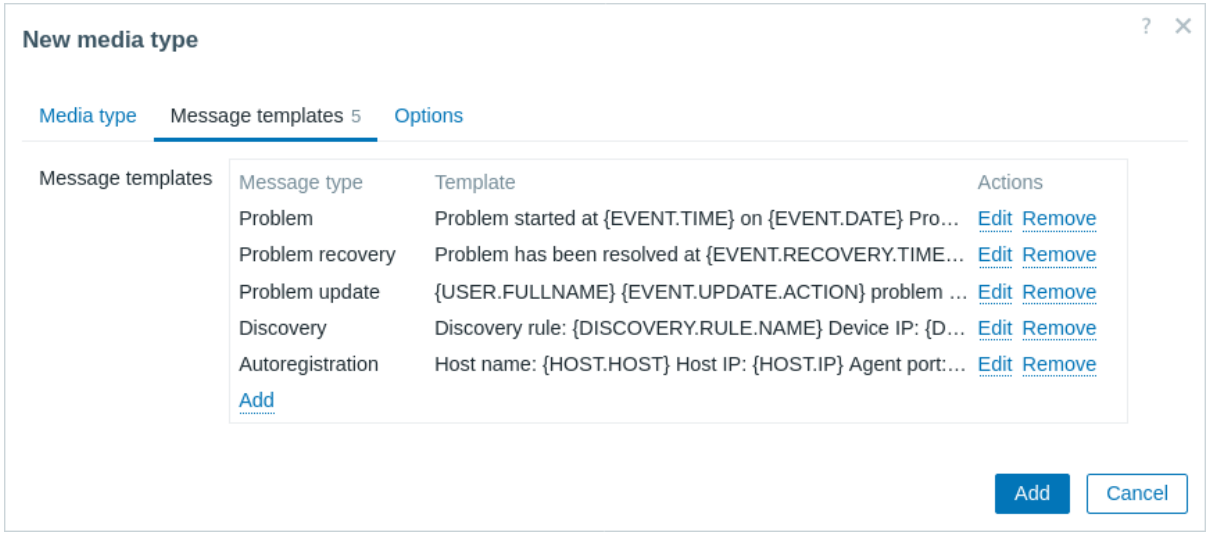


Parámetro	Descripción
Nombre	Nombre del tipo de medio.
Tipo	Seleccione el método de entrega para el tipo de medio.
Descripción	Introduzca una descripción para el tipo de medio.
Habilitado	Marque la casilla para habilitar el tipo de medio.

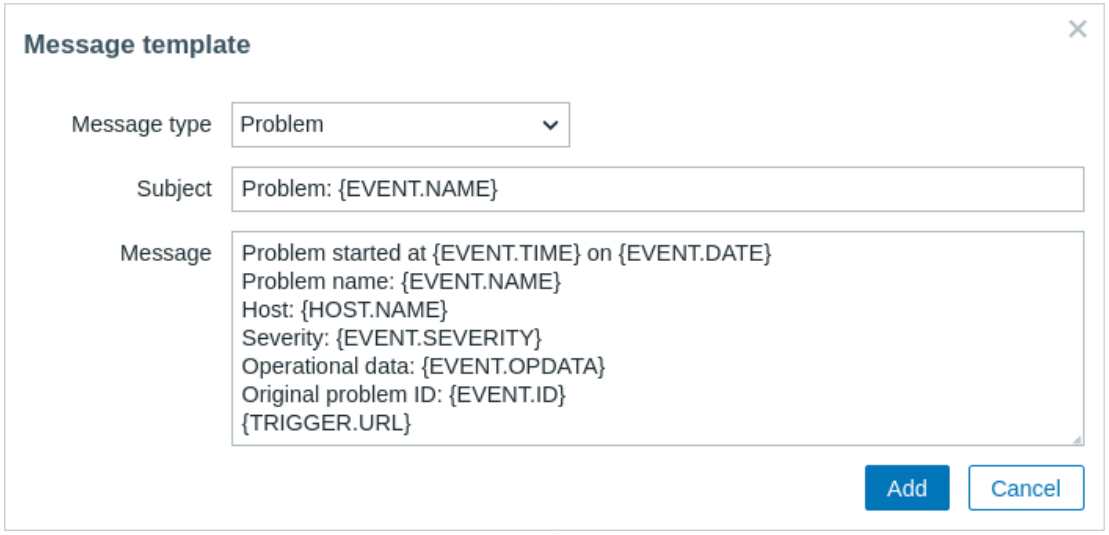
Para consultar los parámetros específicos de cada método, consulte las páginas de correo electrónico, notificación push, SMS, alertscript personalizado o webhook.

Plantillas de mensaje

La pestaña **Plantillas de mensaje** contiene mensajes predeterminados por tipo de evento (problema, recuperación de problema, descubrimiento, etc.).



Haga clic en **Añadir** para definir un mensaje predeterminado (o en **Editar** para actualizar un mensaje existente):



Parameter	Description
Message type	Tipo de evento para el que se debe usar el mensaje predeterminado. Solo se puede definir un mensaje predeterminado para cada tipo de evento.
Subject	Asunto del mensaje predeterminado. El asunto puede contener macros. Está limitado a 255 caracteres. El asunto no está disponible para el tipo de medio SMS.
Message	El mensaje predeterminado. Está limitado a una cierta cantidad de caracteres según el tipo de base de datos (consulte Sending messages para obtener más información). El mensaje puede contener macros compatibles. En los mensajes de problema y actualización de problema, se admiten macros de expresión (por ejemplo, `{?avg(/host/key,1h)}`).

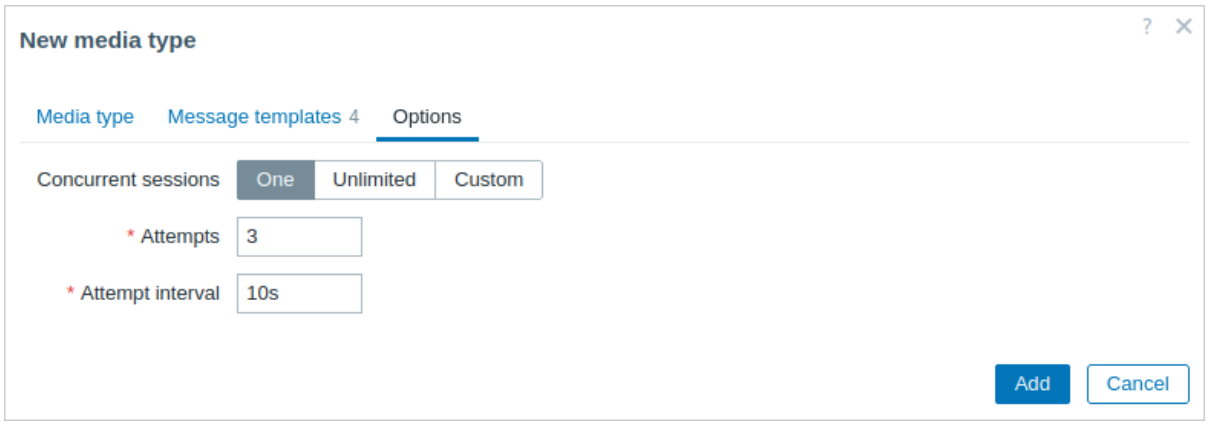
Tenga en cuenta que los mensajes predeterminados se sobrescriben con mensajes personalizados, si se definen en action operations.

Warning:
Definir plantillas de mensaje es obligatorio para todos los métodos de entrega, incluidos webhooks o scripts de alerta personalizados que no usan mensajes predeterminados para las notificaciones. Por ejemplo, la acción "Send message to Pushover webhook" no podrá enviar notificaciones de problema si no se define el mensaje de problema para el webhook de Pushover.

Opciones

La pestaña **Opciones** contiene la configuración del procesamiento de alertas.
El mismo conjunto de opciones se puede configurar para cada tipo de medio.

Todos los tipos de medio se procesan en paralelo.
Aunque el número máximo de sesiones simultáneas se puede configurar por tipo de medio,
el número total de procesos alerter en el server solo se puede limitar mediante el parámetro StartAlerters.
Las alertas generadas por un trigger se procesan secuencialmente.
Por lo tanto, varias notificaciones solo se pueden procesar simultáneamente si han sido generadas por varios triggers.



Parámetro	Descripción
Sesiones simultáneas	Seleccione el número de sesiones alerter paralelas para el tipo de medio: **Una**: una sesión **Ilimitadas**: número ilimitado de sesiones **Personalizado**: seleccione un número personalizado de sesiones Los valores ilimitados o altos significan más sesiones paralelas y una mayor capacidad para enviar notificaciones. Los valores ilimitados o altos deben utilizarse en entornos grandes donde sea necesario enviar muchas notificaciones simultáneamente. Si es necesario enviar más notificaciones que el número de sesiones simultáneas, las notificaciones restantes se pondrán en cola; no se perderán.
Intentos	Número de intentos para enviar una notificación. Se pueden especificar hasta 100 intentos; el valor predeterminado es '3'. Si se especifica '1', Zabbix enviará la notificación una sola vez y no volverá a intentarlo si el envío falla.
Intervalo entre intentos	Frecuencia con la que se intenta reenviar una notificación si el envío falla, en segundos (0-3600). Si se especifica '0', Zabbix volverá a intentarlo inmediatamente. Se admiten sufijos de tiempo (por ejemplo, 30s, 1m, 1h).

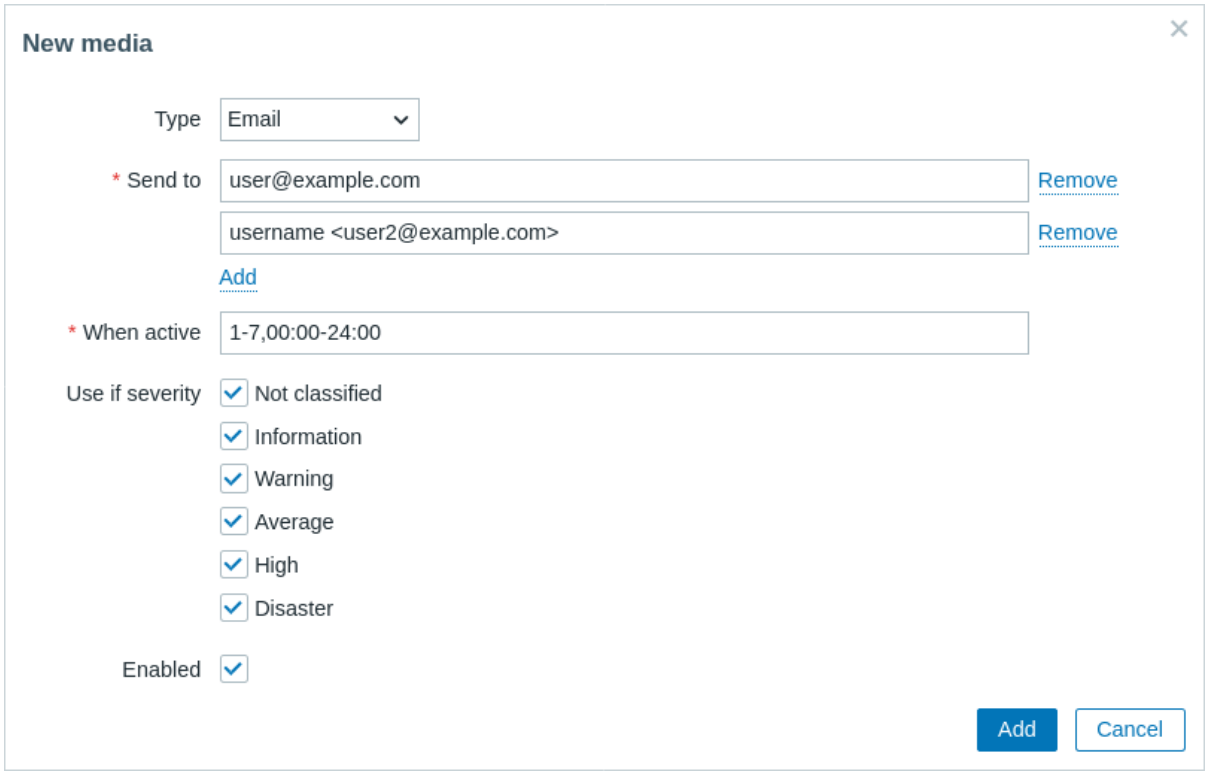
Medios del usuario

Mientras que los tipos de medios definen **cómo** se enviará una notificación, los medios del usuario definen **dónde** debe enviarse la notificación.

Los medios del usuario (por ejemplo, una dirección de correo electrónico, un ID de usuario de webhook, etc.) deben definirse en el perfil del usuario independientemente del método de entrega. Una acción que envíe mensajes al usuario *Admin* mediante el webhook *X* no podrá entregar los mensajes si los detalles de entrega del webhook *X* no están definidos en el perfil del usuario Admin.

Para definir los medios del usuario:

1. Vaya a *Usuarios* > *Usuarios* y abra el formulario de propiedades del usuario.
   Como alternativa, vaya a *Configuración del usuario* > *Notificaciones* en su propio perfil de usuario.
2. Haga clic en *Añadir* en la pestaña *Medios*.
3. Introduzca los parámetros de los medios del usuario en el formulario.



Parámetro	Descripción
Tipo	La lista desplegable contiene los nombres de los tipos de medios habilitados. Tenga en cuenta que, al editar un medio de un tipo de medio deshabilitado, el tipo se mostrará en rojo.

Parámetro	Descripción
Enviar a	Introduzca la información de contacto a la que deben enviarse los mensajes. Para el tipo de medio de correo electrónico, se pueden añadir varias direcciones haciendo clic en el botón Add situado debajo del campo de dirección. En este caso, las notificaciones se enviarán a todas las direcciones indicadas. Para consultar ejemplos de direcciones, vea la descripción del parámetro *Correo electrónico* del tipo de medio correo electrónico. Para el tipo de medio de notificaciones push, puede seleccionar los dispositivos móviles activos del usuario o añadir uno o varios dispositivos específicos.
Cuando está activo	Puede limitar el horario en el que se envían los mensajes; por ejemplo, establecer únicamente los días laborables (1-5,09:00-18:00). Tenga en cuenta que este límite se basa en la zona horaria del usuario. Si se cambia la zona horaria del usuario y esta difiere de la zona horaria del sistema, puede ser necesario ajustar este límite para no perder mensajes importantes. Consulte la página Especificación de períodos de tiempo para obtener una descripción del formato. Se admiten macros de usuario.
Usar si la gravedad es	Marque las casillas de las gravedades de trigger para las que desea recibir notificaciones. *Nota*: la gravedad predeterminada («No clasificado») **debe** estar marcada si desea recibir notificaciones de eventos que no sean de trigger. Después de guardar, las gravedades de trigger seleccionadas se mostrarán con los colores de gravedad correspondientes, mientras que las no seleccionadas aparecerán atenuadas.
Estado	Estado de los medios del usuario. **Habilitado**: está en uso. **Deshabilitado**: no se está utilizando.

Correo electrónico

Descripción general

Para configurar el correo electrónico como canal de entrega de mensajes, debe configurarlo como tipo de medio y asignar direcciones específicas a los usuarios.

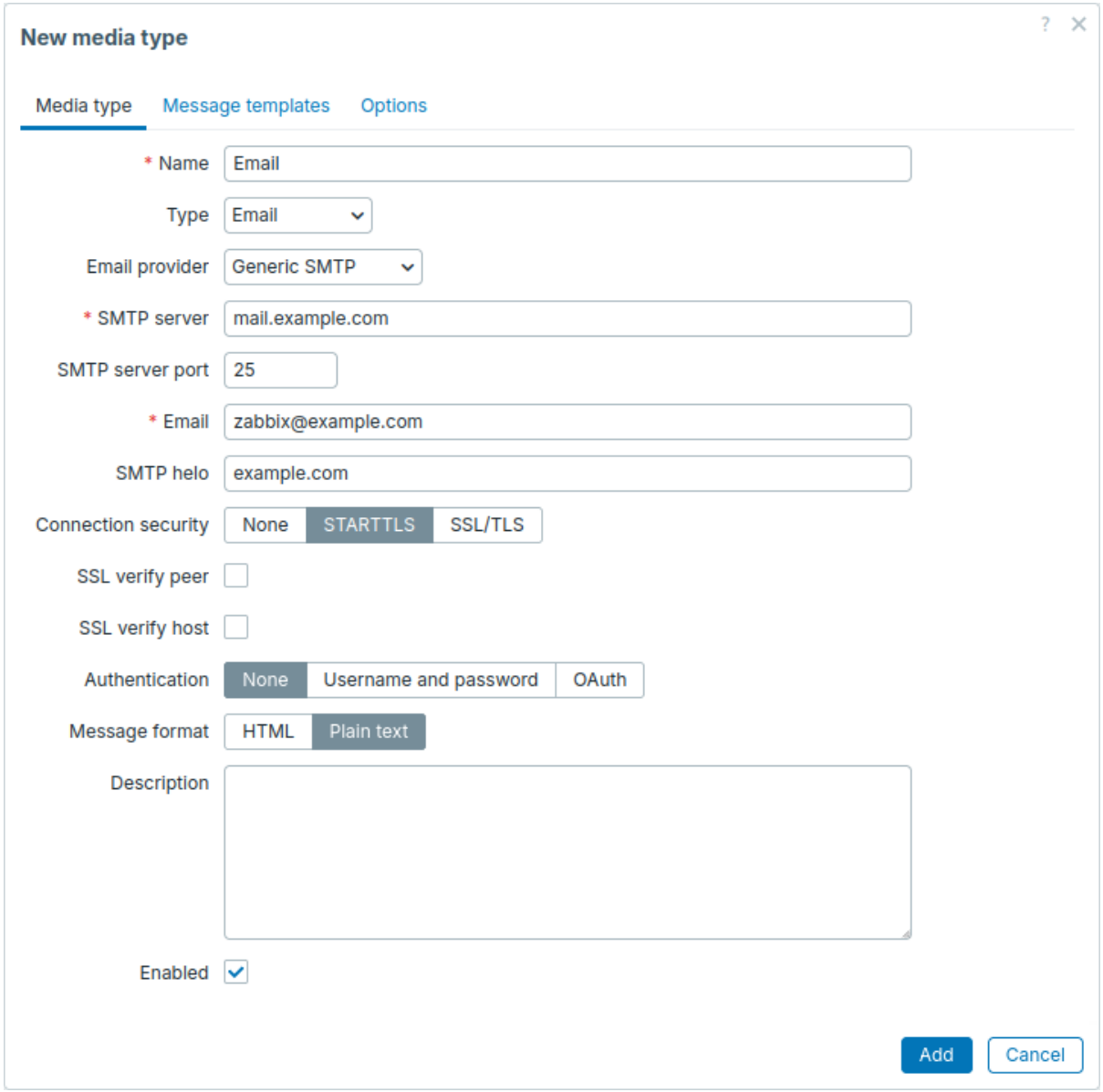
> **Note:**
> Varias notificaciones para un mismo evento se agruparán en el mismo hilo de correo electrónico.

Configuración

Para configurar el correo electrónico como tipo de medio:

1. Vaya a *Alertas* > *Tipos de medio*.
2. Haga clic en *Crear tipo de medio* (o haga clic en *Correo electrónico* en la lista de tipos de medio predefinidos).
3. Introduzca los parámetros del tipo de medio de correo electrónico en el formulario.

La pestaña **Tipo de medio** contiene los atributos generales del tipo de medio:



Todos los campos de entrada obligatorios están marcados con un asterisco rojo.

Warning:
La contraseña se exporta en texto sin formato al exportar la configuración del tipo de medio de correo electrónico.

Los siguientes parámetros son específicos del tipo de medio de correo electrónico:

Parámetro	Descripción
Proveedor de correo electrónico	Seleccione el proveedor de correo electrónico: *SMTP genérico*, *Gmail*, *Retransmisión de Gmail*, *Office365* o *Retransmisión de Office365*. Si selecciona las opciones relacionadas con Gmail/Office365, solo tendrá que proporcionar la dirección de correo electrónico del remitente y la contraseña; opciones como *Servidor SMTP*, *Puerto del servidor SMTP*, *SMTP helo* y *Seguridad de la conexión* se completarán automáticamente mediante Zabbix. Consulte también: Tipos de medio automatizados de Gmail/Office365.
Servidor SMTP	Establezca un servidor SMTP para gestionar los mensajes salientes. Este campo está disponible si se selecciona *SMTP genérico* como proveedor de correo electrónico.
Puerto del servidor SMTP	Establezca el puerto del servidor SMTP para gestionar los mensajes salientes. Este campo está disponible si se selecciona *SMTP genérico* como proveedor de correo electrónico.

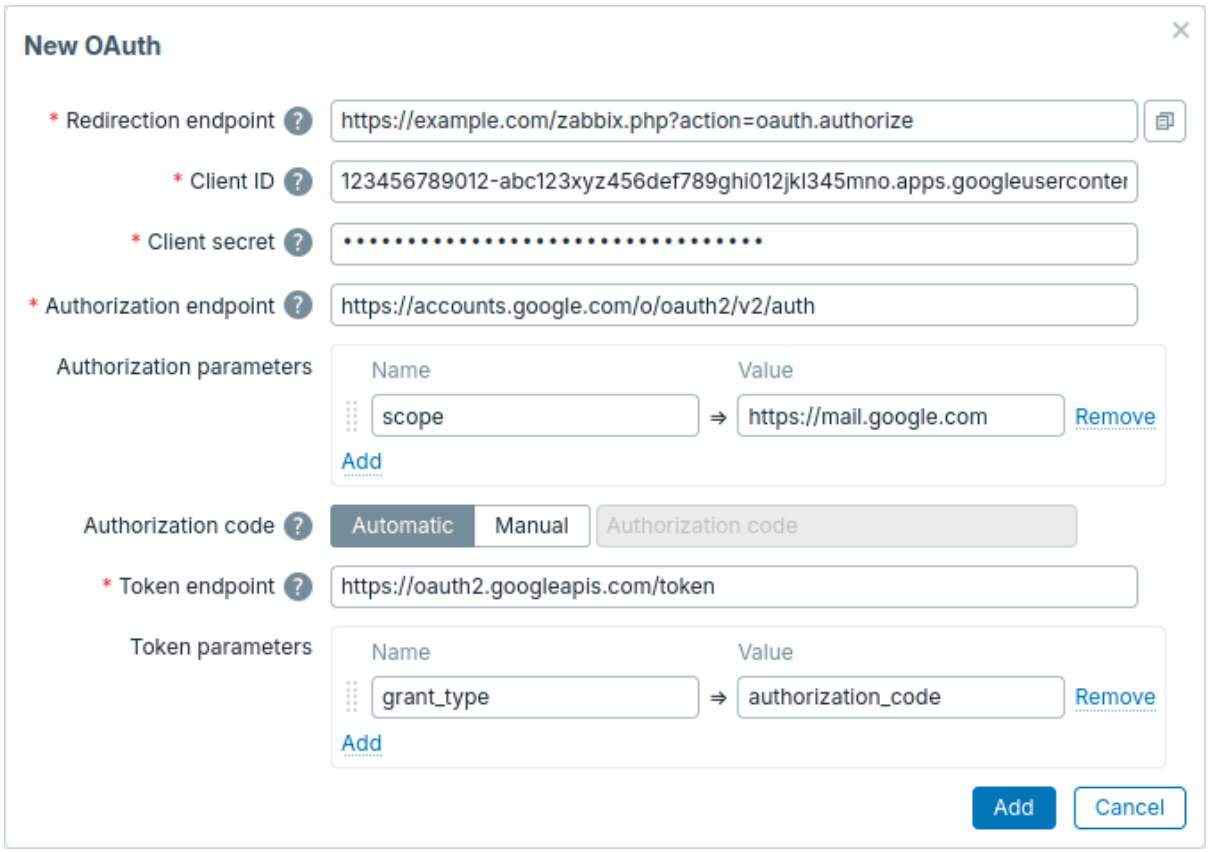
Parámetro	Descripción
Correo electrónico	La dirección introducida aquí se utilizará como dirección **De** para los mensajes enviados. Se admite añadir un nombre visible del remitente junto con la dirección de correo electrónico real (por ejemplo, `Zabbix_info <zabbix@company.com>`). Existen algunas restricciones para los nombres visibles en los correos electrónicos de Zabbix en comparación con los permitidos por RFC 5322, como se muestra en los siguientes ejemplos. Ejemplos válidos: `zabbix@company.com` (solo una dirección de correo electrónico; no se requieren corchetes angulares) `Zabbix_info <zabbix@company.com>` (un nombre visible seguido de la dirección de correo electrónico entre corchetes angulares) `Ω-monitoring <zabbix@company.com>` (caracteres UTF-8 en el nombre visible) Ejemplos no válidos: `Zabbix HQ zabbix@company.com` (hay un nombre visible, pero no hay corchetes angulares alrededor de la dirección de correo electrónico) `"Zabbix@<H(comment)Q>" <zabbix@company.com>` (aunque es válido según RFC 5322, los pares entre comillas y los comentarios no son compatibles con los correos electrónicos de Zabbix).
SMTP helo	Establezca un valor SMTP helo correcto, normalmente un nombre de dominio. Si está vacío, se enviará el nombre de dominio del correo electrónico (es decir, lo que aparece después de `@` en el campo *Correo electrónico*). Si no es posible obtener el nombre de dominio, se registrará una advertencia de nivel de depuración y se enviará el nombre de host del server como dominio para el comando HELO. Este campo está disponible si se selecciona *SMTP genérico* como proveedor de correo electrónico.
Seguridad de la conexión	Seleccione el nivel de seguridad de la conexión: **Ninguna**: no utilizar la opción CURLOPT_USE_SSL **STARTTLS**: utilizar la opción CURLOPT_USE_SSL con el valor CURLUSESSL_ALL **SSL/TLS**: el uso de CURLOPT_USE_SSL es opcional.
Verificar par SSL	Marque la casilla para verificar el certificado SSL del servidor SMTP. El valor de la directiva de configuración del server "SSLCALocation" debe incluirse en CURLOPT_CAPATH para la validación del certificado. Esto establece la opción de cURL CURLOPT_SSL_VERIFYPEER.
Verificar host SSL	Marque la casilla para verificar que el campo *Nombre común* o el campo *Nombre alternativo del sujeto* del certificado del server SMTP coincida. Esto establece la opción de cURL CURLOPT_SSL_VERIFYHOST.
Autenticación	Seleccione el nivel de autenticación: **Ninguna**: no se establecen opciones de cURL **Nombre de usuario y contraseña**: implica "AUTH=*", dejando que cURL elija el mecanismo de autenticación **OAuth**: autenticación OAuth La autenticación OAuth no es compatible con el proveedor de correo electrónico *Retransmisión de Office365*.
Nombre de usuario	Nombre de usuario que se utilizará en la autenticación. Esto establece el valor de CURLOPT_USERNAME. Se admiten macros de usuario.
Contraseña	Contraseña que se utilizará en la autenticación. Esto establece el valor de CURLOPT_PASSWORD. Se admiten macros de usuario. El valor introducido aquí se exporta tal cual al exportar la configuración del tipo de medio. Para evitar exponer credenciales confidenciales en los archivos exportados, utilice una macro de usuario (por ejemplo, `{$EMAIL_SMTP_PASSWORD}`) en lugar de una contraseña literal; se exportará la referencia a la macro, mientras que el valor secreto deberá configurarse o volver a configurarse en el sistema de destino.
Tokens OAuth	Haga clic en *Configurar* para establecer los parámetros para recuperar tokens OAuth en una nueva ventana. Este campo solo está disponible si se selecciona "OAuth" en el campo *Autenticación*.
Formato del mensaje	Seleccione el formato del mensaje: **HTML**: enviar como HTML; **Texto sin formato**: enviar como texto sin formato.

Attention:
Para habilitar la autenticación SMTP, el server de Zabbix debe compilarse con la opción de compilación `--with-libcurl`, que requiere libcurl (consulte los requisitos del server para obtener información sobre las versiones).

Consulte también los parámetros comunes de los tipos de medio para obtener información detallada sobre cómo configurar los mensajes predeterminados y las opciones de procesamiento de alertas.

Tokens OAuth

Los siguientes parámetros son obligatorios para recuperar tokens OAuth:



Warning:
Client secret se exporta en texto claro al exportar la configuración del tipo de medio de correo electrónico.

Parameter	Description
Redirection endpoint	Introduzca la URL del frontend de Zabbix a la que el servicio OAuth redirigirá de vuelta la autorización OAuth (usando el formato `https://<zabbix-frontend-url>/zabbix.php?action=oauth.authorize`). Se establece automáticamente para un nuevo tipo de medio de correo electrónico cuando se define la frontend URL.
Client ID	Introduzca el identificador único de la aplicación cliente registrada en el servidor de autorización OAuth.
Client secret	Introduzca el secreto privado de la aplicación cliente registrada en el servidor de autorización OAuth.
Authorization endpoint	Introduzca la URL del servidor de autorización OAuth para solicitar la autorización del usuario.
Authorization parameters	Introduzca los parámetros para el endpoint de autorización.
Authorization code	Introduzca el código de autorización: **Automatic** - el código se recuperará automáticamente mediante una página de redirección; **Manual** - introduzca el código manualmente si falla la recuperación automática.
Token endpoint	Introduzca la URL del servidor de autorización OAuth para intercambiar el código de autorización por tokens de acceso y actualización.
Token parameters	Introduzca los parámetros para el token de acceso.

Note:
La recuperación de tokens OAuth tiene funciones automatizadas para los proveedores de correo electrónico **Gmail**, **Gmail relay** y **Office365**. Solo es necesario proporcionar los valores de los parámetros *Redirection endpoint*, *Client ID* y *Client secret*. Zabbix completará automáticamente los demás valores requeridos (consulte Valores predeterminados de URL OAuth por proveedor).

Obtención de tokens de acceso y actualización OAuth

Después de enviar los parámetros OAuth:

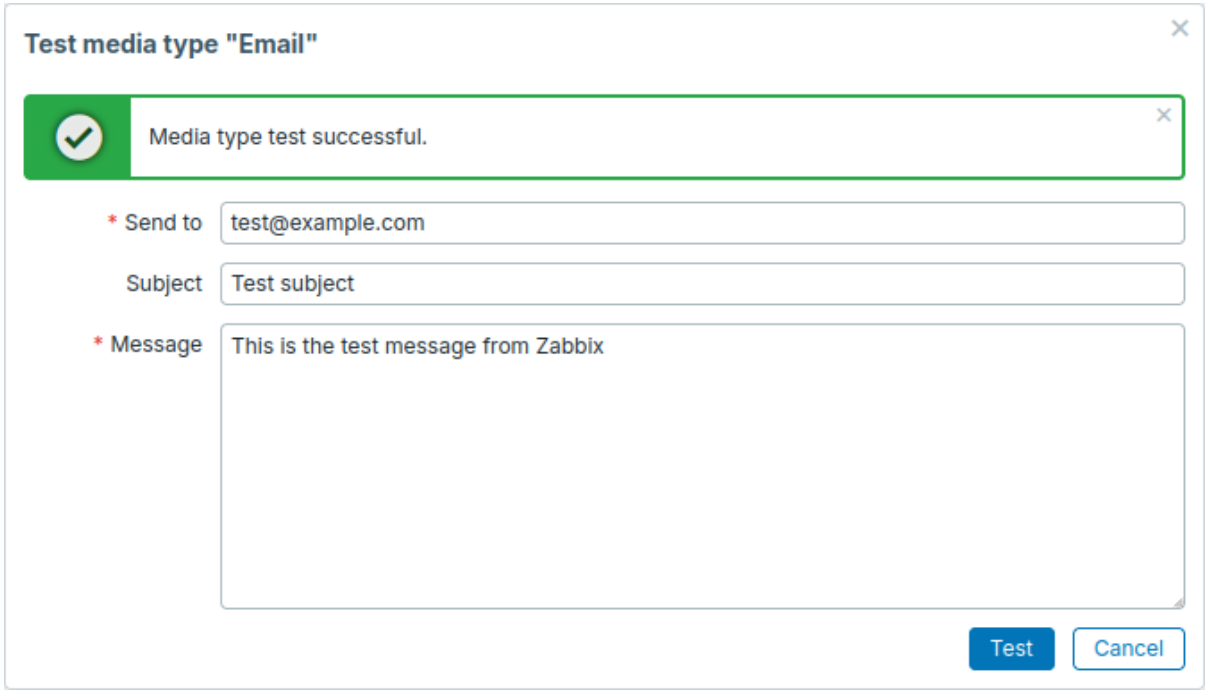
1. Se abre una ventana emergente del navegador que dirige al usuario al *Punto final de autorización*.
2. El usuario autoriza a Zabbix en el servicio OAuth.
3. El servicio OAuth redirige al usuario a la acción del frontend de Zabbix `oauth.authorize` con el código de autorización y el valor del alcance.
4. En respuesta, Zabbix realizará una solicitud al *Punto final de token* para intercambiar el código de autorización por los tokens de acceso y actualización.

Pruebas

Para comprobar si un tipo de medio de correo electrónico configurado funciona correctamente:

1. Localice el correo electrónico correspondiente en la lista de tipos de medio.
2. Haga clic en *Test* en la última columna de la lista (se abrirá una ventana de prueba).
3. Introduzca una dirección de destinatario en *Send to*, el cuerpo del mensaje y, opcionalmente, el asunto.
4. Haga clic en *Test* para enviar un mensaje de prueba.

En la misma ventana se mostrará un mensaje de éxito o de error de la prueba:



Medios de usuario

Una vez configurado el tipo de medio de correo electrónico, vaya a la sección *Users* > *Users* y edite el perfil del usuario para asignarle el medio de correo electrónico. Los pasos para configurar los medios de usuario, que son comunes para todos los tipos de medios, se describen en la página Media types.

Tipos de medio automatizados de Gmail/Office365

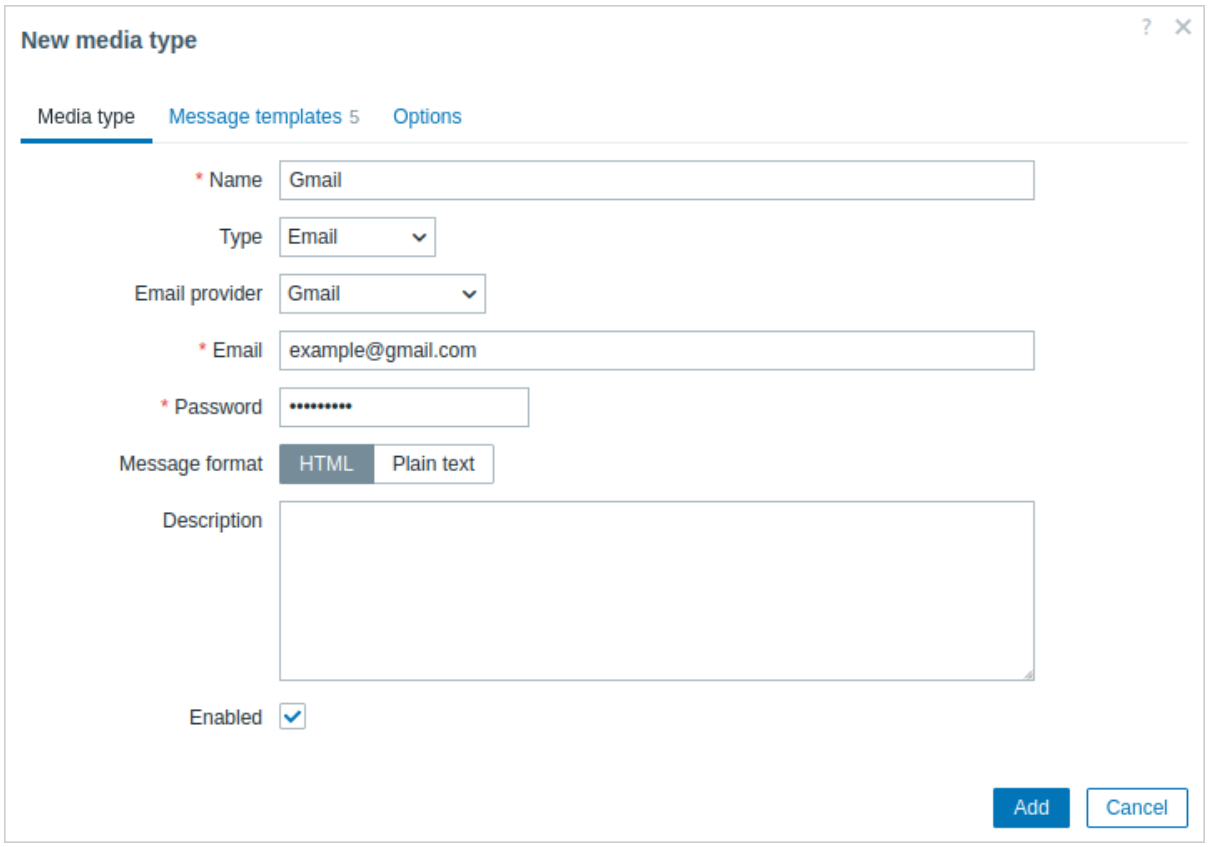
Descripción general

Los usuarios de Gmail u Office365 pueden beneficiarse de las funciones automatizadas en la configuración del tipo de medio.

Configuración

El campo *Proveedor de correo electrónico* en la configuración del tipo de medio de correo electrónico permite seleccionar opciones preconfiguradas para Gmail y Office 365.

Al seleccionar las opciones relacionadas con Gmail/Office365, solo es necesario proporcionar la dirección de correo electrónico y la contraseña del remitente para crear un tipo de medio funcional.



En cuanto se proporcionan la dirección de correo electrónico y la contraseña, Zabbix puede completar automáticamente todos los ajustes necesarios para los tipos de medio de Gmail/Office365 con los valores reales o recomendados, es decir, *server SMTP*, *puerto del server SMTP*, *SMTP helo* y *seguridad de la conexión*. Debido a esta automatización, estos campos ni siquiera se muestran; sin embargo, es posible consultar el server SMTP y los detalles del correo electrónico en la lista de tipos de medio (consulte la columna *Detalles*).

Tenga en cuenta también lo siguiente:

- La contraseña no es necesaria para las opciones de retransmisión.
- Para la retransmisión de Office365, el nombre de dominio de la dirección de correo electrónico proporcionada se utilizará para completar dinámicamente el server SMTP (es decir, reemplazar "`example.com`" en `example-com.mail.protection.outlook.com` por el valor real).

Tokens OAuth

La obtención de tokens OAuth tiene funciones automatizadas para los proveedores de correo electrónico *Gmail*, *Gmail relay* y *Office365*.



Solo es necesario proporcionar un conjunto limitado de parámetros para la obtención del token de acceso y actualización de OAuth: *Redirection endpoint*, *Client ID* y *Client secret*. Zabbix completará automáticamente los otros valores requeridos (consulte Valores predeterminados de URL de OAuth por proveedor).

Tenga en cuenta que también es posible utilizar el formulario SMTP genérico para configurar la autorización OAuth para estos proveedores.

Note:
SmtpClientAuthentication debe estar habilitado por usuario/buzón compartido o para todo el tenant en Office365 (no está habilitado por defecto).

Valores predeterminados de URL de OAuth por proveedor

La siguiente tabla enumera los valores predeterminados de URL de OAuth y los parámetros por proveedor que utiliza Zabbix.

Parámetro	Gmail	Office365	SMTP genérico
Punto final de autorización	`https://accounts.google.com/o/oauth2/auth`	`https://login.microsoftonline.com/common/oauth2/v2.0/authori`	Sin valor predeterminado
scope	`https://mail.google.com`	`https://outlook.office.com/SMTP.Send offline_access`	Sin valor predeterminado
access_type	`offline`	No se utiliza	Sin valor predeterminado
prompt	`consent`	No se utiliza	Sin valor predeterminado
redirect_uri	El valor del campo *Punto final de redirección* se añade automáticamente por código. No se muestra en los parámetros de *Punto final de autorización*.		
state	El código añade automáticamente un hash único para identificar el medio que se está actualizando con el token OAuth. No se muestra en los parámetros de *Punto final de autorización*.		
Punto final de token	`https://oauth2.googleapis.com/token`	`https://login.microsoftonline.com/common/oauth2/v2.0/token`	Sin valor predeterminado
grant_type	`authorization_code`	`authorization_code`	Sin valor predeterminado
redirect_uri	El valor del campo *Punto final de redirección* se añade automáticamente por código. No se muestra en los parámetros de *Punto final de token*.		
state	El código añade automáticamente un hash único para identificar el medio que se está actualizando con el token OAuth. No se muestra en los parámetros de *Punto final de token*.		

Solución alternativa para cuentas personales de Office365

Las cuentas personales de Office365 no admiten cadenas de consulta en la URL de redirección.

Para solucionar este problema, el administrador del servidor web Apache puede agregar la siguiente regla de reescritura al archivo `.htaccess`:

```
RewriteEngine On
RewriteRule ^/oauth\.authorize$ /zabbix.php?action=oauth.authorize [QSA,L,PT]
```

Esto permite el uso de una URL de redirección simplificada como:

```
http://nombre-del-servidor/zabbix/oauth.authorize
```

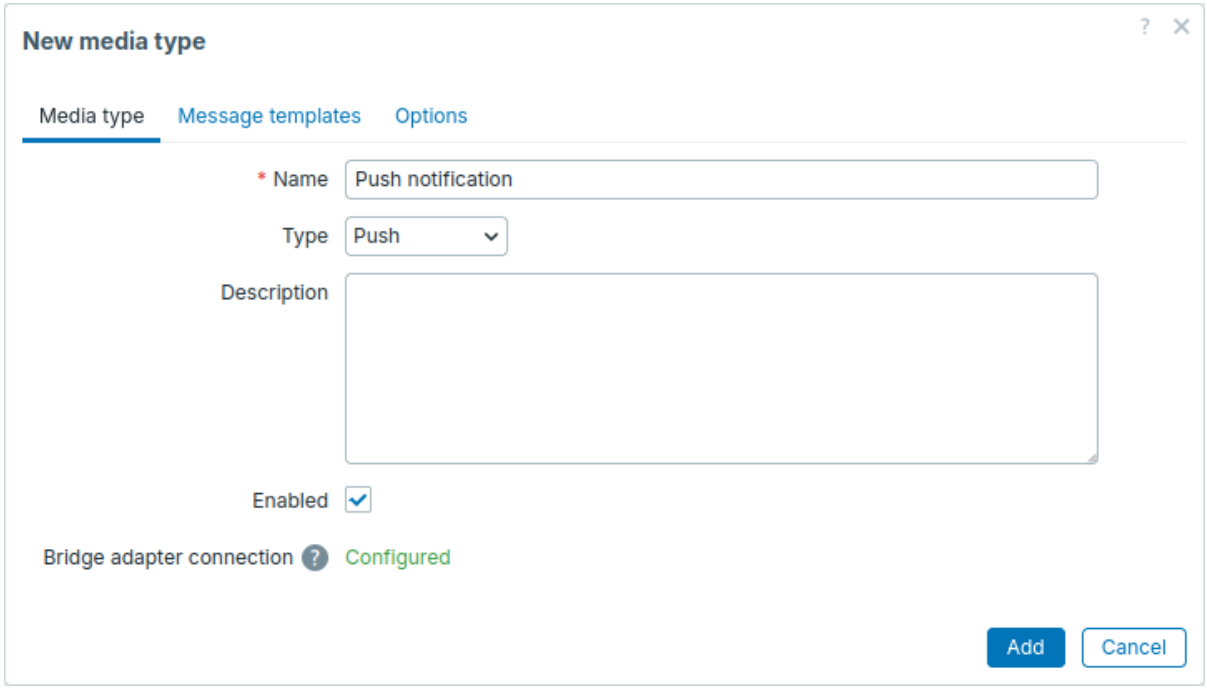
en lugar de una URL con parámetros de consulta, asegurando la compatibilidad con cuentas personales de Office365.

Notificación push

Este tipo de medio permite que el server de Zabbix envíe notificaciones push a la aplicación móvil de Zabbix en su dispositivo móvil.

Configuración

Al crear un tipo de medio, seleccione *Push* como su *Type*:



El formulario de configuración contiene parámetros, plantillas de mensajes y opciones comunes del tipo de medio.

El formulario también muestra si el adaptador de Zabbix bridge está configurado.
 Si el adaptador no está configurado, aún puede crear el tipo de medio, pero no funcionará.

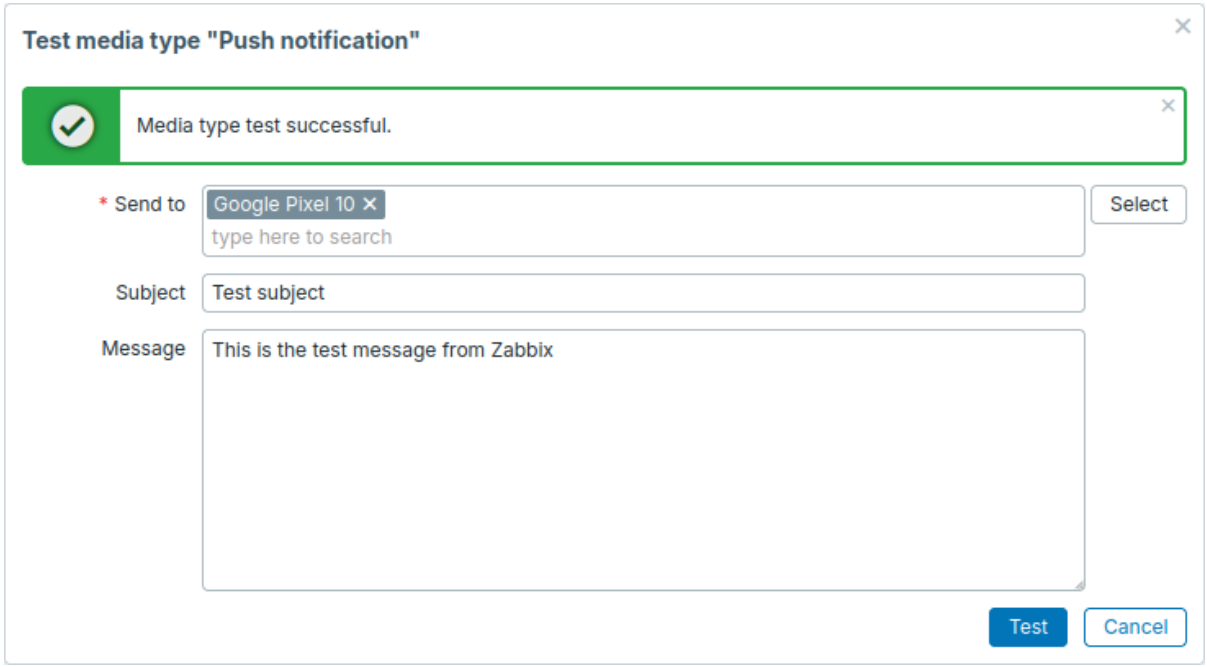
Después de crear y habilitar el tipo de medio, añada un dispositivo para un usuario a fin de agregar automáticamente el tipo de medio a ese usuario (o añada medios de usuario manualmente). Esto le permite recibir notificaciones push en su dispositivo móvil.

Pruebas

Para comprobar si este tipo de medio funciona:

1. Busque el tipo de medio en la lista de tipos de medios.
2. Haga clic en *Probar* en la última columna de la lista para abrir el formulario de prueba.
3. En el campo *Enviar a*, seleccione un dispositivo.
4. Haga clic en *Probar* para enviar una notificación push de prueba al dispositivo seleccionado.

En la misma ventana aparecerá un mensaje de éxito o de error:



SMS

Resumen

Zabbix admite el envío de mensajes SMS mediante un módem GSM serie conectado al puerto serie del server de Zabbix.

Asegúrese de que:

- La velocidad del dispositivo serie (normalmente /dev/ttyS0 en Linux) coincida con la del módem GSM. Zabbix no establece la velocidad del enlace serie. Utiliza la configuración predeterminada.
- El usuario 'zabbix' tenga acceso de lectura/escritura al dispositivo serie. Ejecute el comando ls -l /dev/ttyS0 para ver los permisos actuales del dispositivo serie.
- El módem GSM tenga el PIN introducido y lo conserve después de un reinicio de alimentación. Como alternativa, puede deshabilitar el PIN en la tarjeta SIM. El PIN puede introducirse emitiendo el comando AT+CPIN="NNNN" (NNNN es su número PIN, las comillas deben estar presentes) en un software de terminal, como minicom de Unix o HyperTerminal de Windows.

Zabbix ha sido probado con estos módems GSM:

- Siemens MC35
- Teltonika ModemCOM/G10

Para configurar SMS como canal de entrega de mensajes, también debe configurar SMS como tipo de medio e introducir los números de teléfono correspondientes para los usuarios.

Configuración

Para configurar SMS como tipo de medio:

1. Vaya a *Alertas* > *Tipos de medios*.
2. Haga clic en *Crear tipo de medio* (o haga clic en *SMS* en la lista de tipos de medios predefinidos).
3. Introduzca los parámetros del tipo de medio SMS en el formulario.

Los siguientes parámetros son específicos del tipo de medio SMS:

Parámetro	Descripción
Módem GSM	Establezca el nombre del dispositivo serie del módem GSM. La ruta introducida aquí se validará con el parámetro del server SMSDevices (si se especifica).

Consulte los parámetros comunes de los tipos de medios para obtener información detallada sobre cómo configurar los mensajes predeterminados y las opciones de procesamiento de alertas. Tenga en cuenta que no es posible procesar en paralelo el envío de notificaciones SMS.

Medios del usuario

Una vez configurado el tipo de medio SMS, vaya a la sección *Usuarios* > *Usuarios* y edite el perfil del usuario para asignarle el medio SMS. Los pasos para configurar los medios del usuario, que son comunes para todos los tipos de medios, se describen en la página Tipos de medios.

Scripts de alerta personalizados

Descripción general

Si no está satisfecho con los tipos de medios existentes para enviar alertas, existe una forma alternativa de hacerlo. Puede crear un script que gestione la notificación a su manera.

Los scripts de alertas personalizados se ejecutan en el server de Zabbix. Estos scripts deben estar ubicados en el directorio especificado en el parámetro AlertScriptsPath del archivo de configuración del server.

A continuación se muestra un ejemplo de script de alerta personalizado:

```
#####!/bin/bash

to=$1
subject=$2
body=$3
host=$4
value=$5

cat <<EOF | mail -s "$subject" "$to"
$body

Host: $host
Value: $value
EOF
```

Attention:
Zabbix comprueba el código de salida de los comandos y scripts ejecutados. Cualquier código de salida distinto de **0** se considera un error de ejecución de comandos. En tales casos, Zabbix intentará repetir la ejecución fallida.

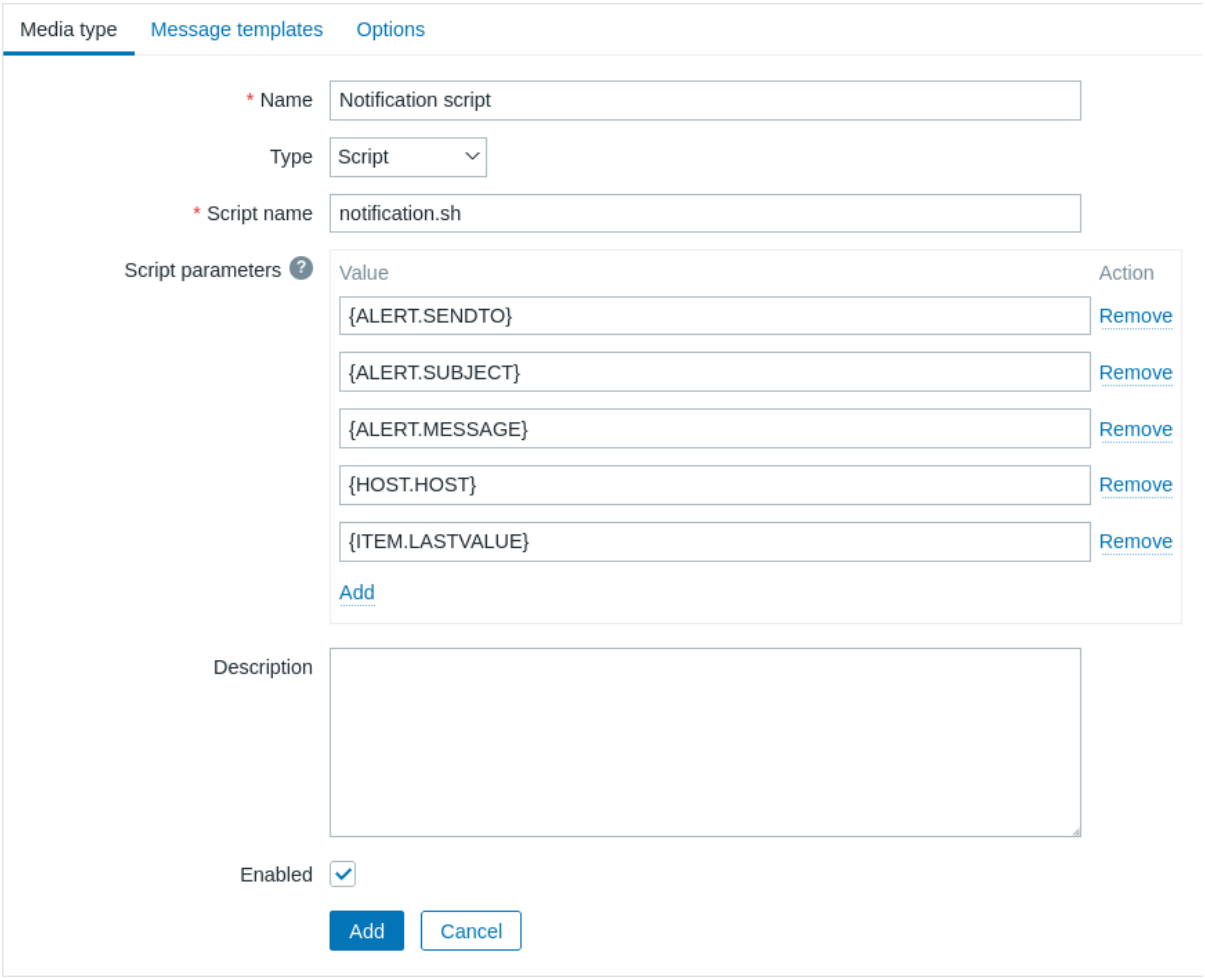
Las variables de entorno no se conservan ni se crean para el script, por lo que deben gestionarse explícitamente.

Configuración

Para configurar scripts de alerta personalizados como un tipo de medio:

1. Vaya a *Alertas* > *Tipos de medio*.
2. Haga clic en *Crear tipo de medio*.
3. Introduzca los parámetros del tipo de medio de scripts en el formulario.

La pestaña **Tipo de medio** contiene los atributos generales del tipo de medio:



Todos los campos de entrada obligatorios están marcados con un asterisco rojo.

Los siguientes parámetros son específicos del tipo de medio de script:

Parámetro	Descripción
Nombre del script	Introduzca el nombre del archivo de script (por ejemplo, notification.sh) que se encuentra en el directorio especificado en el parámetro de configuración del server AlertScriptsPath.
Parámetros del script	Añada parámetros opcionales del script que se pasarán al script como argumentos de la línea de comandos, en el orden en que se definan. Los parámetros del script admiten las macros {ALERT.SENDTO}, {ALERT.SUBJECT}, {ALERT.MESSAGE} y todas las macros compatibles con las notificaciones, así como las macros de usuario.

Consulte parámetros comunes de los tipos de medio para obtener información detallada sobre cómo configurar los mensajes predeterminados y las opciones de procesamiento de alertas.

Warning:
Aunque un script de alerta no utilice mensajes predeterminados, las plantillas de mensajes para los tipos de operación utilizados por este tipo de medio deben estar definidas. De lo contrario, no se enviará ninguna notificación.

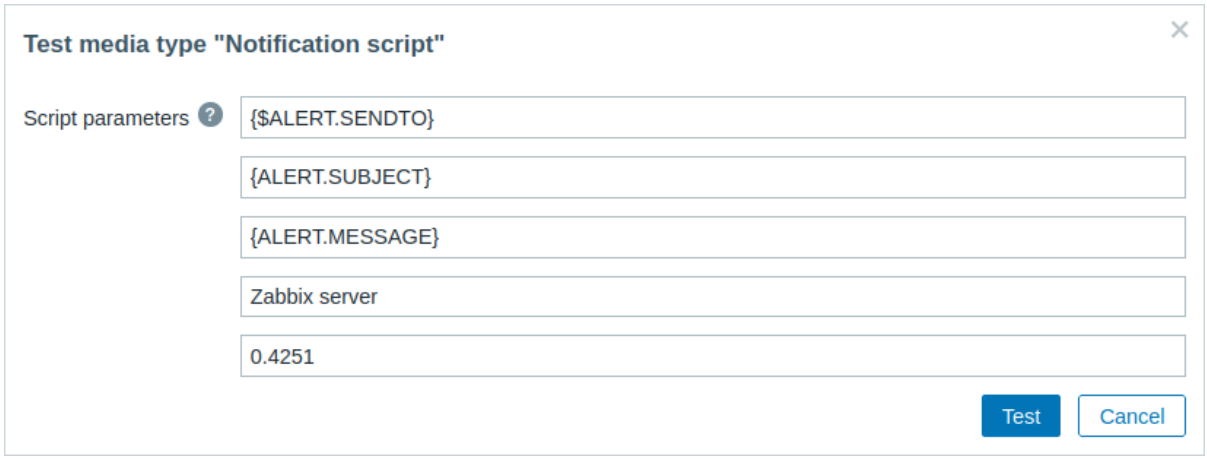
Attention:
Si se configura más de un tipo de medio de script, estos scripts pueden ser procesados en paralelo por los procesos alerter. El número total de procesos alerter está limitado por el parámetro `StartAlerters` del archivo de configuración del server.

Prueba

Para probar un tipo de medio de script configurado:

1. Localice el script correspondiente en la lista de tipos de medio.

2. Haga clic en *Test* en la última columna de la lista; se abrirá un formulario de prueba en una ventana emergente. El formulario de prueba contendrá el mismo número de parámetros que están configurados para el tipo de medio de script.
3. Edite los valores de los parámetros del script si es necesario. La edición solo afecta al procedimiento de prueba; los valores reales no se cambiarán.
4. Haga clic en *Test*.



Note:
Al probar un tipo de medio de script configurado, {ALERT.SENDTO}, {ALERT.SUBJECT}, {ALERT.MESSAGE} y los macros de usuario se resolverán con sus valores, pero los macros relacionados con eventos (por ejemplo, {HOST.HOST}, {ITEM.LASTVALUE}, etc.) no se resolverán, ya que durante la prueba no hay un evento relacionado del que obtener los detalles. Tenga en cuenta que los macros dentro de {ALERT.SUBJECT} y {ALERT.MESSAGE} tampoco se resolverán. Por ejemplo, si el valor de {ALERT.SUBJECT} está compuesto por "Problem: {EVENT.NAME}", entonces el macro {EVENT.NAME} no se resolverá.

Medio del usuario

Una vez configurado el tipo de medio, vaya a la sección *Users* > *Users* y edite el perfil de un usuario asignando este tipo de medio al usuario. Los pasos para configurar el medio del usuario, que son comunes para todos los tipos de medio, se describen en la página Media types.

Tenga en cuenta que, al definir el medio del usuario, el campo *Send to* no puede estar vacío. Si este campo no se utiliza en el script de alerta, introduzca cualquier combinación de caracteres admitidos para omitir los requisitos de validación.

webhook

Descripción general

El tipo de medio webhook es útil para realizar llamadas HTTP utilizando código JavaScript personalizado para una integración sencilla con software externo como sistemas de helpdesk, chats o mensajería. Puede optar por importar una integración proporcionada por Zabbix o crear una integración personalizada desde cero.

Integraciones

Las siguientes integraciones están disponibles y permiten utilizar tipos de medio webhook predefinidos para enviar notificaciones de Zabbix a:

- brevis.one
- Discord
- Event-Driven Ansible
- Express.ms messenger
- GitHub
- GLPI
- IBM Maximo Service Request
- iLert
- iTop
- Jira
- Jira Service Management
- ManageEngine ServiceDesk
- Mantis Bug Tracker

- Mattermost
- MS Teams Workflows
- LINE
- Opsgenie
- OTRS CE
- Pagerduty
- Pushover
- Redmine
- Rocket.Chat
- ServiceNow
- SIGNL4
- Slack
- SolarWinds
- SysAid
- Telegram
- TOPdesk
- VictorOps
- Zammad
- Zendesk

Note:
Además de los servicios enumerados aquí, Zabbix puede integrarse con **Spiceworks** (no se requiere ningún webhook). Para convertir las notificaciones de Zabbix en tickets de Spiceworks, cree un tipo de medio de correo electrónico e introduzca la dirección de correo electrónico del servicio de asistencia de Spiceworks (por ejemplo, help@zabbix.on.spiceworks.com) en la configuración del perfil de un usuario de Zabbix designado.

Configuración

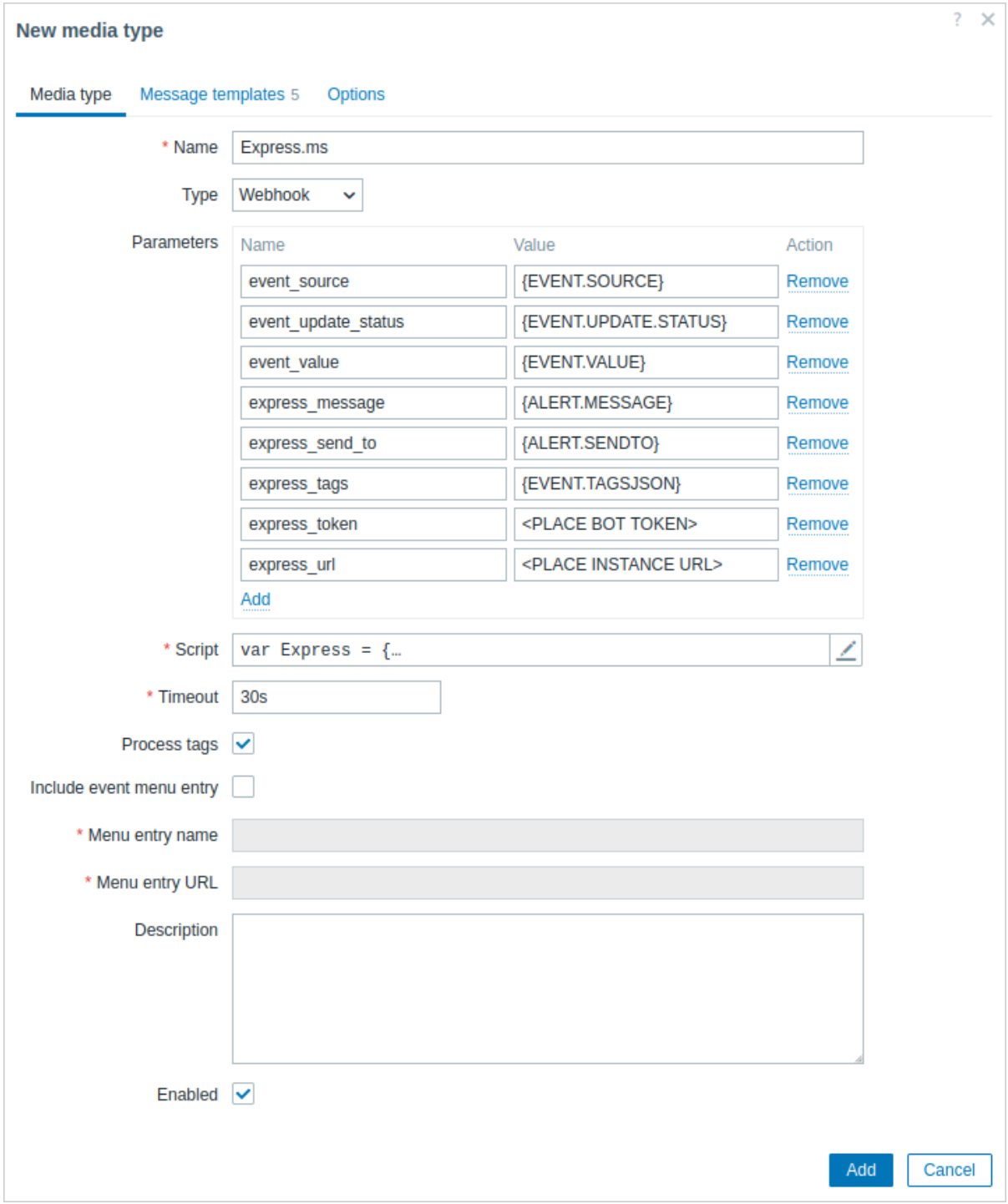
Para comenzar a utilizar una integración de webhook:

1. Localice el archivo `.yaml` requerido en el directorio `templates/media` de la versión de Zabbix descargada o descárguelo del repositorio git de Zabbix.
2. Importe el archivo en su instalación de Zabbix. El webhook aparecerá en la lista de tipos de medios.
3. Configure el webhook según las instrucciones del archivo `Readme.md` (puede hacer clic en el nombre de un webhook más arriba para acceder rápidamente a `Readme.md`).

Para crear un webhook personalizado desde cero:

1. Vaya a *Alertas* > *Tipos de medios*.
2. Haga clic en *Crear tipo de medio*.
3. Introduzca los parámetros del tipo de medio webhook en el formulario.

La pestaña **Tipo de medio** contiene varios atributos específicos de este tipo de medio:



Todos los campos de entrada obligatorios están marcados con un asterisco rojo.

Los siguientes parámetros son específicos del tipo de medio webhook:

Parámetro	Descripción
Parámetros	Especifique las variables del webhook como pares de atributo y valor. En los webhooks preconfigurados, la lista de parámetros varía según el servicio. Consulte el archivo *Readme.md* del webhook para obtener la descripción de los parámetros. En los webhooks nuevos, se incluyen varias variables comunes de forma predeterminada (URL:<empty>, HTTPProxy:<empty>, To:{ALERT.SENDTO}, Subject:{ALERT.SUBJECT}, Message:{ALERT.MESSAGE}); puede conservarlas o eliminarlas. Los parámetros del webhook admiten macros de usuario, todas las macros compatibles con las notificaciones de problemas y, adicionalmente, las macros {ALERT.SENDTO}, {ALERT.SUBJECT} y {ALERT.MESSAGE}. Si especifica un proxy HTTP, el campo admite la misma funcionalidad que el campo proxy HTTP de la configuración del item. La cadena del proxy puede llevar el prefijo `[scheme]://` para especificar el tipo de proxy utilizado (por ejemplo, https, socks4, socks5; consulte la documentación).
Script	Introduzca el código JavaScript en el editor modal que se abre al hacer clic en el campo del parámetro o en el icono del lápiz situado junto a él. Este código realizará la operación del webhook. El script es un código de función que acepta pares de parámetro y valor. Los valores deben convertirse en objetos JSON mediante el método JSON.parse(), por ejemplo: `var params = JSON.parse(value);`. El código tiene acceso a todos los parámetros, puede realizar solicitudes HTTP GET, POST, PUT y DELETE, admitir métodos adicionales como CONNECT, PATCH, HEAD, OPTIONS y TRACE, y controlar las cabeceras HTTP y el cuerpo de la solicitud. El script debe contener un operador return; de lo contrario, no será válido. Puede devolver un estado OK junto con una lista opcional de etiquetas y valores de etiquetas (consulte la opción *Procesar etiquetas*) o una cadena de error. Los eventos de recuperación (generados automáticamente o como resultado de un cierre manual) son creados por el server e incluyen las etiquetas del evento resuelto (incluidas las etiquetas heredadas de templates, hosts y triggers). Los scripts de webhook se ejecutan después de crear la alerta; por lo tanto, las etiquetas devueltas por un script de webhook se añaden únicamente después de la creación inicial de la alerta y no estarán presentes en las macros `{EVENT.TAGS}` y `{EVENT.RECOVERY.TAGS}` del mensaje inicial del problema ni del mensaje de recuperación inmediato. *Nota*: Se recomienda utilizar variables locales (por ejemplo, `var local = 1`) en lugar de variables globales (por ejemplo, `global = 1`) para garantizar que cada script opere con sus propios datos y evitar colisiones entre llamadas simultáneas (consulte los problemas conocidos). Consulte también: Directrices para el desarrollo de webhooks, Ejemplos de scripts de webhook, Objetos JavaScript adicionales.
Tiempo de espera	Tiempo de espera de ejecución de JavaScript (1-60 s, valor predeterminado: 30 s). Se admiten sufijos de tiempo (por ejemplo, `30s`, `1m`).
Procesar etiquetas	Marque la casilla para procesar los valores de las propiedades JSON devueltas como etiquetas. Estas etiquetas se añaden a las etiquetas de problemas existentes. Tenga en cuenta que, al utilizar etiquetas de webhook, el webhook debe devolver un objeto JSON que contenga al menos un objeto de etiquetas vacío: `var result = {tags: {}};` Ejemplos de etiquetas que se pueden devolver: *jira-id:prod-1234*, *responsible:John Smith*, *processed:<no value>*
Incluir entrada en el menú de eventos	Marque la casilla para incluir una entrada en el menú de eventos que enlace con un ticket externo creado. Se incluirá una entrada para cada webhook que esté habilitado y tenga marcada esta casilla. Tenga en cuenta que, si los parámetros *Nombre de la entrada del menú* y *URL de la entrada del menú* contienen macros {EVENT.TAGS.<tag name>}, se incluirá una entrada únicamente si estas macros se pueden resolver (es decir, si el evento tiene definidas estas etiquetas). Si se marca, el webhook no debe utilizarse para enviar notificaciones a distintos usuarios (considere la posibilidad de crear un usuario dedicado) y no debe utilizarse en varias acciones de alerta para un único evento de problema.

Parámetro	Descripción
Nombre de la entrada del menú	Especifique el nombre de la entrada del menú. Se admite la macro {EVENT.TAGS.<tag name>}. Este campo solo es obligatorio si se marca *Incluir entrada en el menú de eventos*.
URL de la entrada del menú	Especifique la URL subyacente de la entrada del menú. Se admite la macro {EVENT.TAGS.<tag name>}. Este campo solo es obligatorio si se marca *Incluir entrada en el menú de eventos*.

Consulte parámetros comunes de los tipos de medios para obtener información detallada sobre cómo configurar los mensajes predeterminados y las opciones de procesamiento de alertas.

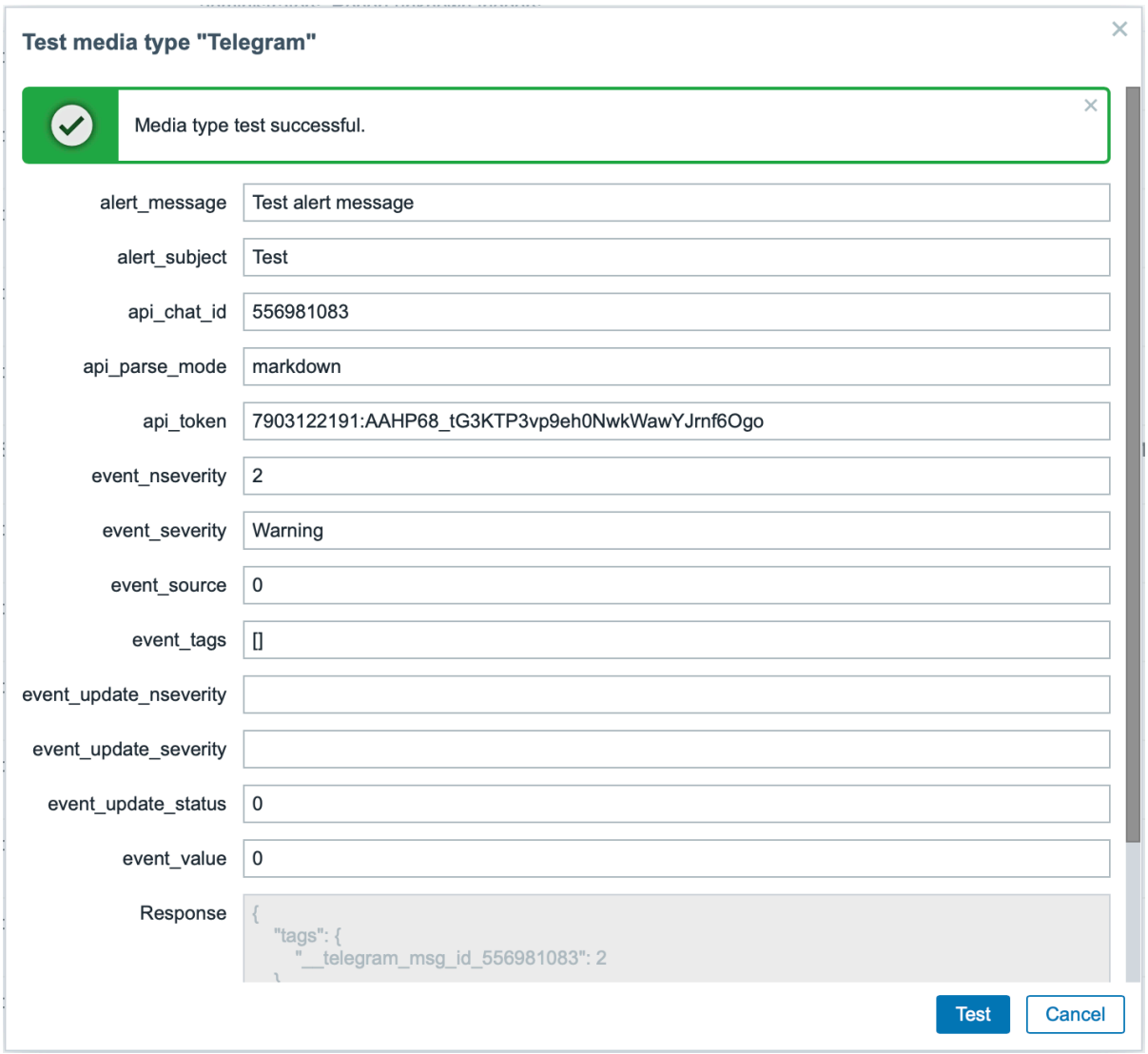
Warning:
Aunque un webhook no utilice mensajes predeterminados, las plantillas de mensajes para los tipos de operación utilizados por este webhook deben seguir estando definidas.

Pruebas

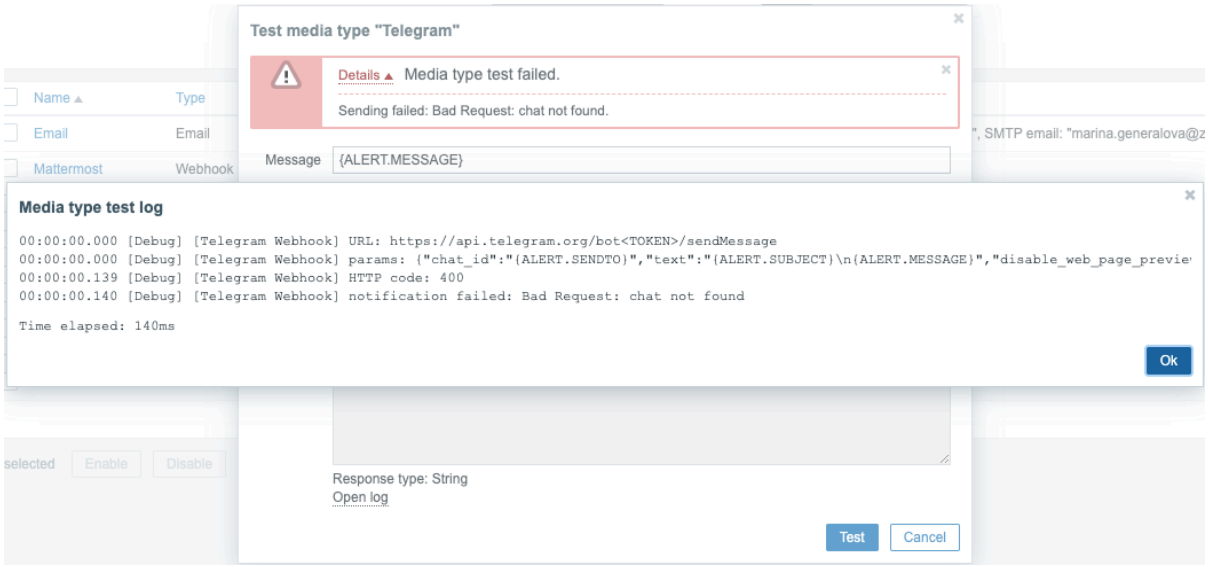
Para probar un tipo de medio webhook configurado:

1. Localice el webhook correspondiente en la lista de tipos de medio.
2. Haga clic en *Probar* en la última columna de la lista (se abrirá una ventana de prueba).
3. Edite los valores de los parámetros del webhook según sea necesario.
   Reemplace las macros por valores de ejemplo; de lo contrario, las macros no se resolverán y la prueba fallará.
4. Haga clic en *Probar*.

Reemplazar o eliminar valores en la ventana de prueba solo afecta al procedimiento de prueba; los valores reales de los atributos del webhook permanecerán sin cambios.



Para ver las entradas del registro de pruebas del tipo de medio sin salir de la ventana de prueba, haga clic en *Abrir registro* (se abrirá una nueva ventana emergente).



Si la prueba del webhook es exitosa:

- Se muestra el mensaje *"Prueba de tipo de medio exitosa."*.
- La respuesta del servidor aparece en el campo gris *Respuesta*.
- El tipo de respuesta (JSON o Cadena) se especifica debajo del campo *Respuesta*.

Si la prueba del webhook falla:

- Se muestra el mensaje *"La prueba del tipo de medio falló."*, seguido de detalles adicionales sobre el fallo.

Medios del usuario

Una vez configurado el tipo de medio, vaya a la sección *Usuarios* > *Usuarios* y asigne el medio webhook a un usuario existente o cree un usuario nuevo que represente al webhook. Los pasos para configurar los medios de usuario para un usuario existente, que son comunes a todos los tipos de medios, se describen en la página Tipos de medios.

Si un webhook utiliza etiquetas para almacenar el ID del ticket\mensaje, evite asignar el mismo webhook como medio a diferentes usuarios, ya que esto puede provocar errores del webhook (se aplica a la mayoría de los webhooks que utilizan la opción *Incluir entrada en el menú de eventos*). En este caso, la práctica recomendada es crear un usuario dedicado que represente al webhook:

1. Después de configurar el tipo de medio webhook, vaya a la sección *Usuarios* > *Usuarios* y cree un usuario de Zabbix dedicado para representar al webhook; por ejemplo, con el nombre de usuario *Slack* para el webhook de Slack. Todos los ajustes, excepto los medios, pueden dejarse con sus valores predeterminados, ya que este usuario no iniciará sesión en Zabbix.
2. En el perfil del usuario, vaya a la pestaña *Medios* y añada un webhook con la información de contacto necesaria. Si el webhook no utiliza un campo *Enviar a*, introduzca cualquier combinación de caracteres compatibles para omitir los requisitos de validación.
3. Conceda a este usuario al menos permisos de lectura para todos los hosts para los que deba enviar alertas.

Al configurar la acción de alerta, añada este usuario en el campo *Enviar a usuarios* de los detalles de la operación; esto indicará a Zabbix que utilice el webhook para las notificaciones de esta acción.

Configuración de acciones de alerta

Las acciones determinan qué notificaciones deben enviarse a través del webhook. Los pasos para configurar acciones que implican webhooks son los mismos que para todos los demás tipos de medios, con estas excepciones:

- Si un webhook usa etiquetas de webhook para almacenar el ID de ticket\mensaje y gestionar las operaciones de actualización\resolución, evite usar el mismo webhook en varias acciones de alerta para un único evento de problema. Si {EVENT.TAGS.<tag name>} existe y se actualiza en el webhook, su valor resultante no estará definido. Para evitar esto, use un nuevo nombre de etiqueta en el webhook para almacenar los valores actualizados. Esto se aplica a los webhooks de Jira, Jira Service Desk, Mattermost, Opsgenie, OTRS, Redmine, ServiceNow, Slack, Zammad y Zendesk proporcionados por Zabbix, así como a la mayoría de los webhooks que utilizan la opción *Include event menu entry*. Tenga en cuenta, sin embargo, que un único webhook puede usarse en varias operaciones o pasos de escalado de la misma acción, así como en diferentes acciones que no se activarán por el mismo evento de problema debido a diferentes condiciones.
- Al usar un webhook en acciones para eventos internos, asegúrese de marcar la casilla *Custom message* y definir un mensaje personalizado en la configuración de la operación de la acción. De lo contrario, no se enviará ninguna notificación.

Ejemplos de scripts de webhook

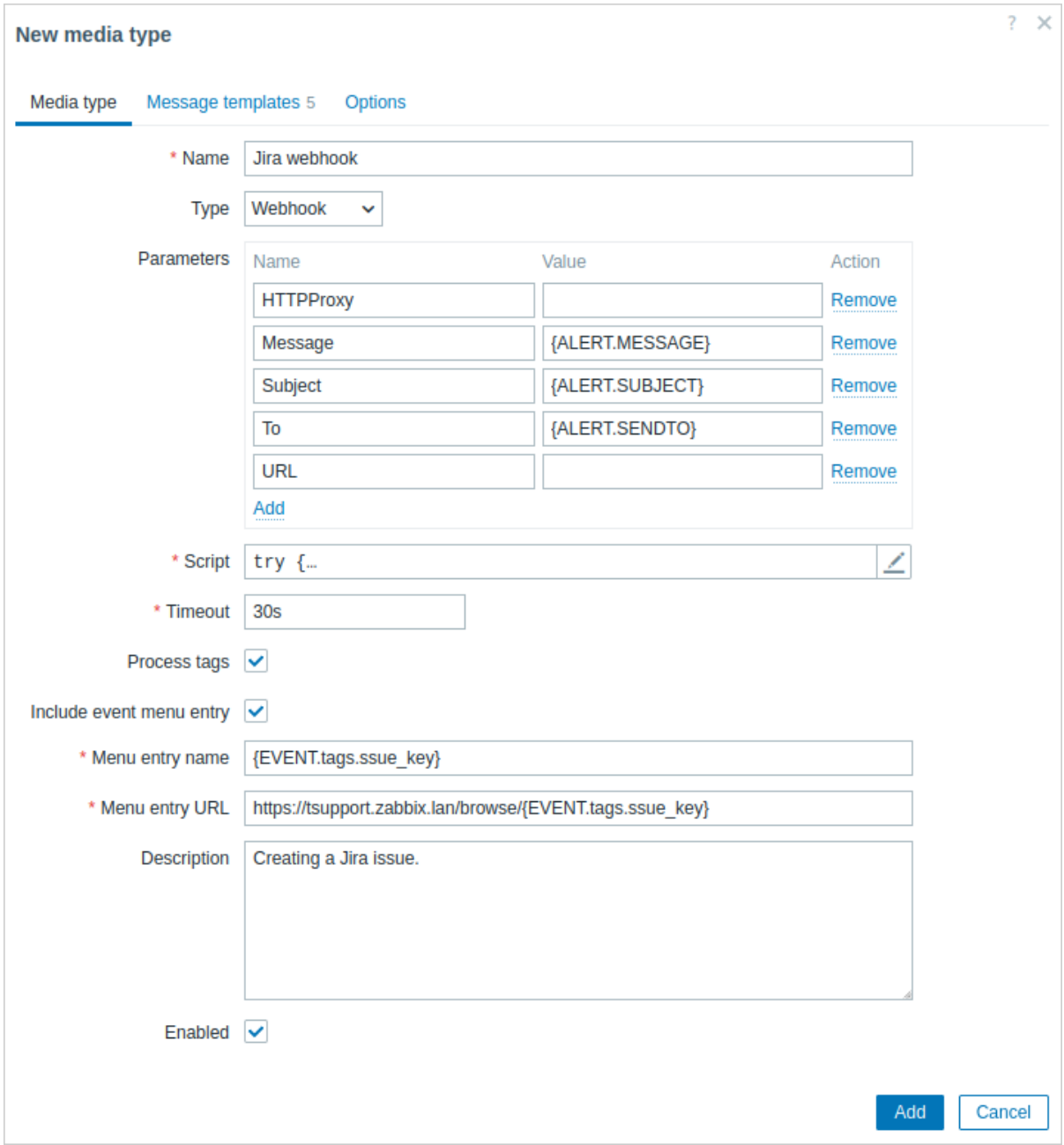
Resumen

Aunque Zabbix ofrece una gran cantidad de integraciones de webhook disponibles de forma predeterminada, es posible que desee crear sus propios webhooks. Esta sección proporciona ejemplos de scripts de webhook personalizados (utilizados en el parámetro *Script*). Consulte la sección webhook para obtener una descripción de otros parámetros de webhook.

Attention:
No use asignaciones no declaradas en JavaScript de preprocesamiento. Use `var` para declarar variables locales.

Los eventos de recuperación (ya sea generados automáticamente o tras un cierre manual) incluyen las etiquetas del evento resuelto (incluidas las etiquetas heredadas de templates, hosts y triggers). Los scripts de webhook se ejecutan después de crear la alerta; por lo tanto, las etiquetas devueltas por un script de webhook se aplican después de la creación inicial de la alerta y pueden no estar presentes en la alerta que activó el webhook. Si una integración requiere que las etiquetas producidas por un webhook estén presentes en la notificación inicial, obtenga las etiquetas del evento desde el server (por ejemplo, usando la API Event) o almacene las etiquetas producidas por el webhook en un almacén externo persistente y correlaciónelas allí.

Webhook de Jira (personalizado)



Este script creará una incidencia en JIRA y devolverá información sobre la incidencia creada.

```
try {
    Zabbix.log(4, '[ Jira webhook ] Iniciado con parámetros: ' + value);

    var result = {
            'tags': {
                'endpoint': 'jira'
            }
        },
        params = JSON.parse(value),
        req = new HttpRequest(),
        fields = {},
        resp;

    if (params.HTTPProxy) {
        req.setProxy(params.HTTPProxy);
    }

    req.addHeader('Content-Type: application/json');
    req.addHeader('Authorization: Basic ' + params.authentication);

    fields.summary = params.summary;
    fields.description = params.description;
    fields.project = {key: params.project_key};
    fields.issuetype = {id: params.issue_id};

    resp = req.post('https://jira.example.com/rest/api/2/issue/',
        JSON.stringify({"fields": fields})
    );

    if (req.getStatus() != 201) {
        throw 'Código de respuesta: ' + req.getStatus();
    }

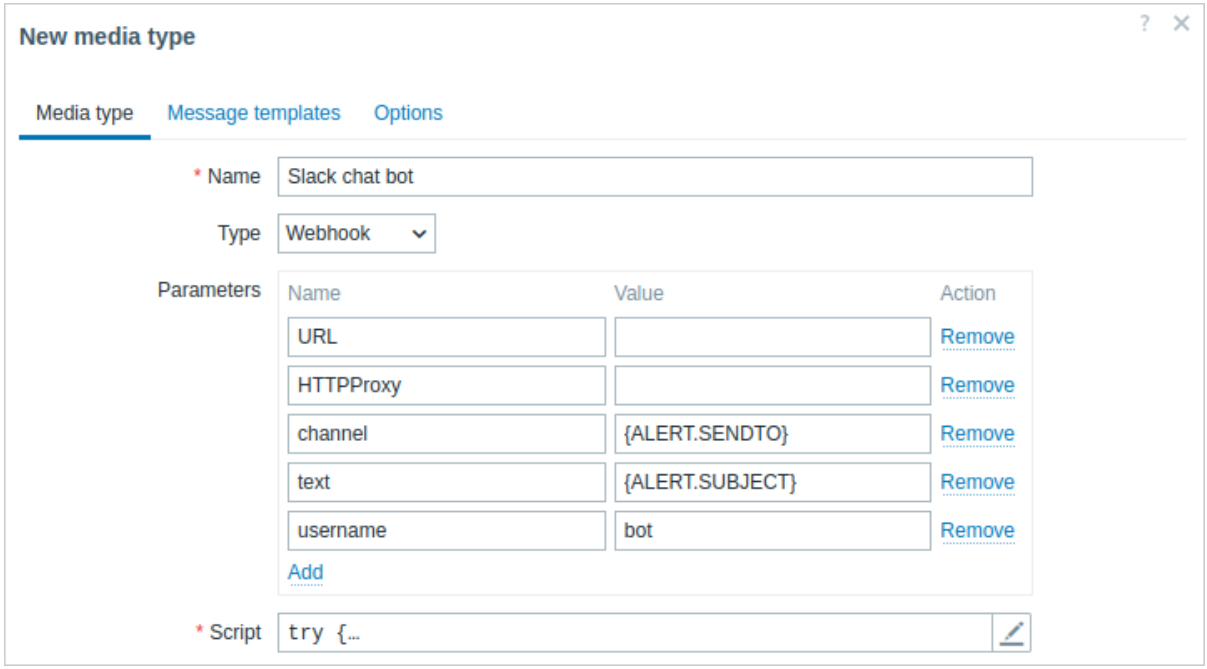
    resp = JSON.parse(resp);
    result.tags.issue_id = resp.id;
    result.tags.issue_key = resp.key;

    return JSON.stringify(result);
}
catch (error) {
    Zabbix.log(4, '[ Jira webhook ] Fallo en la creación de la incidencia json : ' + JSON.stringify({"fiel
    Zabbix.log(3, '[ Jira webhook ] fallo en la creación de la incidencia : ' + error);

    throw 'Falló con el error: ' + error;
}
```

Slack webhook (personalizado)

Este webhook reenviará las notificaciones de Zabbix a un canal de Slack.




```
try {
    var params = JSON.parse(value),
        req = new HttpRequest(),
        response;

    if (params.HTTPProxy) {
        req.setProxy(params.HTTPProxy);
    }

    req.addHeader('Content-Type: application/x-www-form-urlencoded');

    Zabbix.log(4, '[ Slack webhook ] Webhook request with value=' + value);

    response = req.post(params.URL, 'payload=' + encodeURIComponent(value));
    Zabbix.log(4, '[ Slack webhook ] Responded with code: ' + req.getStatus() + '. Response: ' + response)

    try {
        response = JSON.parse(response);
    }
    catch (error) {
        if (req.getStatus() < 200 || req.getStatus() >= 300) {
            throw 'Request failed with status code ' + req.getStatus();
        }
        else {
            throw 'Request success, but response parsing failed.';
        }
    }

    if (req.getStatus() !== 200 || !response.ok || response.ok === 'false') {
        throw response.error;
    }

    return 'OK';
}
catch (error) {
    Zabbix.log(3, '[ Slack webhook ] Sending failed. Error: ' + error);

    throw 'Failed with error: ' + error;
}
```

Acciones

Descripción general

Si desea que se realicen algunas operaciones como resultado de eventos (por ejemplo, que se envíen notificaciones), debe configurar acciones.

Las acciones pueden definirse en respuesta a eventos de todos los tipos admitidos:

- Acciones de trigger: para eventos cuando el estado del trigger cambia entre `OK` y `Problem`
- Acciones de servicio: para eventos cuando el estado del servicio cambia entre `OK` y `Problem`
- Acciones de descubrimiento: para eventos cuando se produce el descubrimiento de red
- Acciones de autorregistro: para eventos cuando nuevos agents activos se autorregistran (o cambian los metadatos del host para los ya registrados)
- Acciones internas: para eventos cuando los items dejan de ser compatibles o los triggers pasan a un estado desconocido

Las principales diferencias de las acciones de servicio son:

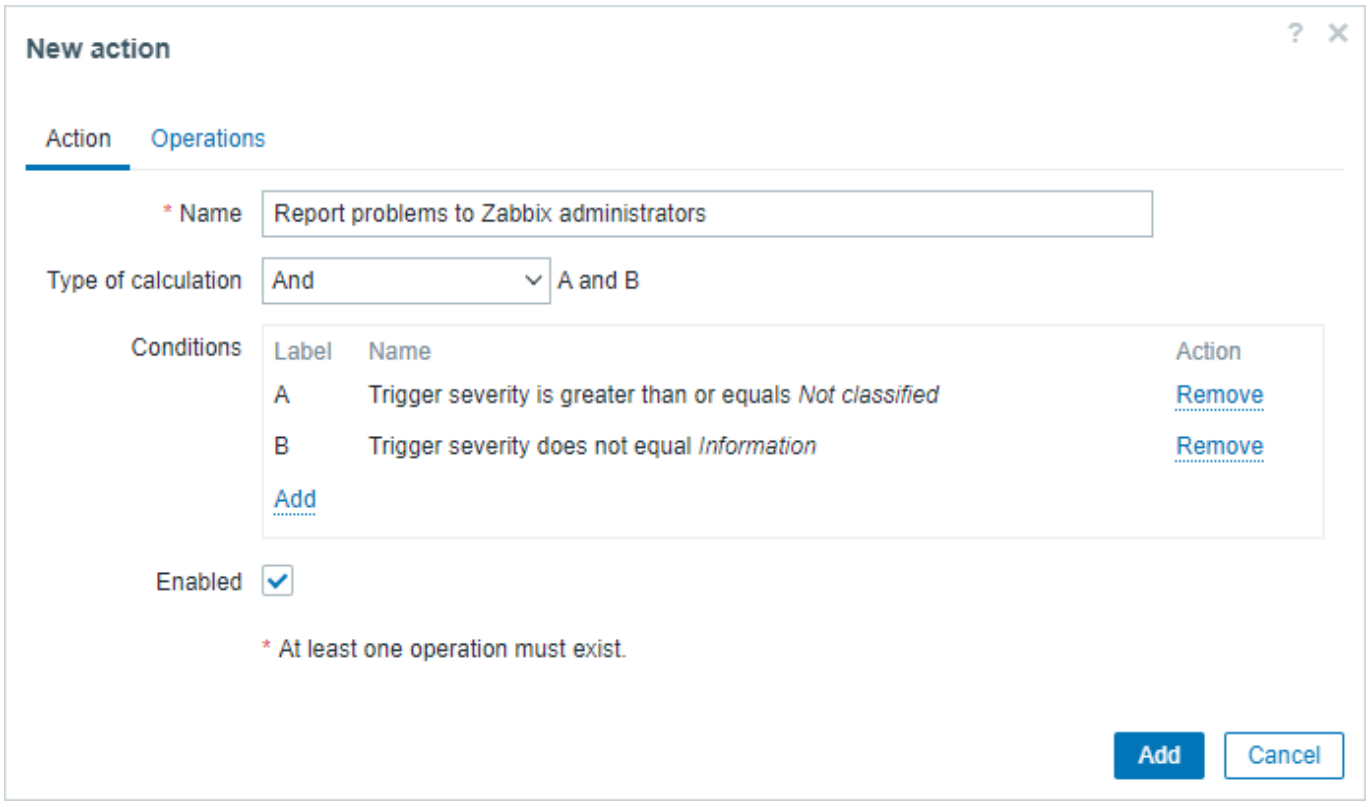
- El acceso del usuario a las acciones de servicio depende de los derechos de acceso a los servicios concedidos por el rol del usuario
- Las acciones de servicio admiten un conjunto diferente de condiciones

Configuración de una action

Para configurar una action, haga lo siguiente:

1. Vaya a *Alerts* > *Actions* y seleccione el tipo de action requerido en el submenú (puede cambiar a otro tipo más adelante usando la lista desplegable del título).
2. Haga clic en *Create action*.
3. Asigne un nombre a la action.
4. Elija las conditions sobre las que se realizarán las operaciones.
5. Elija las operations que se llevarán a cabo.

Atributos generales de la action:



Todos los campos obligatorios están marcados con un asterisco rojo.

Parameter	Description
Name	Nombre único de la action.

Parameter	Description
Type of calculation	Seleccione la opción de evaluación para las conditions de la action (con más de una condition): **And** - todas las conditions deben cumplirse. **Or** - basta con que se cumpla una condition. **And/Or** - combinación de ambas: AND con distintos tipos de condition y OR con el mismo tipo de condition. **Custom expression** - una fórmula de cálculo definida por el usuario para evaluar las conditions de la action.
Conditions	Lista de conditions de la action. Haga clic en *Add* para añadir una nueva condition. Si no se configuran conditions, la action se ejecutará para cada evento que corresponda al tipo de action que se esté configurando.
Enabled	Marque la casilla para habilitar la action. En caso contrario, quedará deshabilitada.

Condiciones

Resumen

Es posible definir que una acción se ejecute solo si el evento coincide con un conjunto definido de condiciones. Las condiciones se establecen al configurar la action.

La coincidencia de condiciones distingue entre mayúsculas y minúsculas.

Acciones de disparador

Se pueden utilizar las siguientes condiciones en acciones basadas en disparadores:

Tipo de condición	Operadores soportados	Descripción
Grupo de equipos	igual no igual	Especifique grupos de equipos o grupos de equipos a excluir. **igual**: el evento pertenece a este grupo de equipos. **no igual**: el evento no pertenece a este grupo de equipos. Al especificar un grupo de equipos padre, se seleccionan implícitamente todos los grupos de equipos anidados. Para especificar solo el grupo padre, todos los grupos anidados deben establecerse adicionalmente con el operador **no igual**.
Plantilla	igual no igual	Especifique plantillas o plantillas a excluir. **igual**: el evento pertenece a un disparador heredado de esta plantilla. **no igual**: el evento no pertenece a un disparador heredado de esta plantilla.
Equipo	igual no igual	Especifique equipos o equipos a excluir. **igual**: el evento pertenece a este equipo. **no igual**: el evento no pertenece a este equipo.
Nombre de etiqueta	igual no igual contiene no contiene	Especifique una etiqueta de evento o una etiqueta de evento a excluir. **igual**: el evento tiene esta etiqueta. **no igual**: el evento no tiene esta etiqueta. **contiene**: el evento tiene una etiqueta que contiene esta cadena. **no contiene**: el evento no tiene una etiqueta que contenga esta cadena.
Valor de etiqueta	igual no igual contiene no contiene	Especifique una combinación de etiqueta y valor de evento o una combinación de etiqueta y valor a excluir. **igual**: el evento tiene esta etiqueta y valor. **no igual**: el evento no tiene esta etiqueta y valor. **contiene**: el evento tiene una etiqueta y valor que contienen estas cadenas. **no contiene**: el evento no tiene una etiqueta y valor que contengan estas cadenas.
Disparador	igual no igual	Especifique disparadores o disparadores a excluir. **igual**: el evento es generado por este disparador. **no igual**: el evento es generado por cualquier otro disparador, excepto este.

Tipo de condición	Operadores soportados	Descripción
Nombre del evento	contiene no contiene	Especifique una cadena en el nombre del evento generado por el disparador o una cadena a excluir. Por defecto, el nombre del evento coincide con el nombre del disparador a menos que se especifique un nombre de evento personalizado en la configuración del disparador. **contiene**: el nombre del evento contiene esta cadena. **no contiene**: esta cadena se excluye del nombre del evento. Nota: El valor introducido se comparará con el nombre del evento con todas las macros expandidas.
Gravedad del disparador	igual no igual es mayor o igual que es menor o igual que	Especifique la gravedad del disparador. **igual**: igual a la gravedad del disparador. **no igual**: distinto a la gravedad del disparador. **es mayor o igual que**: mayor o igual a la gravedad del disparador. **es menor o igual que**: menor o igual a la gravedad del disparador.
Periodo de tiempo	en no en	Especifique un periodo de tiempo o un periodo de tiempo a excluir. **en**: la hora del evento está dentro del periodo de tiempo. **no en**: la hora del evento no está dentro del periodo de tiempo. Consulte la página de especificación de periodo de tiempo para la descripción del formato. Se admiten macros de usuario.
El problema está suprimido	no sí	Especifique si el problema está suprimido (no mostrado) debido al mantenimiento del equipo. **no**: el problema no está suprimido. **sí**: el problema está suprimido.

Acciones de servicio

Las siguientes condiciones se pueden usar en las acciones de servicio:

Condition type	Supported operators	Description
Service	equals does not equal	Especifique un service o un service que excluir. **equals** - el evento pertenece a este service. **does not equal** - el evento no pertenece a este service. Especificar un service padre selecciona implícitamente todos los servicios secundarios. Para especificar solo el service padre, todos los servicios anidados deben configurarse adicionalmente con el operador **does not equal**.
Service name	contains does not contain	Especifique una cadena en el nombre del service o una cadena que excluir. **contains** - el evento es generado por un service que contiene esta cadena en el nombre. **does not contain** - esta cadena no se puede encontrar en el nombre del service.
Service tag name	equals does not equal contains does not contain	Especifique una etiqueta de evento o una etiqueta de evento que excluir. Las etiquetas de evento del service se pueden definir en la sección de configuración del service *Tags*. **equals** - el evento tiene esta etiqueta. **does not equal** - el evento no tiene esta etiqueta. **contains** - el evento tiene una etiqueta que contiene esta cadena. **does not contain** - el evento no tiene una etiqueta que contiene esta cadena.

Condition type	Supported operators	Description
Service tag value	equals does not equal contains does not contain	Especifique una combinación de etiqueta y valor de evento o una combinación de etiqueta y valor que excluir. Las etiquetas de evento del service se pueden definir en la sección de configuración del service *Tags*. **equals** - el evento tiene esta etiqueta y valor. **does not equal** - el evento no tiene esta etiqueta y valor. **contains** - el evento tiene una etiqueta y valor que contienen estas cadenas. **does not contain** - el evento no tiene una etiqueta y valor que contienen estas cadenas.

Attention:
Asegúrese de definir message templates para las acciones de service en el menú *Alerts* > *Media types*. De lo contrario, no se enviarán las notificaciones.

Acciones de descubrimiento

Las siguientes condiciones se pueden utilizar en eventos basados en descubrimiento:

Tipo de condición	Operadores admitidos	Descripción
IP del host	es igual a no es igual a	Especifique un rango de direcciones IP o un rango que excluir para un host descubierto. **es igual a** - la IP del host está dentro del rango. **no es igual a** - la IP del host no está dentro del rango. Puede tener los siguientes formatos: IP única: 192.168.1.33 Rango de direcciones IP: 192.168.1-10.1-254 Máscara IP: 192.168.4.0/24 Lista: 192.168.1.1-254, 192.168.2.1-100, 192.168.2.200, 192.168.4.0/24 Se admiten espacios en el formato de lista.
Tipo de servicio	es igual a no es igual a	Especifique el tipo de servicio de un servicio descubierto o un tipo de servicio que excluir. **es igual a** - coincide con el servicio descubierto. **no es igual a** - no coincide con el servicio descubierto. Tipos de servicio disponibles: SSH, LDAP, SMTP, FTP, HTTP, HTTPS, POP, NNTP, IMAP, TCP, Zabbix agent, agente SNMPv1, agente SNMPv2, agente SNMPv3, ping ICMP, telnet.
Puerto del servicio	es igual a no es igual a	Especifique un rango de puertos TCP de un servicio descubierto o un rango que excluir. **es igual a** - el puerto del servicio está dentro del rango. **no es igual a** - el puerto del servicio no está dentro del rango.
Regla de descubrimiento	es igual a no es igual a	Especifique una regla de descubrimiento o una regla de descubrimiento que excluir. **es igual a** - utiliza esta regla de descubrimiento. **no es igual a** - utiliza cualquier otra regla de descubrimiento, excepto esta.
Comprobación de descubrimiento	es igual a no es igual a	Especifique una comprobación de descubrimiento o una comprobación de descubrimiento que excluir. **es igual a** - utiliza esta comprobación de descubrimiento. **no es igual a** - utiliza cualquier otra comprobación de descubrimiento, excepto esta.
Objeto de descubrimiento	es igual a	Especifique el objeto descubierto. **es igual a** - es igual al objeto descubierto (un dispositivo o un servicio).
Estado del descubrimiento	es igual a	**Up** - coincide con los eventos 'Host Up' y 'Service Up'. **Down** - coincide con los eventos 'Host Down' y 'Service Down'. **Discovered** - coincide con los eventos 'Host Discovered' y 'Service Discovered'. **Lost** - coincide con los eventos 'Host Lost' y 'Service Lost'.

Tipo de condición	Operadores admitidos	Descripción
Tiempo de actividad/inactividad	es mayor o igual que es menor o igual que	Tiempo de actividad para los eventos 'Host Up' y 'Service Up'. Tiempo de inactividad para los eventos 'Host Down' y 'Service Down'. **es mayor o igual que** - es mayor o igual. El parámetro se indica en segundos. **es menor o igual que** - es menor o igual. El parámetro se indica en segundos.
Valor recibido	es igual a no es igual a es mayor o igual que es menor o igual que contiene no contiene	Especifique el valor recibido de una comprobación de agent (Zabbix, SNMP) en una regla de descubrimiento. Comparación de cadenas. Si se configuran varias comprobaciones de Zabbix agent o SNMP para una regla, se comprueban los valores recibidos para cada una de ellas (cada comprobación genera un evento nuevo que se compara con todas las condiciones). **es igual a** - es igual al valor. **no es igual a** - no es igual al valor. **es mayor o igual que** - es mayor o igual al valor. **es menor o igual que** - es menor o igual al valor. **contiene** - contiene la subcadena. El parámetro se indica como una cadena. **no contiene** - no contiene la subcadena. El parámetro se indica como una cadena.
Proxy	es igual a no es igual a	Especifique un proxy o un proxy que excluir. **es igual a** - utiliza este proxy. **no es igual a** - utiliza cualquier otro proxy, excepto este.
Grupo de proxy	es igual a no es igual a	Especifique un grupo de proxy o un grupo de proxy que excluir. **es igual a** - utiliza este grupo de proxy. **no es igual a** - utiliza cualquier otro grupo de proxy, excepto este.

Note:
Las comprobaciones de servicio de una regla de descubrimiento que generan eventos de descubrimiento no se realizan simultáneamente. Por lo tanto, si se configuran **varios** valores para las condiciones `Service type`, `Service port` o `Received value` en la acción, se compararán con un evento de descubrimiento a la vez, pero **no** con varios eventos simultáneamente. Como resultado, es posible que las acciones con varios valores para los mismos tipos de comprobación no se ejecuten correctamente.

Acciones de autoregistro

Las siguientes condiciones se pueden utilizar en acciones basadas en el autoregistro de un agent activo:

Tipo de condición	Operadores compatibles	Descripción
Metadatos del host	contiene no contiene coincide no coincide	Especifique los metadatos del host o los metadatos del host que desea excluir. **contiene**: los metadatos del host contienen la cadena. **no contiene**: los metadatos del host no contienen la cadena. Los metadatos del host se pueden especificar en un archivo de configuración del agent. **coincide**: los metadatos del host coinciden con la expresión regular. **no coincide**: los metadatos del host no coinciden con la expresión regular.
Nombre del host	contiene no contiene coincide no coincide	Especifique un nombre de host o un nombre de host que desea excluir. **contiene**: el nombre del host contiene la cadena. **no contiene**: el nombre del host no contiene la cadena. **coincide**: el nombre del host coincide con la expresión regular. **no coincide**: el nombre del host no coincide con la expresión regular.
Proxy	es igual a no es igual a	Especifique un proxy o un proxy que desea excluir. **es igual a**: utiliza este proxy. **no es igual a**: utiliza cualquier otro proxy (o server), excepto este proxy.

Tipo de condición	Operadores compatibles	Descripción
Grupo de proxies	es igual a no es igual a	Especifique un grupo de proxies o un grupo de proxies que desea excluir. **es igual a**: utiliza este grupo de proxies. **no es igual a**: utiliza cualquier otro grupo de proxies, excepto este.

Acciones de eventos internos

Se pueden establecer las siguientes condiciones para acciones basadas en eventos internos:

Tipo de condición	Operadores soportados	Descripción
Tipo de evento	igual	**Métrica en estado "no soportado"**: coincide con eventos donde una métrica pasa de un estado 'normal' a 'no soportado'. **Regla de bajo nivel de descubrimiento en estado "no soportado"**: coincide con eventos donde una regla de bajo nivel de descubrimiento pasa de un estado 'normal' a 'no soportado'. **Disparador en estado "desconocido"**: coincide con eventos donde un disparador pasa de un estado 'normal' a 'desconocido'.
Grupo de equipos	igual no igual	Especifique grupos de equipos o grupos de equipos a excluir. **igual**: el evento pertenece a este grupo de equipos. **no igual**: el evento no pertenece a este grupo de equipos.
Nombre de etiqueta	igual no igual contiene no contiene	Especifique la etiqueta del evento o la etiqueta del evento a excluir. **igual**: el evento tiene esta etiqueta. **no igual**: el evento no tiene esta etiqueta. **contiene**: el evento tiene una etiqueta que contiene esta cadena. **no contiene**: el evento no tiene una etiqueta que contenga esta cadena.
Valor de etiqueta	igual no igual contiene no contiene	Especifique la combinación de etiqueta y valor del evento o la combinación de etiqueta y valor a excluir. **igual**: el evento tiene esta etiqueta y valor. **no igual**: el evento no tiene esta etiqueta y valor. **contiene**: el evento tiene una etiqueta y valor que contienen estas cadenas. **no contiene**: el evento no tiene una etiqueta y valor que contengan estas cadenas.
Plantilla	igual no igual	Especifique plantillas o plantillas a excluir. **igual**: el evento pertenece a una métrica/disparador/regla de bajo nivel de descubrimiento heredada de esta plantilla. **no igual**: el evento no pertenece a una métrica/disparador/regla de bajo nivel de descubrimiento heredada de esta plantilla.
Equipo	igual no igual	Especifique equipos o equipos a excluir. **igual**: el evento pertenece a este equipo. **no igual**: el evento no pertenece a este equipo.

Tipo de cálculo

Las siguientes opciones de cálculo de condiciones están disponibles:

- **Y** - todas las condiciones deben cumplirse

Tenga en cuenta que el uso del cálculo "Y" no está permitido entre varios triggers cuando se seleccionan como una condición `Trigger=`. Las acciones solo pueden ejecutarse en función del evento de un trigger.

- **O** - basta con que se cumpla una condición
- **Y/O** - combinación de ambas: Y con diferentes tipos de condición y O con el mismo tipo de condición, por ejemplo:

Grupo de hosts igual a Oracle servers
 Grupo de hosts igual a MySQL servers
 Nombre del evento contiene 'Database is down'
 Nombre del evento contiene 'Database is unavailable'

se evalúa como

(Grupo de hosts igual a Oracle servers **o** Grupo de hosts igual a MySQL servers**) y (**Nombre del evento contiene 'Database is down' **o** Nombre del evento contiene 'Database is unavailable'**)**

- **Expresión personalizada** - una fórmula de cálculo definida por el usuario para evaluar las condiciones de la acción. Debe incluir todas las condiciones (representadas como letras mayúsculas A, B, C, ...) y puede incluir espacios, tabulaciones, paréntesis ( ), **and** (sensible a mayúsculas), **or** (sensible a mayúsculas), **not** (sensible a mayúsculas).

Mientras que el ejemplo anterior con Y/O se representaría como (A o B) y (C o D), en una expresión personalizada también puede tener múltiples otras formas de cálculo:

(A y B) y (C o D)
 (A y B) o (C y D)
 ((A o B) y C) o D
 (no (A o B) y C) o no D
 etc.

Eliminación de objetos utilizados en acciones

Al intentar eliminar objetos que se utilizan en una condición/operación de acción, el usuario recibirá un mensaje de error correspondiente.

Las acciones no se deshabilitan si las condiciones u operaciones de la acción:

- tienen objetos eliminados;
- tienen objetos que ya no son descubiertos por el descubrimiento de bajo nivel y han sido eliminados (después del período *Eliminar recursos perdidos*).

Las condiciones/operaciones de acción no se eliminan si tienen objetos faltantes. En su lugar, los objetos eliminados se muestran como *Grupo de equipos eliminado*, *Equipo eliminado*, *Disparador eliminado*, etc. Se muestra un icono de advertencia con el mensaje "Esta acción tiene condiciones u operaciones que hacen referencia a objeto(s) eliminado(s)." en la columna "Info" de la lista de acciones.

Operaciones

Resumen

Puede definir las siguientes operaciones para todos los eventos:

- Enviar un mensaje
- Ejecutar un comando remoto

> **Attention:**
> Zabbix server no crea alertas si el acceso al host se "deniega" explícitamente para el usuario definido como destinatario de la operación de la acción o si el usuario no tiene ningún permiso definido sobre el host en absoluto.

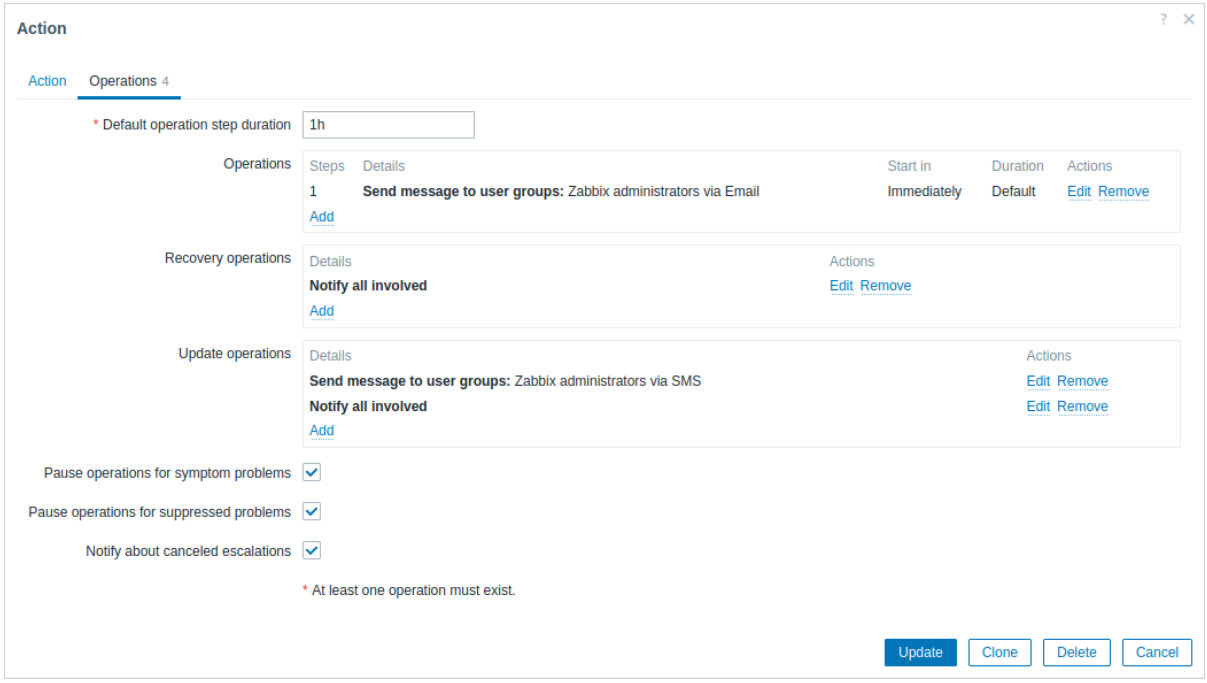
Para los eventos de descubrimiento y autorregistro, hay disponibles operaciones adicionales:

- Añadir host
- Eliminar host
- Habilitar host
- Deshabilitar host
- Añadir a grupo de hosts
- Eliminar de grupo de hosts
- Añadir etiquetas de host
- Eliminar etiquetas de host
- Vincular template
- Desvincular template
- Establecer el modo de inventario del host

Configuración de una operación

Para configurar una operación:

1. Vaya a la pestaña *Operations* en la configuración de la action.
2. Configure los detalles de la operación.



Atributos generales de la operación:

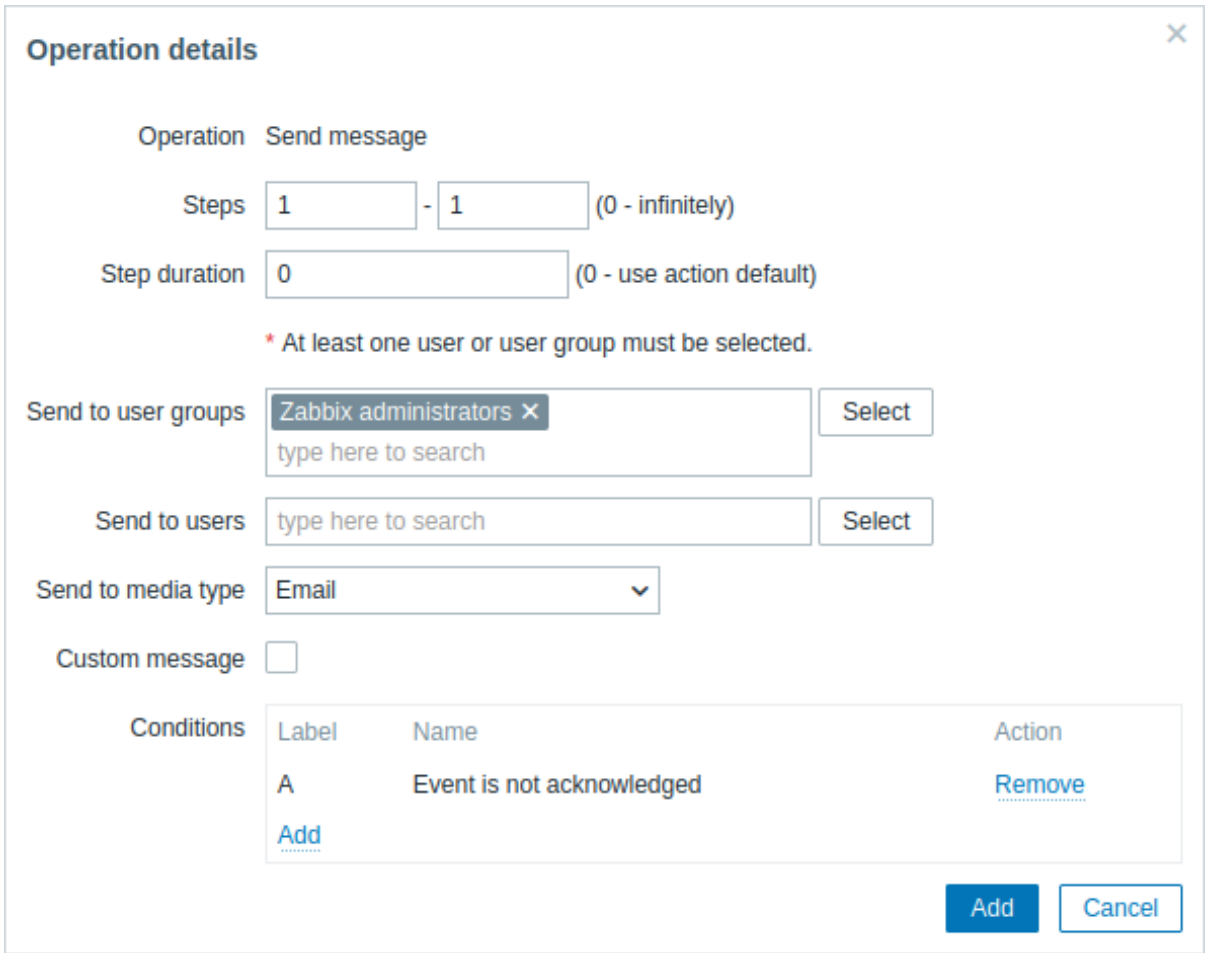
Parameter	Description
Default operation step duration	Duración predeterminada de un paso de operación (de 60 segundos a 1 semana). Por ejemplo, una duración de paso de una hora significa que, si se ejecuta una operación, transcurrirá una hora antes del siguiente paso. Se admiten sufijos de tiempo, por ejemplo, 60s, 1m, 2h, 1d. Se admiten macros de usuario.
Operations	Se muestran las operaciones de la acción (si las hay), con estos detalles: **Steps**: pasos de escalación a los que está asignada la operación. **Details**: tipo de operación y su destinatario/objetivo. La lista de operaciones también muestra el tipo de medio (correo electrónico, SMS o script) utilizado, así como el nombre y los apellidos (entre paréntesis después del nombre de usuario) del destinatario de la notificación. **Start in**: cuánto tiempo después de un evento se ejecuta la operación. **Duration (sec)**: se muestra la duración del paso. Se muestra *Default* si el paso utiliza la duración predeterminada, y se muestra un tiempo si se utiliza una duración personalizada. **Actions**: se muestran enlaces para editar y eliminar una operación.
Recovery operations	Se muestran las operaciones de la acción (si las hay), con estos detalles: **Details**: tipo de operación y su destinatario/objetivo. La lista de operaciones también muestra el tipo de medio (correo electrónico, SMS o script) utilizado, así como el nombre y los apellidos (entre paréntesis después del nombre de usuario) del destinatario de la notificación. **Actions**: se muestran enlaces para editar y eliminar una operación.
Update operations	Se muestran las operaciones de la acción (si las hay), con estos detalles: **Details**: tipo de operación y su destinatario/objetivo. La lista de operaciones también muestra el tipo de medio (correo electrónico, SMS o script) utilizado, así como el nombre y los apellidos (entre paréntesis después del nombre de usuario) del destinatario de la notificación. **Actions**: se muestran enlaces para editar y eliminar una operación.
Pause operations for symptom problems	Marque esta casilla para pausar las operaciones (después de la primera operación) para los problemas de tipo síntoma. Tenga en cuenta que esta configuración solo afecta a las escalaciones de problemas; las operaciones de recuperación y actualización no se verán afectadas. Esta opción solo está disponible para las *Trigger actions*.

Parameter	Description
Pause operations for suppressed problems	Marque esta casilla para retrasar el inicio de las operaciones durante un período de mantenimiento. Cuando se inicien las operaciones, después del mantenimiento se ejecutarán todas las operaciones, incluidas las correspondientes a los eventos ocurridos durante el mantenimiento. Tenga en cuenta que esta configuración solo afecta a las escalaciones de problemas; las operaciones de recuperación y actualización no se verán afectadas. Si desmarca esta casilla, las operaciones se ejecutarán sin demora incluso durante un período de mantenimiento. Esta opción no está disponible para las *Service actions*.
Notify about canceled escalations	Desmarque esta casilla para deshabilitar las notificaciones sobre escalaciones canceladas (cuando se deshabilita un host, item, trigger o action).

Todos los campos de entrada obligatorios están marcados con un asterisco rojo.

Para configurar los detalles de una operación nueva, haga clic en Add en el bloque *Operations*. Para editar una operación existente, haga clic en Edit junto a la operación. Se abrirá una ventana emergente en la que podrá editar los detalles del paso de la operación.

Detalles de la operación



Parámetro	Descripción
Operación	Seleccione la operación: **Enviar mensaje**: envía un mensaje al usuario. **<nombre del comando remoto>**: ejecuta un comando remoto. Los comandos están disponibles para su ejecución si se han definido previamente en scripts globales con *Operación de acción* seleccionada como su ámbito. Hay más operaciones disponibles para eventos basados en descubrimiento y autoregistro (consulte arriba).

Parámetro	Descripción
Pasos	Seleccione el o los pasos a los que asignar la operación en una programación de escalación: **Desde**: ejecuta comenzando con este paso. **Hasta**: ejecuta hasta este paso (0=infinito; la ejecución no estará limitada).
Duración del paso	Duración personalizada para estos pasos (0=usar la duración de paso predeterminada). Se admiten sufijos de tiempo, por ejemplo, 60s, 1m, 2h, 1d. Se admiten macros de usuario. Se pueden asignar varias operaciones al mismo paso. Si estas operaciones tienen definida una duración de paso diferente, se tiene en cuenta la más corta y se aplica al paso.
Tipo de operación: enviar mensaje	
Enviar a grupos de usuarios	Seleccione los grupos de usuarios a los que enviar el mensaje. El grupo de usuarios debe tener al menos permisos de ”lectura” sobre el host para poder recibir la notificación.
Enviar a usuarios	Seleccione los usuarios a los que enviar el mensaje. El usuario debe tener al menos permisos de ”lectura” sobre el host para poder recibir la notificación.
Enviar al tipo de medio	Envía el mensaje a todos los tipos de medio disponibles (configurados y habilitados) o solo a uno específico.
Mensaje personalizado	Si se selecciona, se puede configurar el mensaje personalizado. Para las notificaciones sobre eventos internos mediante webhooks, el mensaje personalizado es obligatorio.
Asunto	Asunto del mensaje personalizado. El asunto puede contener macros. Está limitado a 255 caracteres.
Mensaje	Mensaje personalizado. El mensaje puede contener macros. Está limitado a una cantidad determinada de caracteres según el tipo de base de datos (consulte Enviar mensaje para obtener más información).
Tipo de operación: comando remoto	

Parámetro	Descripción
Lista de destinos	Seleccione los destinos en los que se ejecutará el comando: **Host actual**: el comando se ejecuta en el host del trigger que causó el evento de problema. Esta opción no funcionará si hay varios hosts en el trigger. **Host**: seleccione el o los hosts en los que se ejecutará el comando. **Grupo de hosts**: seleccione el o los grupos de hosts en los que se ejecutará el comando. Especificar un grupo de hosts principal selecciona implícitamente todos los grupos de hosts anidados. Por lo tanto, el comando remoto también se ejecutará en los hosts de los grupos anidados. Un comando en un host se ejecuta una sola vez, incluso si el host coincide más de una vez (por ejemplo, desde varios grupos de hosts; individualmente y desde un grupo de hosts). La lista de destinos no tiene sentido si se ejecuta un script personalizado en Zabbix server. Seleccionar más destinos en este caso solo hace que el script se ejecute más veces en el server. Tenga en cuenta que, para los scripts globales, la selección de destinos también depende de la configuración *Grupo de hosts* en la configuración del script global. La opción *Lista de destinos* no está disponible para las *acciones de servicio*, ya que en este caso los comandos remotos siempre se ejecutan en Zabbix server.
Condiciones	Condición para realizar la operación: **El evento no está reconocido**: solo cuando el evento no está reconocido. **El evento está reconocido**: solo cuando el evento está reconocido. La opción *Condiciones* solo está disponible para las *acciones de trigger*.

Cuando haya terminado, haga clic en *Añadir* para añadir la operación a la lista de *Operaciones*.

Envío de mensaje

Descripción general

Enviar un mensaje es una de las mejores formas de notificar a las personas sobre un problema. Por eso es una de las acciones principales que ofrece Zabbix.

Configuración

Para poder enviar y recibir notificaciones de Zabbix, debe:

- Definir el medio al que se enviará el mensaje.

Si la operación tiene lugar fuera del período de tiempo Activo definido para el medio seleccionado en la configuración del usuario, el mensaje no se enviará.

La severidad de trigger predeterminada («No clasificado») **debe estar** seleccionada en la configuración de medios del usuario si desea recibir notificaciones de eventos que no sean triggers, como eventos de descubrimiento, autorregistro activo de agent o eventos internos.

- Configure una operación de acción que envíe un mensaje a uno de los medios definidos.

Attention:
Zabbix envía notificaciones únicamente a los usuarios que tienen al menos permisos de «lectura» sobre el host que generó el evento. Al menos un host de una expresión de trigger debe ser accesible.

Puede configurar escenarios personalizados para enviar mensajes mediante escalaciones.

Para recibir y leer correctamente los correos electrónicos de Zabbix, los servidores/clientes de correo electrónico deben admitir el formato estándar de «correo electrónico SMTP/MIME», ya que Zabbix envía datos UTF-8 (si el asunto contiene únicamente caracteres ASCII, no se codifica en UTF-8). El asunto y el cuerpo del mensaje se codifican en base64 para cumplir el estándar de formato de «correo electrónico SMTP/MIME».

El límite de mensajes después de expandir todas las macros es el mismo que el límite de mensajes para comandos remotos.

Seguimiento de mensajes

Puede ver el estado de los mensajes enviados en *Monitoring* > *Problems*.

En la columna *Actions* puede ver información resumida sobre las acciones realizadas. En ella, los números verdes representan mensajes enviados, y los rojos, mensajes fallidos. *In progress* indica que se ha iniciado una acción. *Failed* informa de que ninguna acción se ha ejecutado correctamente.

Si hace clic en la hora del evento para ver los detalles del evento, podrá ver los detalles de los mensajes enviados (o no enviados) debido al evento en el bloque *Actions*.

En *Reports* > *Action log* verá los detalles de todas las acciones realizadas para aquellos eventos que tienen una acción configurada.

Comandos remotos

Descripción general

Con los comandos remotos puede definir que un comando predefinido determinado se ejecute automáticamente en el host monitorizado cuando se cumpla alguna condición.

Por lo tanto, los comandos remotos son un mecanismo eficaz para realizar una monitorización proactiva e inteligente.

Entre los usos más evidentes de esta funcionalidad se incluyen:

- Reiniciar automáticamente alguna aplicación (servidor web, middleware, CRM) si no responde
- Utilizar el comando «reboot» de IPMI para reiniciar algún server remoto si no responde a las solicitudes
- Liberar automáticamente espacio en disco (eliminando archivos antiguos, limpiando /tmp) si se está agotando el espacio en disco
- Migrar una VM de un equipo físico a otro en función de la carga de la CPU
- Añadir nuevos nodos a un entorno de nube cuando los recursos de CPU (disco, memoria o cualquier otro) sean insuficientes

Configurar una acción para comandos remotos es similar a configurar una acción para enviar un mensaje; la única diferencia es que Zabbix ejecuta un comando en lugar de enviar un mensaje.

Los comandos remotos pueden ser ejecutados por Zabbix server, Zabbix proxy o Zabbix agent. Tenga en cuenta lo siguiente:

- Los comandos remotos en Zabbix agent pueden ser ejecutados directamente por Zabbix server o a través de Zabbix proxy.
- Los comandos remotos ejecutados por Zabbix server se ejecutan tal como se describe en Ejecución de comandos, incluida la comprobación del código de salida.
- Los comandos remotos se ejecutan incluso si el host de destino se encuentra en modo de mantenimiento.

Los comandos remotos están deshabilitados de forma predeterminada tanto en Zabbix agent como en Zabbix proxy. Pueden habilitarse de la siguiente manera:

- Para los comandos remotos ejecutados por Zabbix agent, añada un parámetro `AllowKey=system.run[<command>,*]` o `AllowKeyRegexp=^system\.run\[<command>,.*\]$` para cada comando permitido en la configuración del agent, donde `*` coincide con cualquiera de los modos de ejecución (`wait` o `nowait`).
- Para los comandos remotos ejecutados por Zabbix proxy, establezca el parámetro `EnableRemoteCommands` en `1` en la configuración del proxy. Tenga en cuenta que no es necesario habilitar los comandos remotos en Zabbix proxy cuando los ejecuta Zabbix agent en un host monitorizado por el proxy.

Attention:
No utilice `AllowKey=system.run[*]` ni `AllowKeyRegexp=^system\.run\[.*\]$` a menos que sea necesario, ya que permite ejecutar cualquier comando en el host. Por motivos de seguridad, se recomienda permitir únicamente comandos exactos, mientras que `system.run[*]` está denegado de forma predeterminada (`DenyKey=system.run[*]`).

Consulte también Restricción de las comprobaciones del agent.

Límite de comandos remotos

El límite de comandos remotos después de resolver todas las macros depende del tipo de base de datos y del conjunto de caracteres (los caracteres no ASCII requieren más de un byte para almacenarse):

Base de datos	Límite en caracteres	Límite en bytes
MySQL	65535	65535
PostgreSQL	65535	sin límite
SQLite (solo Zabbix proxy)	65535	sin límite

La salida de la ejecución de comandos remotos (valor de retorno) está limitada a 16MB (incluyendo los espacios en blanco finales que se truncan). El límite de comandos remotos IPMI depende de la biblioteca IPMI instalada. Tenga en cuenta que los límites de la base de datos se aplican a todos los comandos remotos.

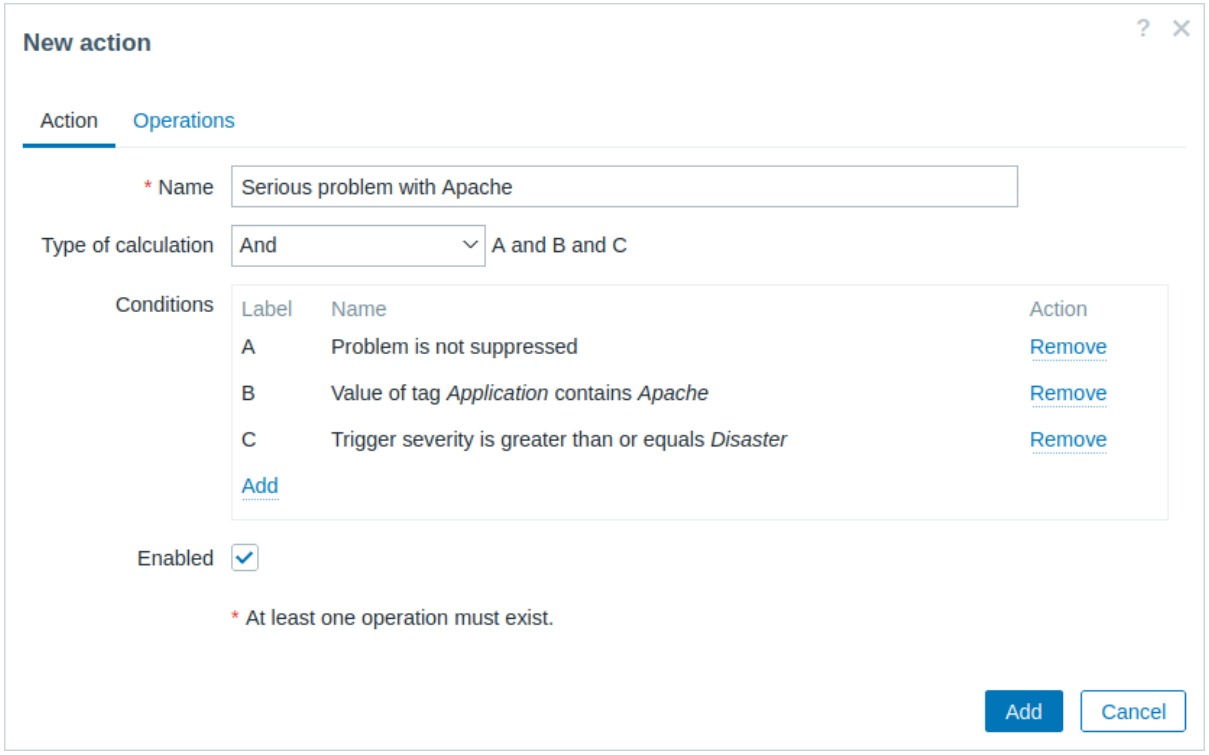
Configuración

Los comandos remotos ejecutados por el Zabbix agent deben permitirse primero en la configuración del agent mediante el parámetro `AllowKey` o `AllowKeyRegexp`.

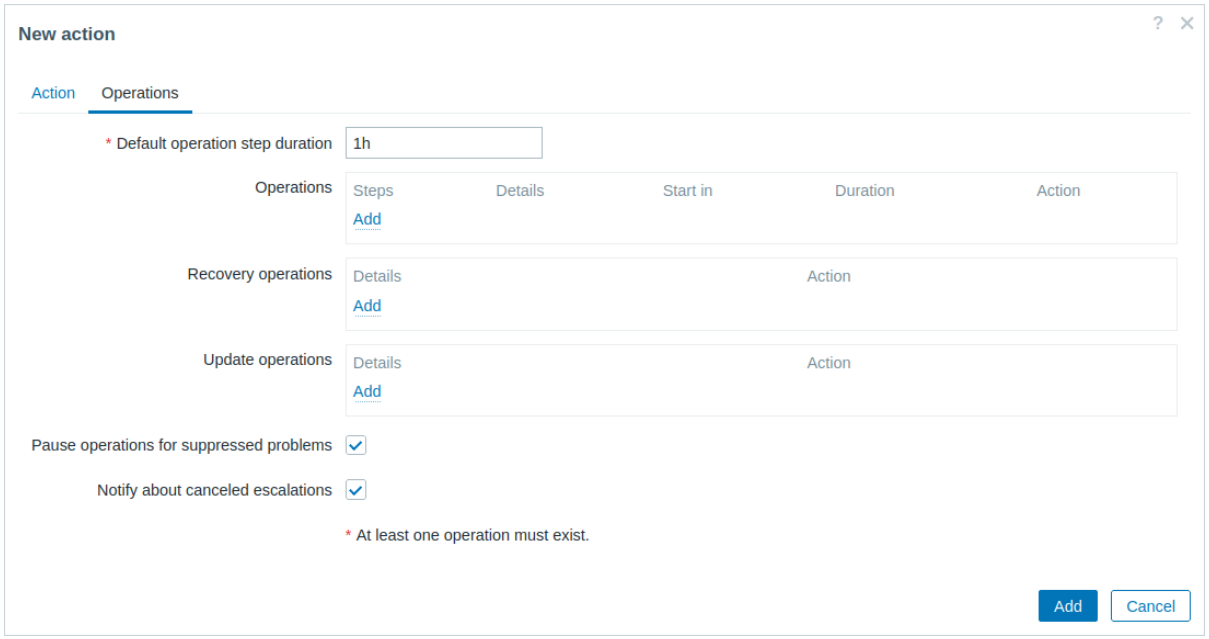
Asegúrese de añadir el parámetro `AllowKey=system.run[<command>,*]` o `AllowKeyRegexp=^system\.run\[<command>,.*\]$` a la configuración del agent para cada comando permitido, a fin de permitir su ejecución en modo `nowait`. Reinicie el daemon del agent después de cambiar el parámetro.

A continuación, al configurar una nueva acción en *Alertas* > *Acciones* > *Acciones de trigger*:

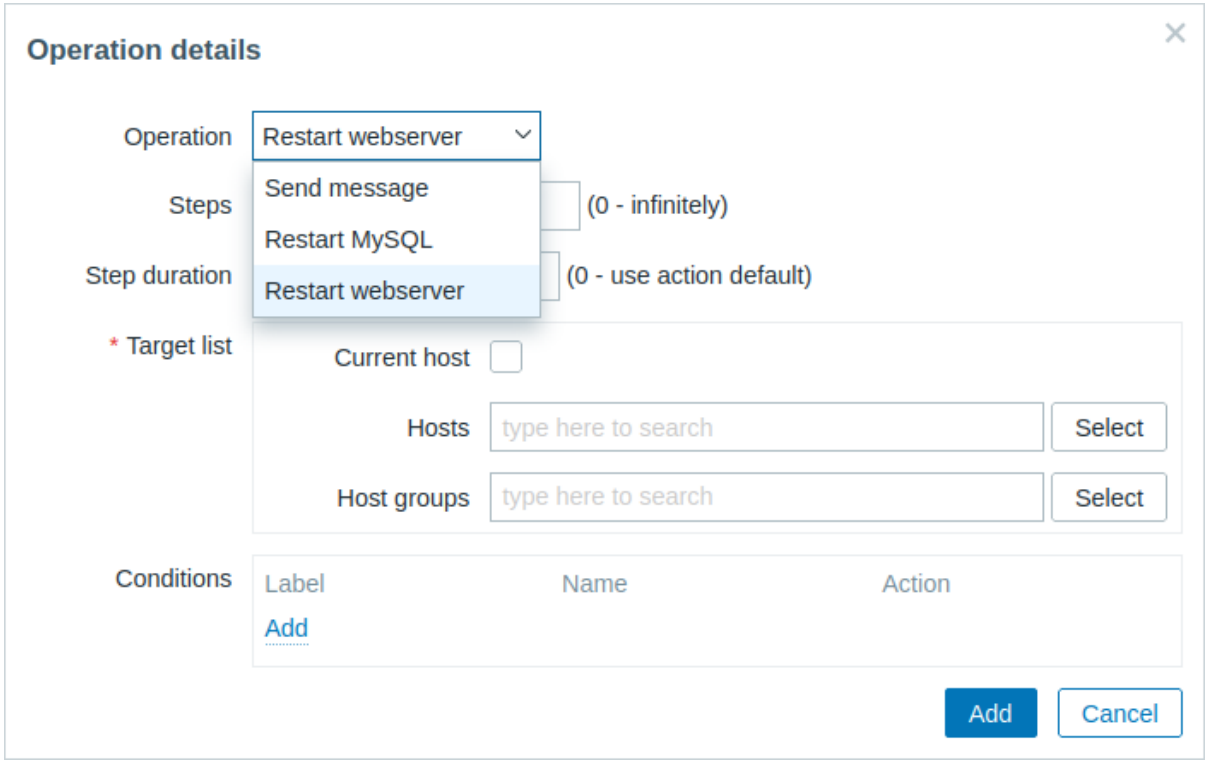
1. Defina las condiciones adecuadas; por ejemplo, establezca que la acción se active ante cualquier problema de desastre relacionado con una de las aplicaciones de Apache.



2. En la pestaña *Operaciones*, haga clic en *Añadir* en la sección *Operaciones*, *Operaciones de recuperación* u *Operaciones de actualización*.



3. Seleccione uno de los scripts predefinidos en la lista desplegable *Operación* y establezca la *Lista de destinos* para el script.



Scripts predefinidos

Los scripts que están disponibles para las operaciones de acción (webhook, script, SSH, Telnet, IPMI) se definen en global scripts con *Action operation* seleccionado como su ámbito.

Por ejemplo:

```
sudo /etc/init.d/apache restart
```

En este caso, Zabbix intentará reiniciar un proceso de Apache. Con este comando, asegúrese de que el comando se ejecute en el agent de Zabbix (haga clic en el botón *Zabbix agent* junto a *Execute on*).

Attention:
Tenga en cuenta el uso de sudo - el usuario de Zabbix no tiene permisos para reiniciar servicios del sistema de forma predeterminada. Consulte más abajo para ver sugerencias sobre cómo configurar sudo.

Note:
A partir de Zabbix agent 7.0, los comandos remotos también pueden ejecutarse en un agent que opera en modo activo. Zabbix agent, ya sea activo o pasivo, debe ejecutarse en el host remoto y ejecuta los comandos en segundo plano.

Los comandos remotos en Zabbix agent se ejecutan sin tiempo de espera mediante la clave `system.run[,nowait]` y no se comprueban los resultados de la ejecución. En Zabbix server y Zabbix proxy, los comandos remotos se ejecutan con el tiempo de espera establecido en el parámetro `TrapperTimeout` del archivo `zabbix_server.conf` o `zabbix_proxy.conf` y se comprueban los resultados de la ejecución. Para obtener información adicional, consulte *Script timeout*.

Permisos de acceso

Asegúrese de que el usuario 'zabbix' tenga permisos de ejecución para los comandos configurados. Puede ser útil usar sudo para dar acceso a comandos privilegiados. Para configurar el acceso, ejecute como root:

```
visudo
```

Ejemplos de líneas que podrían usarse en el archivo sudoers:

```
# permite que el usuario 'zabbix' ejecute todos los comandos sin contraseña.
zabbix ALL=NOPASSWD: ALL

# permite que el usuario 'zabbix' reinicie apache sin contraseña.
zabbix ALL=NOPASSWD: /etc/init.d/apache restart
```

Note:
En algunos sistemas, el archivo `sudoers` impedirá que los usuarios no locales ejecuten comandos. Para cambiar esto, comente la opción `requiretty` en `/etc/sudoers`.

Comandos remotos con múltiples interfaces

Si el sistema de destino tiene múltiples interfaces del tipo seleccionado (agente Zabbix o IPMI), los comandos remotos se ejecutarán en la interfaz predeterminada.

Es posible ejecutar comandos remotos mediante SSH y Telnet utilizando una interfaz diferente a la del agente Zabbix. La interfaz disponible para usar se selecciona en el siguiente orden:

- Interfaz predeterminada del agente Zabbix
- Interfaz predeterminada SNMP
- Interfaz predeterminada JMX
- Interfaz predeterminada IPMI

Comandos remotos IPMI

Para los comandos remotos IPMI se debe utilizar la siguiente sintaxis:

```
<comando> [<valor>]
```

donde

- <comando>: uno de los comandos IPMI sin espacios
- <valor>: 'on', 'off' o cualquier entero sin signo. <valor> es un parámetro opcional.

Ejemplos

Ejemplos de scripts globales que pueden utilizarse como comandos remotos en las operaciones de acción.

Ejemplo 1

Reinicio de Windows bajo ciertas condiciones.

Para reiniciar Windows automáticamente cuando Zabbix detecte un problema, defina el siguiente script:

Script parameter	Value
Scope	'Action operation'
Type	'Script'
Command	`c:\windows\system32\shutdown.exe -r -f`

Ejemplo 2

Reiniciar el equipo utilizando el control IPMI.

Parámetro del script	Valor
Ámbito	'Operación de acción'
Tipo	'IPMI'
Comando	reset

Ejemplo 3

Apagar el equipo utilizando el control IPMI.

Parámetro del script	Valor
Ámbito	'Operación de acción'
Tipo	'IPMI'
Comando	apagar

Operaciones adicionales

Descripción general

En esta sección puede encontrar algunos detalles de operaciones adicionales para eventos de descubrimiento/autoregistro.

Añadiendo un equipo

Los equipos se añaden durante el proceso de descubrimiento, tan pronto como se descubre un equipo, en lugar de al final del proceso de descubrimiento.

> **Note:**
> Como el descubrimiento de red puede llevar algún tiempo, debido a muchos equipos/servicios no disponibles, se recomienda tener paciencia y utilizar rangos de IP razonables.

Al añadir un equipo, su nombre se decide mediante la función estándar **gethostbyname**. Si el equipo puede resolverse, se utiliza el nombre resuelto. Si no, se utiliza la dirección IP. Además, si se debe utilizar una dirección IPv6 para un nombre de equipo, entonces todos los ":" (dos puntos) se reemplazan por "_" (guiones bajos), ya que los dos puntos no están permitidos en los nombres de equipo.

> **Attention:**
> Si se realiza el descubrimiento a través de un proxy, actualmente la búsqueda del nombre de equipo todavía se realiza en el servidor Zabbix.

> **Attention:**
> Si ya existe un equipo en la configuración de Zabbix con el mismo nombre que uno recién descubierto, Zabbix añadirá **_N** al nombre del equipo, donde **N** es un número creciente, comenzando por 2.

Uso de macros en mensajes

Descripción general

En los asuntos y textos de los mensajes puede utilizar macros para informar de problemas de manera más eficiente.

Además de una serie de macros integradas, también se admiten macros de usuario y macros de expresión. Está disponible una lista completa de macros admitidas por Zabbix.

Ejemplos

Aquí se muestran ejemplos de cómo puede utilizar macros en los mensajes.

Ejemplo 1

Asunto del mensaje:

```
Problema: {TRIGGER.NAME}
```

Cuando reciba el mensaje, el asunto del mensaje será reemplazado por algo como:

```
Problema: La carga del procesador es demasiado alta en el servidor Zabbix
```

Ejemplo 2

Mensaje:

```
La carga del procesador es: {?last(/zabbix.zabbix.com/system.cpu.load[,avg1])}
```

Cuando reciba el mensaje, el mensaje será reemplazado por algo como:

```
La carga del procesador es: 1.45
```

Ejemplo 3

Mensaje:

```
Último valor: {?last(/{HOST.HOST}/{ITEM.KEY})}
MÁX en 15 minutos: {?max(/{HOST.HOST}/{ITEM.KEY},15m)}
MÍN en 15 minutos: {?min(/{HOST.HOST}/{ITEM.KEY},15m)}
```

Cuando reciba el mensaje, el mensaje será reemplazado por algo como:

```
Último valor: 1.45
MÁX en 15 minutos: 2.33
MÍN en 15 minutos: 1.01
```

Ejemplo 4

Mensaje:

```
http://<ip_o_nombre_del_servidor>/zabbix/tr_events.php?triggerid={TRIGGER.ID}&eventid={EVENT.ID}
```

Cuando reciba el mensaje, contendrá un enlace a la página de *Detalles del evento*, que proporciona información sobre el evento, su disparador y una lista de los últimos eventos generados por el mismo disparador.

Ejemplo 5

Informando sobre valores de varios equipos en una expresión de disparador.

Mensaje:

```
Nombre del problema: {TRIGGER.NAME}
Expresión del disparador: {TRIGGER.EXPRESSION}

1. Valor del elemento en {HOST.NAME1}: {ITEM.VALUE1} ({ITEM.NAME1})
2. Valor del elemento en {HOST.NAME2}: {ITEM.VALUE2} ({ITEM.NAME2})
```

Cuando reciba el mensaje, el mensaje será reemplazado por algo como:

```
Nombre del problema: La carga del procesador es demasiado alta en un equipo local
Expresión del disparador: last(/MiEquipo/system.cpu.load[percpu,avg1])>5 o last(/MiOtroEquipo/system.cpu.l

1. Valor del elemento en MiEquipo: 0.83 (Carga del procesador (promedio de 1 min por núcleo))
2. Valor del elemento en MiOtroEquipo: 5.125 (Carga del procesador (promedio de 1 min por núcleo))
```

Ejemplo 6

Recibir detalles tanto del evento de problema como del evento de recuperación en un mensaje de recuperación:

Mensaje:

```
Problema:

ID de evento: {EVENT.ID}
Valor del evento: {EVENT.VALUE}
Estado del evento: {EVENT.STATUS}
Hora del evento: {EVENT.TIME}
Fecha del evento: {EVENT.DATE}
Antigüedad del evento: {EVENT.AGE}
Reconocimiento del evento: {EVENT.ACK.STATUS}
Historial de actualizaciones del evento: {EVENT.UPDATE.HISTORY}

Recuperación:

ID de evento: {EVENT.RECOVERY.ID}
Valor del evento: {EVENT.RECOVERY.VALUE}
Estado del evento: {EVENT.RECOVERY.STATUS}
Hora del evento: {EVENT.RECOVERY.TIME}
Fecha del evento: {EVENT.RECOVERY.DATE}
Datos operativos: {EVENT.OPDATA}
```

Cuando reciba el mensaje, las macros serán reemplazadas por algo como:

```
Problema:

ID de evento: 21874
Valor del evento: 1
Estado del evento: PROBLEMA
Hora del evento: 13:04:30
Fecha del evento: 2018.01.02
Antigüedad del evento: 5m 0s
Reconocimiento del evento: Sí
Historial de actualizaciones del evento: 2018.01.02 13:05:51 "John Smith (Admin)"
Acciones: reconocido.

Recuperación:
```

```
ID de evento: 21896
Valor del evento: 0
Estado del evento: OK
Hora del evento: 13:10:07
Fecha del evento: 2018.01.02
Datos operativos: El valor actual es 0.83
```

Operaciones de recuperación

Descripción general

Las operaciones de recuperación permiten que se le notifique cuando los problemas se resuelvan.

Tanto los mensajes como los comandos remotos son compatibles en las operaciones de recuperación. Aunque se pueden agregar varias operaciones, la escalada no es compatible: todas las operaciones se asignan a un solo paso y, por lo tanto, se ejecutarán simultáneamente.

Casos de uso

Algunos casos de uso para las operaciones de recuperación son los siguientes:

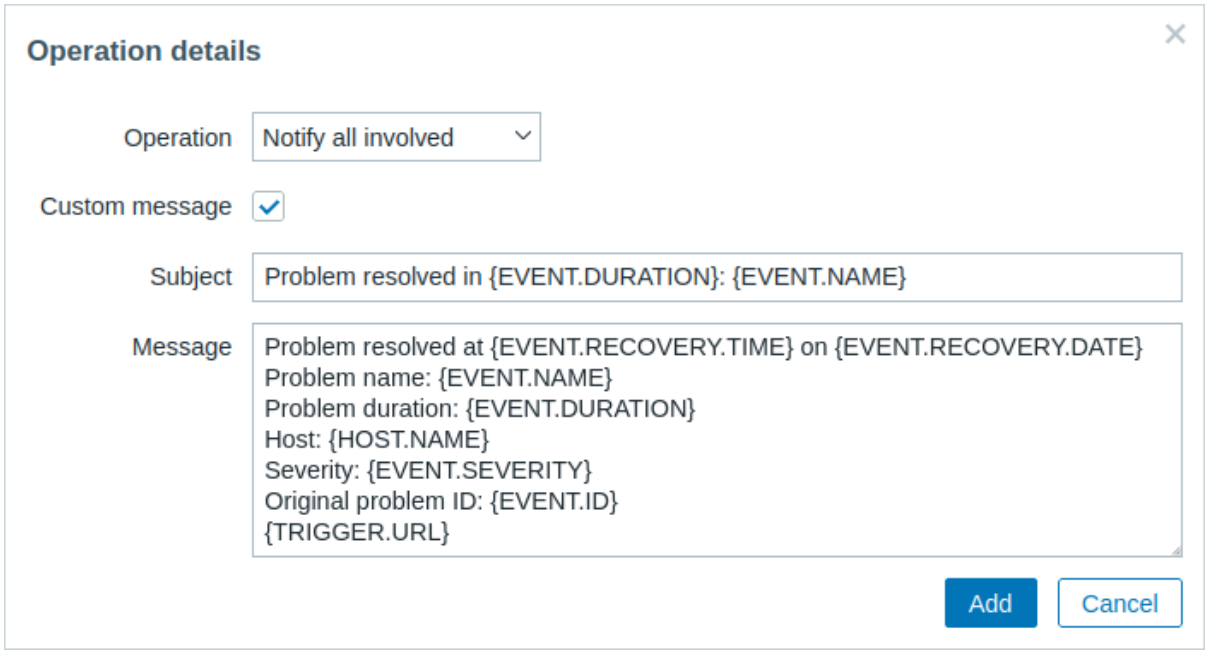
1. Notificar en una recuperación a todos los usuarios que fueron notificados sobre el problema:
   - Seleccione *Notificar a todos los involucrados* como tipo de operación.
2. Tener múltiples operaciones al recuperarse: enviar una notificación y ejecutar un comando remoto:
   - Añada tipos de operación para enviar un mensaje y ejecutar un comando.
3. Abrir un ticket en un sistema externo de helpdesk/tickets y cerrarlo cuando se resuelva el problema:
   - Cree un script externo que se comunique con el sistema de helpdesk.
   - Cree una acción que tenga una operación que ejecute este script y así abra un ticket.
   - Tenga una operación de recuperación que ejecute este script con otros parámetros y cierre el ticket.
   - Utilice la macro {EVENT.ID} para hacer referencia al problema original.

Configuración de una operación de recuperación

Para configurar una operación de recuperación, siga estos pasos:

1. Vaya a la pestaña *Operaciones* en el formulario de configuración de la acción.
2. Para configurar una nueva operación de recuperación, haga clic en *Añadir* en la sección *Operaciones de recuperación*. Para editar una operación existente, haga clic en *Editar* junto a la operación.
3. Configure los detalles de la operación.

Detalles de la operación de recuperación



Hay tres tipos de operación disponibles para los eventos de recuperación:

- **Enviar mensaje**: envía un mensaje de recuperación al usuario especificado;

- **Notificar a todos los involucrados**: envía un mensaje de recuperación a todos los usuarios que fueron notificados sobre el evento de problema;
- **<nombre del comando remoto>**: ejecuta un comando remoto. Los comandos están disponibles para su ejecución si se definieron previamente en los scripts globales con *Operación de acción* seleccionada como su alcance.

A continuación se describen los parámetros de cada tipo de operación. Todos los campos de entrada obligatorios están marcados con un asterisco rojo. Cuando haya terminado, haga clic en *Añadir* para agregar la operación a la lista de *Operaciones de recuperación*.

Note:
Tenga en cuenta que, si el mismo destinatario está definido en varios tipos de operación sin especificar *Mensaje personalizado*, no se envían notificaciones duplicadas.

Tipo de operación: enviar mensaje

Parámetro	Descripción
Enviar a grupos de usuarios	Seleccione los grupos de usuarios a los que enviar el mensaje de recuperación. El grupo de usuarios debe tener al menos permisos de "lectura" permisos sobre el equipo para ser notificado.
Enviar a usuarios	Seleccione los usuarios a los que enviar el mensaje de recuperación. El usuario debe tener al menos permisos de "lectura" permisos sobre el equipo para ser notificado.
Enviar al tipo de medio	Enviar el mensaje de recuperación por defecto a todos los tipos de medio disponibles (configurados y habilitados) o solo a uno específico.
Mensaje personalizado	Si se selecciona, se puede definir un mensaje personalizado.
Asunto	Asunto del mensaje personalizado. El asunto puede contener macros.
Mensaje	El mensaje personalizado. El mensaje puede contener macros.

Tipo de operación: comando remoto

Parámetro	Descripción
Lista de destinos	Seleccione los destinos en los que se ejecutará el comando: **Host actual**: el comando se ejecuta en el host del trigger que provocó el evento de problema. Esta opción no funcionará si hay varios hosts en el trigger. **Host**: seleccione los hosts en los que se ejecutará el comando. **Grupo de hosts**: seleccione los grupos de hosts en los que se ejecutará el comando. Especificar un grupo de hosts principal selecciona implícitamente todos los grupos de hosts anidados. Por lo tanto, el comando remoto también se ejecutará en los hosts de los grupos anidados. Un comando en un host se ejecuta solo una vez, incluso si el host coincide más de una vez (por ejemplo, desde varios grupos de hosts; individualmente y desde un grupo de hosts). La lista de destinos no tiene sentido si el comando se ejecuta en el Zabbix server. Seleccionar más destinos en este caso solo hace que el comando se ejecute más veces en el server. Tenga en cuenta que, para los scripts globales, la selección de destinos también depende de la configuración de *Grupo de hosts* en la configuración del script global.

Tipo de operación: notificar a todos los involucrados

Parámetro		Descripción
Mensaje personalizado		Si se selecciona, se puede definir un mensaje personalizado.
	Asunto	Asunto del mensaje personalizado. El asunto puede contener macros.
	Mensaje	El mensaje personalizado. El mensaje puede contener macros.

Operaciones de actualización

Resumen

Las operaciones de actualización están disponibles en acciones con las siguientes fuentes de eventos:

- *Triggers* - cuando los problemas son actualizados por otros usuarios, es decir, se les añade un comentario, se reconocen, se cambia su severidad, se cierran (manualmente);
- *Services* - cuando la severidad de un servicio ha cambiado, pero el servicio aún no se ha recuperado.

Tenga en cuenta que los usuarios no reciben notificaciones sobre sus propias actualizaciones.

Tanto los mensajes como los comandos remotos son compatibles con las operaciones de actualización. Aunque se pueden añadir varias operaciones, la escalada no es compatible: todas las operaciones se asignan a un único paso y, por lo tanto, se ejecutarán simultáneamente.

Configuración de una operación de actualización

Para configurar una operación de actualización, siga estos pasos:

1. Vaya a la pestaña *Operaciones* en el formulario de configuración de la acción.
2. Para configurar una nueva operación de actualización, haga clic en *Añadir* en la sección *Operaciones de actualización*. Para editar una operación existente, haga clic en *Editar* junto a la operación.
3. Configure los detalles de la operación.

Detalles de la operación de actualización

Las operaciones de actualización ofrecen el mismo conjunto de parámetros que las operaciones de recuperación.

Escalaciones

Resumen

Con las escalaciones puede crear escenarios personalizados para enviar notificaciones o ejecutar comandos remotos.

En términos prácticos, esto significa que:

- Los usuarios pueden ser informados sobre nuevos problemas de inmediato.
- Las notificaciones pueden repetirse hasta que el problema se resuelva.
- El envío de una notificación puede retrasarse.
- Las notificaciones pueden escalarse a otro grupo de usuarios "superior".
- Los comandos remotos pueden ejecutarse de inmediato o cuando un problema no se resuelve durante un período prolongado.

Las acciones se escalan en función del **paso de escalación**. Cada paso tiene una duración determinada.

Puede definir tanto la duración predeterminada como una duración personalizada de un paso individual. La duración mínima de un paso de escalación es de 60 segundos.

Puede iniciar acciones, como enviar notificaciones o ejecutar comandos, desde cualquier paso. El paso uno es para acciones inmediatas. Si desea retrasar una acción, puede asignarla a un paso posterior. Para cada paso, se pueden definir varias acciones.

El número de pasos de escalación no está limitado.

Las escalaciones se definen al configurar una operación. Las escalaciones son compatibles solo con operaciones de problema, no de recuperación.

Aspectos diversos del comportamiento de las escalaciones

Consideremos qué sucede en diferentes circunstancias si una acción contiene varios pasos de escalación.

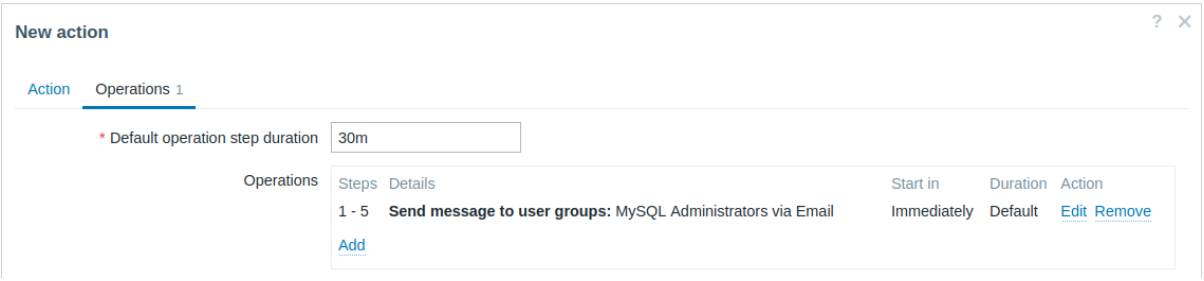
Situación	Comportamiento
El host en cuestión entra en mantenimiento después de que se envía la notificación inicial del problema	Según la configuración *Pausar las operaciones para problemas suprimidos* en la configuración de la acción, todos los pasos de escalación restantes se ejecutan con un retraso causado por el período de mantenimiento o sin retraso. Un período de mantenimiento no cancela las operaciones.
*El período definido en la condición de acción **Período de tiempo** termina después de que se envía la notificación inicial*	Se ejecutan todos los pasos de escalación restantes. La condición *Período de tiempo* no puede detener las operaciones; tiene efecto sobre cuándo se inician o no las acciones, no sobre las operaciones.
Un problema comienza durante el mantenimiento y continúa (no se resuelve) después de que termina el mantenimiento	Según la configuración *Pausar las operaciones para problemas suprimidos* en la configuración de la acción, todos los pasos de escalación se ejecutan desde el momento en que termina el mantenimiento o inmediatamente.
Un problema comienza durante un mantenimiento sin datos y continúa (no se resuelve) después de que termina el mantenimiento	Debe esperar a que se active el trigger antes de que se ejecuten todos los pasos de escalación.
Se producen diferentes escalaciones en rápida sucesión y se superponen	La ejecución de cada nueva escalación reemplaza la escalación anterior, pero al menos un paso de escalación siempre se ejecuta en la escalación anterior. Este comportamiento es relevante en las acciones sobre eventos creados con CADA evaluación del problema del trigger.
Durante una escalación en curso (por ejemplo, mientras se envía un mensaje), según cualquier tipo de evento: - la acción está deshabilitada Según un evento de trigger: - el trigger está deshabilitado - el host o el item está deshabilitado Según un evento interno sobre triggers: - el trigger está deshabilitado Según un evento interno sobre items/reglas de descubrimiento de bajo nivel: - el item está deshabilitado - el host está deshabilitado	El mensaje en curso se envía y, a continuación, se envía un mensaje adicional sobre la escalación. El mensaje posterior incluirá el texto de cancelación al principio del cuerpo del mensaje (*NOTA: Escalación cancelada*), indicando el motivo (por ejemplo, *NOTA: Escalación cancelada: acción '<Action name>' deshabilitada*). De esta forma, el destinatario recibe información de que la escalación se ha cancelado y no se ejecutarán más pasos. Este mensaje se envía a todos los que recibieron las notificaciones anteriormente. El motivo de la cancelación también se registra en el archivo de registro del server (a partir de Nivel de depuración 3=Advertencia). Tenga en cuenta que el mensaje *Escalación cancelada* también se envía si las operaciones han finalizado, pero hay operaciones de recuperación configuradas que aún no se han ejecutado.
Durante una escalación en curso (por ejemplo, mientras se envía un mensaje), se elimina la acción	No se envían más mensajes. La información se registra en el archivo de registro del server (a partir de Nivel de depuración 3=Advertencia), por ejemplo: `escalation canceled: action id:334 deleted`

Ejemplos de escalado

Ejemplo 1

Envío de una notificación repetida cada 30 minutos (5 veces en total) a un grupo "Administradores de MySQL". Para configurar:

- En la pestaña *Operaciones*, establezca la *Duración por defecto del paso de operación* en "30m" (30 minutos).
- Establezca los *Pasos* de escalado desde "1" hasta "5".
- Seleccione el grupo "Administradores de MySQL" como destinatarios del mensaje.



Las notificaciones se enviarán a las 0:00, 0:30, 1:00, 1:30, 2:00 horas después de que comience el problema (a menos que, por supuesto, el problema se resuelva antes).

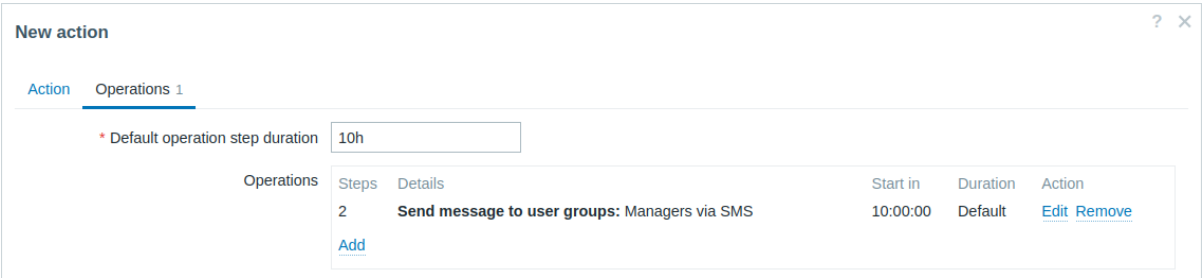
Si el problema se resuelve y se configura un mensaje de recuperación, se enviará a quienes hayan recibido al menos un mensaje de problema dentro de este escenario de escalado.

Note:
Si el disparador que generó una escalada activa es deshabilitado, Zabbix envía un mensaje informativo al respecto a todos aquellos que ya han recibido notificaciones.

Ejemplo 2

Envío de una notificación retrasada sobre un problema prolongado. Para configurar:

- En la pestaña *Operaciones*, establezca la *Duración por defecto del paso de operación* en ”10h” (10 horas).
- Establezca los *Pasos* de la escalada desde ”2” hasta ”2”.



Solo se enviará una notificación en el Paso 2 del escenario de escalada, o 10 horas después de que comience el problema.

Puede personalizar el texto del mensaje a algo como ”El problema tiene más de 10 horas”.

Ejemplo 3

Escalando el problema al Jefe.

En el primer ejemplo anterior configuramos el envío periódico de mensajes a los administradores de MySQL. En este caso, los administradores recibirán cuatro mensajes antes de que el problema se escale al responsable de la base de datos. Tenga en cuenta que el responsable recibirá un mensaje solo en caso de que el problema aún no haya sido reconocido, suponiendo que nadie está trabajando en él.

Detalles de la Operación 2:

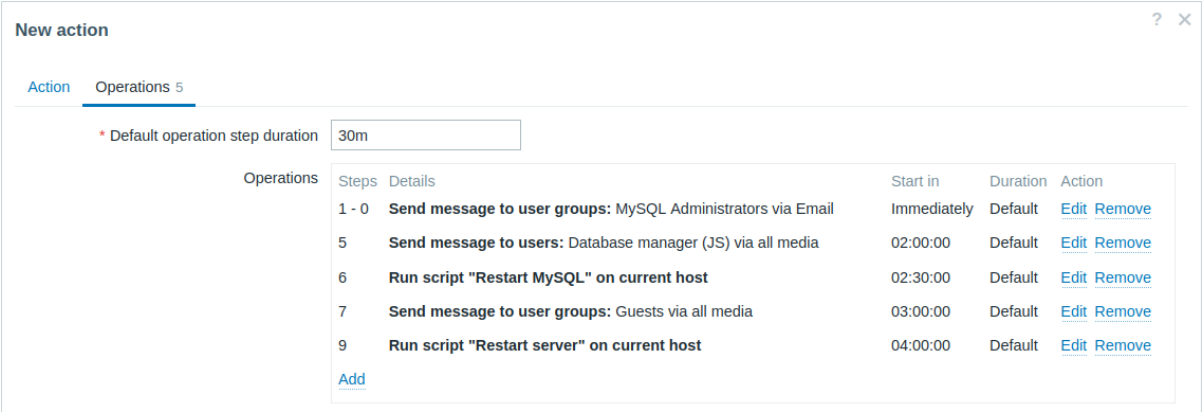
Observe el uso de la macro {ESC.HISTORY} en el mensaje personalizado. La macro contendrá información sobre todos los pasos ejecutados previamente en esta escalada, como notificaciones enviadas y comandos ejecutados.

Ejemplo 4

Un escenario más complejo. Después de múltiples mensajes a los administradores de MySQL y la escalada al gerente, Zabbix intentará reiniciar la base de datos MySQL. Esto ocurrirá si el problema existe durante 2:30 horas y no ha sido reconocido.

Si el problema aún existe, después de otros 30 minutos Zabbix enviará un mensaje a todos los usuarios invitados.

Si esto no ayuda, después de otra hora Zabbix reiniciará el servidor con la base de datos MySQL (segundo comando remoto) usando comandos IPMI.



Ejemplo 5

Una escalada con varias operaciones que tienen rangos de pasos superpuestos e intervalos personalizados. La duración por defecto de los pasos de la operación es de 30 minutos.

Las notificaciones se enviarán de la siguiente manera:

- A los administradores de MySQL a las 0:00, 0:30, 1:00 y 1:30 después de que comience el problema.
- Al administrador de la base de datos a las 2:00 y 2:10 (la duración de paso personalizada más corta de 10 minutos definida en la operación posterior anula la duración de paso más larga de 1 hora configurada aquí, como se describe en Detalles de la operación para *Duración del paso* cuando los pasos se superponen).
- A los administradores de Zabbix a las 2:00, 2:10 y 2:20 después de que comience el problema (se aplica la duración de paso personalizada de 10 minutos).
- A los usuarios invitados a las 4:00 después de que comience el problema (la duración de paso por defecto de 30 minutos entra en vigor entre los pasos 8 y 11).

Recepción de notificaciones sobre items no compatibles

Resumen

Es posible recibir notificaciones sobre items no compatibles en Zabbix.

Forma parte del concepto de eventos internos en Zabbix, lo que permite a los usuarios ser notificados en estas ocasiones. Los eventos internos reflejan un cambio de estado:

- cuando los items pasan de 'normal' a 'unsupported' (y viceversa);
- cuando los triggers pasan de 'normal' a 'unknown' (y viceversa);
- cuando las reglas de descubrimiento de bajo nivel pasan de 'normal' a 'unsupported' (y viceversa).

Esta sección presenta una guía práctica para **recibir notificaciones** cuando un item pasa a estar unsupported.

Configuración

En general, el proceso de configuración de la notificación debería resultar familiar para quienes hayan configurado alertas en Zabbix anteriormente.

Paso 1

Configure algún medio, como correo electrónico, SMS o script, para usarlo en las notificaciones. Consulte las secciones correspondientes del manual para realizar esta tarea.

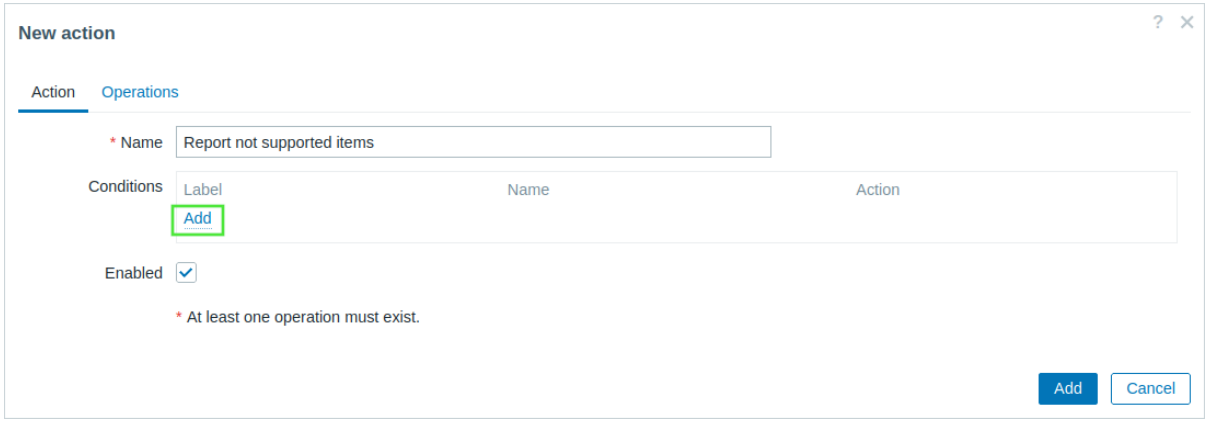
Attention:
Para notificar sobre eventos internos se utiliza la gravedad predeterminada ('No clasificado'), así que déjela marcada al configurar los medios de usuario si desea recibir notificaciones de eventos internos.

Paso 2

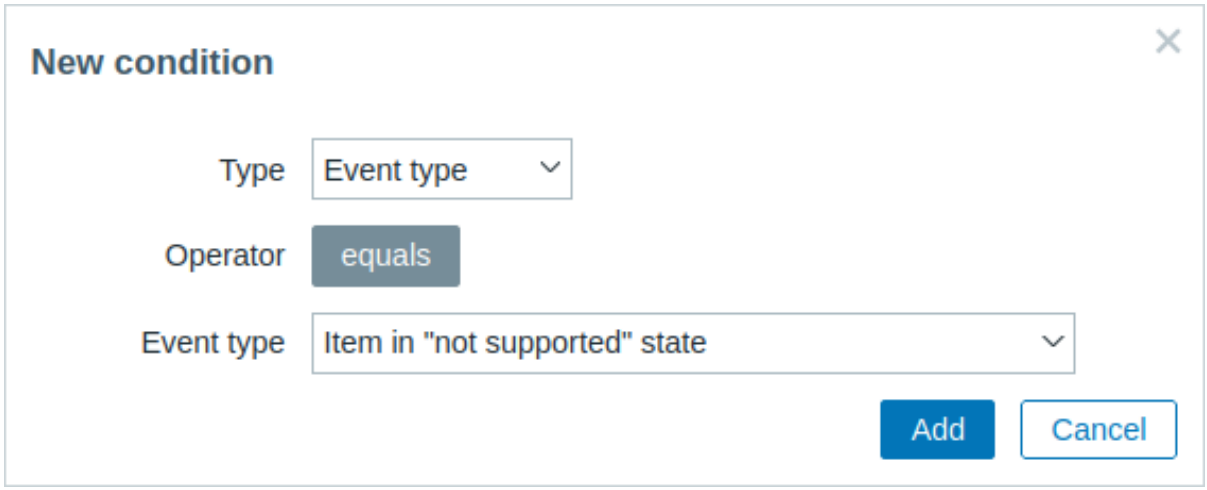
1. Vaya a *Alerts* > *Actions* > *Internal actions*.
2. Haga clic en *Create action* en la esquina superior derecha de la página para abrir un formulario de configuración de acción.

Paso 3

1. En la pestaña *Action*, introduzca un nombre para la acción.
2. Haga clic en *Add* en el bloque *Conditions* para añadir una nueva condición.



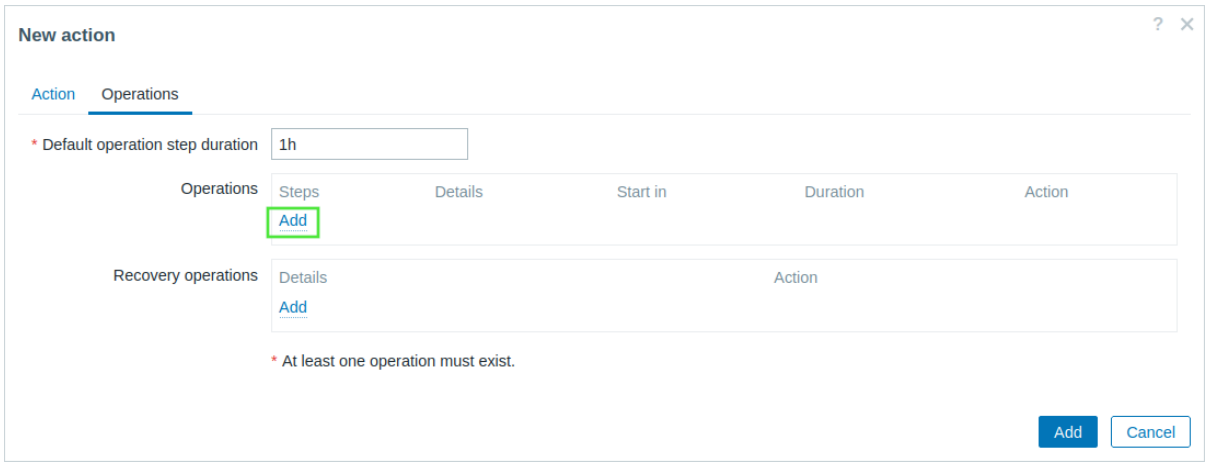
3. En la ventana emergente *New condition*, seleccione ”Event type” como tipo de condición y, a continuación, seleccione ”Item in ’not supported’ state” como tipo de evento.



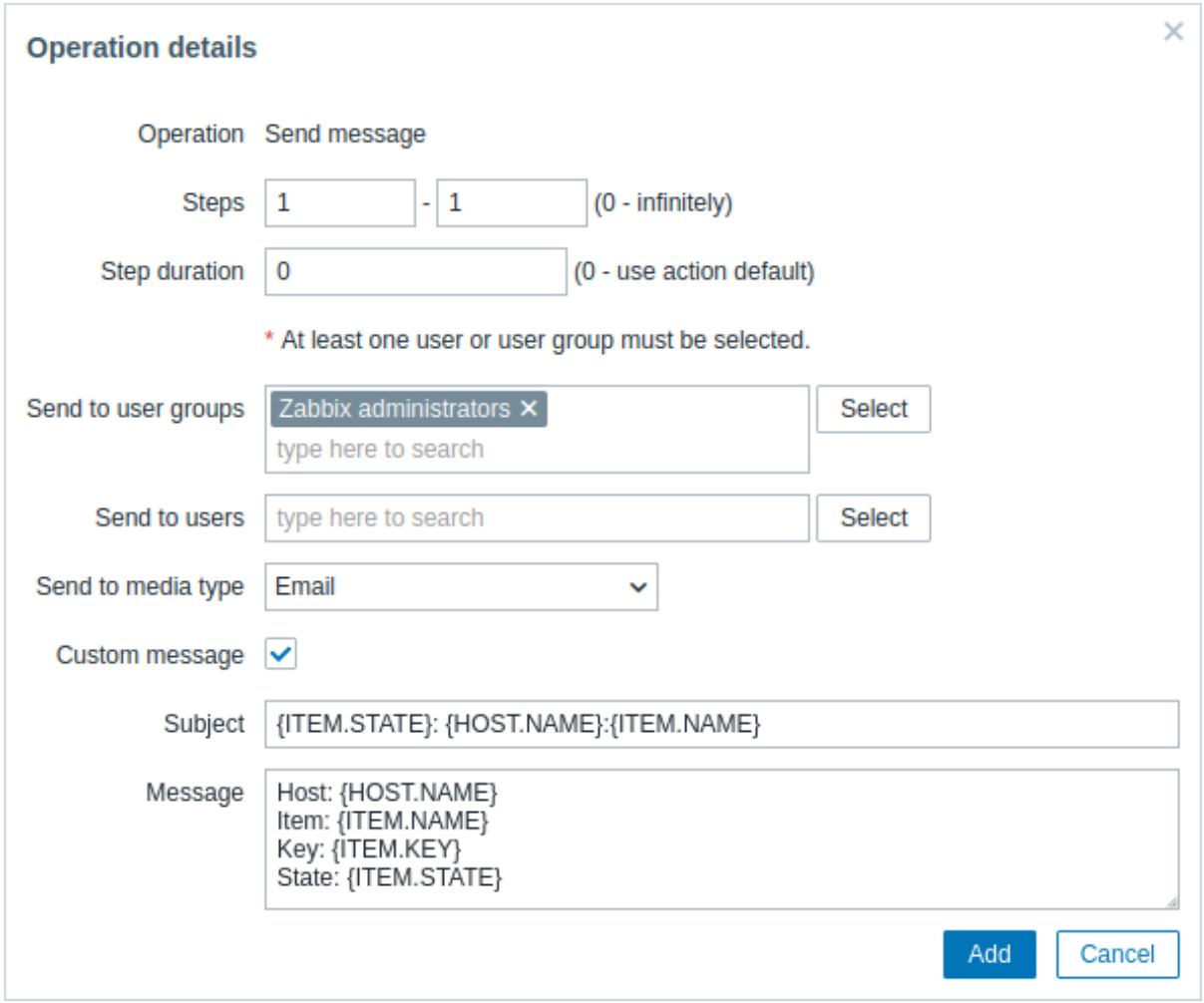
4. Haga clic en *Add* para incluir la condición en el bloque *Conditions*.

Paso 4

1. En la pestaña *Operations*, haga clic en *Add* en el bloque *Operations* para añadir una nueva operación.



2. Seleccione algunos destinatarios del mensaje (grupos de usuarios/usuarios) y el tipo de medio (o ”All”) que se utilizará para la entrega.
3. Marque la casilla *Custom message* si desea introducir el asunto o el contenido personalizados del mensaje del problema.



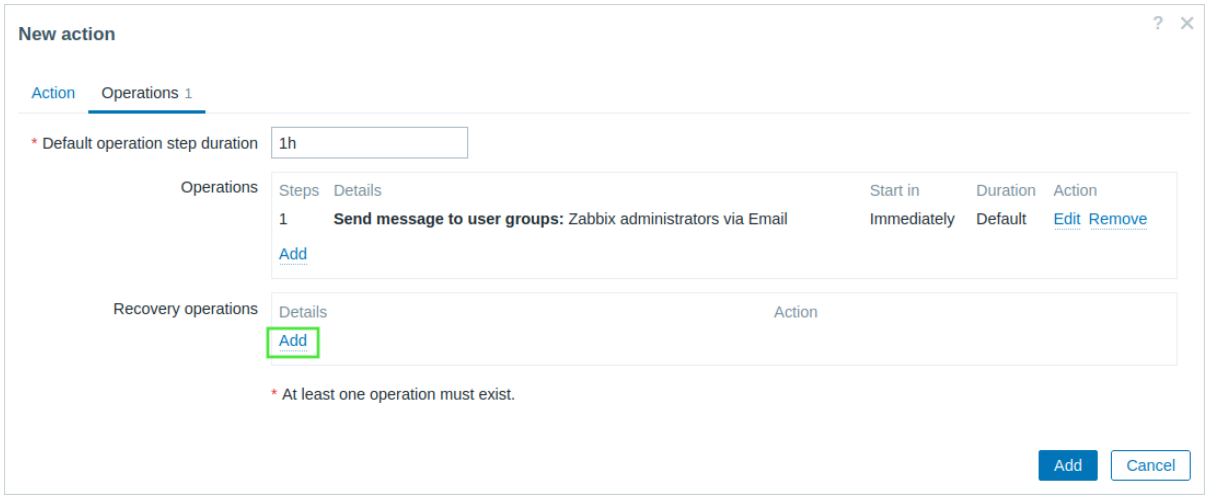
4. Haga clic en *Add* para incluir la operación en el bloque *Operations*.

Si desea recibir más de una notificación, establezca la duración del paso de la operación (intervalo entre los mensajes enviados) y añada otro paso.

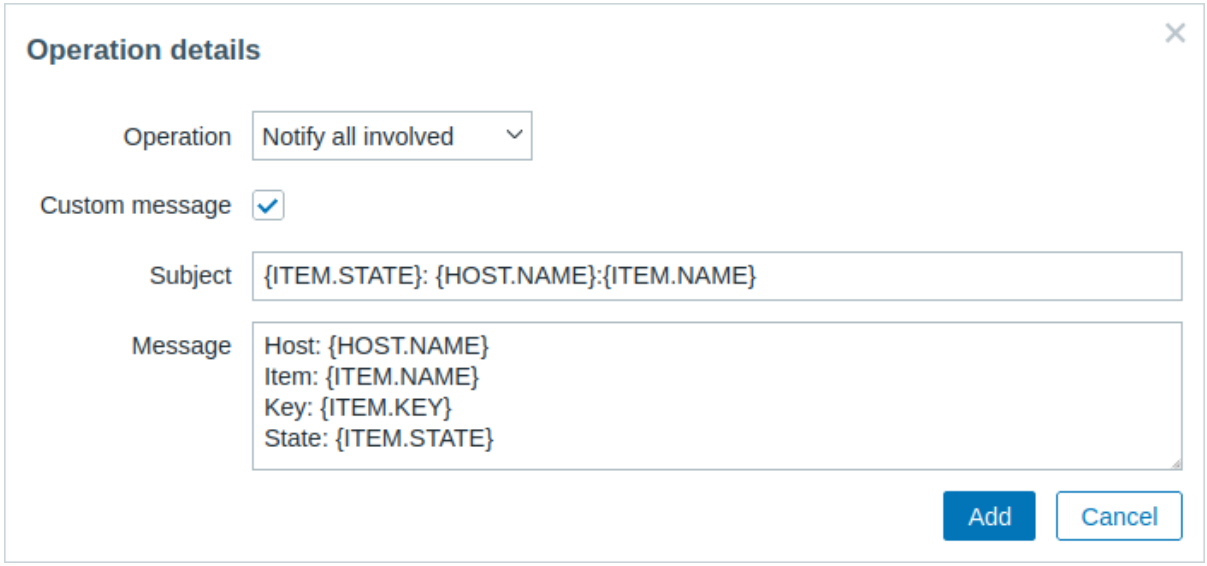
Paso 5

El bloque *Operaciones de recuperación* permite configurar una notificación de recuperación cuando un item vuelve al estado normal.

1. Haga clic en *Añadir* en el bloque *Operaciones de recuperación* para añadir una nueva operación de recuperación.



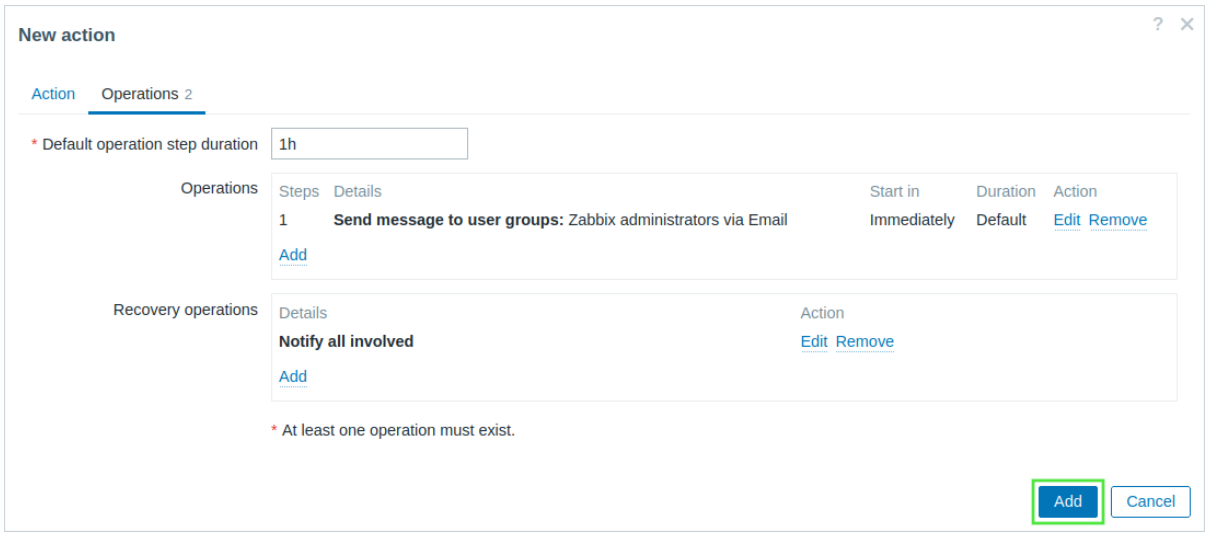
2. Seleccione el tipo de operación ”Notificar a todos los involucrados”.
3. Seleccione la casilla *Mensaje personalizado* si desea introducir el asunto o el contenido personalizados del mensaje de recuperación.



4. Haga clic en *Añadir* en la ventana emergente *Detalles de la operación* para incluir la operación en el bloque *Operaciones de recuperación*.

Paso 6

Cuando haya terminado, haga clic en el botón *Add* en la parte inferior del formulario.



¡Y eso es todo, ya ha terminado! Ahora puede esperar recibir su primera notificación de Zabbix si algún item pasa a estado no compatible.

Macros

Descripción general

Zabbix admite varias macros integradas que pueden utilizarse en distintas situaciones. Estas macros son variables identificadas mediante una sintaxis específica:

`{MACRO}`

Las macros se resuelven en un valor específico según el contexto.

El uso eficaz de las macros permite ahorrar tiempo y hacer que la configuración de Zabbix sea más transparente.

En uno de los usos habituales, una macro puede utilizarse en un template. Por ejemplo, un trigger en un template puede llamarse "La carga del procesador es demasiado alta en {HOST.NAME}". Cuando el template se aplica al host, como el server de Zabbix, el nombre se resolverá como "La carga del procesador es demasiado alta en el server de Zabbix" cuando el trigger se muestre en la sección Monitoring.

Las macros pueden utilizarse en los parámetros de las claves de los item. Una macro puede utilizarse solo para una parte del parámetro, por ejemplo, `item.key[server_{HOST.HOST}_local]`. No es necesario poner el parámetro entre comillas dobles,

ya que Zabbix se encargará de cualquier símbolo especial ambiguo, si está presente en la macro resuelta.

En Zabbix existen otros tipos de macros.

Zabbix admite las siguientes macros:

- {MACRO} - macro integrada (consulte la lista completa)
- {<macro>.<func>(<params>)} - funciones de macros
- {$MACRO} - macro definida por el usuario, opcionalmente con contexto
- {#MACRO} - macro para el descubrimiento de bajo nivel
- {?EXPRESSION} - macro de expresión

Funciones de macro

Resumen

Las funciones de macro ofrecen la posibilidad de personalizar los valores de macro (por ejemplo, acortar o extraer subcadenas específicas), lo que facilita su uso.

La sintaxis de una función de macro es:

```
{macro.func(params)}
```

donde

- **macro** - la macro que se va a personalizar;
- **func** - la función que se va a aplicar (consulte funciones compatibles);
- **params** - una lista de parámetros de función separada por comas, que debe ir entre **comillas dobles** si:
  - comienza con un espacio o comillas dobles;
  - contiene paréntesis de cierre o una coma.

Por ejemplo:

```
{{TIME}.fmttime(format,time_shift)}
{{ITEM.VALUE}.regsub(pattern, output)}
{{$USERMACRO}.regsub(pattern, output)}
{{#LLDMACRO}.regsub(pattern, output)}
```

Las funciones de macro son compatibles con

- Macros integradas
- Macros de usuario
- Macros de descubrimiento de bajo nivel
- Macros de expresión

Las funciones de macro se pueden usar en todas las ubicaciones que admiten las macros enumeradas. Esto se aplica salvo que se indique explícitamente que solo se espera una macro (por ejemplo, al configurar macros de host o filtros de una regla de descubrimiento de bajo nivel).

Se admite una sola función por macro; no se admiten varias funciones de macro en cadena.

> **Note:**
> Consulte los ejemplos de escape para los casos en que las funciones de macro se usan dentro de otros contextos (función, clave de item, otra macro, etc).

Funciones compatibles

Las funciones se muestran sin información adicional.
Haga clic en una función para ver todos los detalles.

Función	Descripción
btoa	Codificación del valor de una macro en formato Base64.
fmtnum	Formato de números para controlar la cantidad de dígitos que se muestran después del separador decimal.
fmttime	Formato de hora.
htmldecode	Decodificación del valor de una macro desde la codificación HTML.
htmlencode	Codificación del valor de una macro en formato HTML.
iregsub	Extracción de una subcadena mediante una coincidencia de expresión regular (sin distinguir mayúsculas de minúsculas).

Función	Descripción
jsonpath	Devuelve el resultado de JSONPath.
lowercase	Transformación de los caracteres del valor de una macro a minúsculas.
regrepl	Reemplazo de un carácter o una subcadena en el valor de una macro.
regsub	Extracción de una subcadena mediante una coincidencia de expresión regular (distinguiendo mayúsculas de minúsculas).
tr	Transliteración de los caracteres del valor de una macro.
uppercase	Transformación de los caracteres del valor de una macro a mayúsculas.
urldecode	Decodificación del valor de una macro desde la codificación URL.
urlencode	Codificación del valor de una macro en formato URL.
xmlxpath	Devuelve el resultado de XML XPath.

Detalles de la función

Los parámetros opcionales de la función se indican mediante < >.

btoa

Codificación de un valor de macro en codificación Base64. La codificación Base64 es un método para representar datos binarios como texto, útil para almacenar y transmitir de forma segura contenido binario a través de protocolos basados en texto.

Ejemplo:

```
{{ITEM.VALUE}.btoa()} - codificará en Base64 un valor como "zabbix" en "emFiYml4"
```

fmtnum(dígitos)

Formato de número para controlar la cantidad de dígitos impresos después del punto decimal.

Parámetros:

- **dígitos**: el número de dígitos después del punto decimal. Rango válido: 0-20. No se producirán ceros finales.

Ejemplos:

```
{{ITEM.VALUE}.fmtnum(2)} - devolverá "24.35" a partir de un valor recibido de "24.3483523"
{{ITEM.VALUE}.fmtnum(0)} - devolverá "24" a partir de un valor recibido de "24.3483523"
```

fmttime(format,<time_shift>)

Formato de hora.
 Tenga en cuenta que esta función se puede usar con macros que resuelven a un valor en uno de los siguientes formatos de hora:

- `hh:mm:ss`
- `yyyy-mm-ddThh:mm:ss[tz]` (estándar ISO8601)
- marca de tiempo UNIX

Parámetros:

- **format** - cadena de formato obligatoria, compatible con el formato de la función `strftime`;
- **time_shift** (opcional) - desplazamiento de tiempo aplicado a la hora antes del formateo; debe comenzar con `-<N><time_unit>` o `+<N><time_unit>`, donde:
  - `N` - el número de unidades de tiempo que se van a sumar o restar;
  - `time_unit` - h (hora), d (día), w (semana), M (mes) o y (año).

Comentarios:

- El parámetro `time_shift` admite operaciones de tiempo de varios pasos y puede incluir `/<time_unit>` para desplazar al inicio de la unidad de tiempo (`/d` - medianoche, `/w` - primer día de la semana (lunes), `/M` - primer día del mes, etc.). Ejemplos: `-1w` - exactamente 7 días atrás; `-1w/w` - lunes de la semana anterior; `-1w/w+1d` - martes de la semana anterior.
- Las operaciones de tiempo se calculan de izquierda a derecha sin prioridades. Por ejemplo, `-1M/d+1h/w` se analizará como `((-1M/d)+1h)/w`.

Ejemplos:

```
{{TIMESTAMP}.fmttime(%B)} - devolverá "October" a partir de un valor recibido de "1633098961"
{{TIMESTAMP}.fmttime(%d %B,-1M/M)} - devolverá "1 September" a partir de un valor recibido de "1633098961"
{{EVENT.TIMESTAMP}.fmttime("%x %X")} - devolverá la representación local de la fecha y la hora; por ejempl
{{EVENT.TIME}.fmttime("%x %X")} - devolverá la representación local de la fecha y la hora; por ejemplo, "0
```

htmldecode

Decodifica un valor de macro desde la codificación HTML.

Se admiten los siguientes caracteres:

Valor	Valor decodificado
&	&
>	>
"	"
'	'
'	'

Ejemplo:

```
{{ITEM.VALUE}.htmldecode()} - decodificará un valor HTML como "<" en "<"
```

htmlencode

Codifica un valor de macro en codificación HTML.

Se admiten los siguientes caracteres:

Valor	Valor codificado
&	&
>	>
"	"
'	'

Ejemplo:

```
{{ITEM.VALUE}.htmlencode()} - codificará en HTML un carácter como "<" en "<"
```

iregsub(patrón,salida)

Extracción de subcadenas mediante coincidencia de expresión regular (sin distinguir mayúsculas/minúsculas).

Parámetros:

- **patrón**: la expresión regular a buscar;
- **salida**: las opciones de salida. Se admiten los marcadores de posición **\1 - \9** para capturar grupos. **\0** devuelve el texto coincidente.

Comentarios:

- Si no hay coincidencia para la expresión regular, la función devuelve una cadena vacía.
- Si el patrón de la función es una expresión regular incorrecta, entonces la macro se evalúa como 'UNKNOWN' (excepto para macros de bajo nivel de descubrimiento, en cuyo caso la función será ignorada y la macro permanecerá sin resolver).
- Las referencias a grupos de captura inexistentes en la cadena de reemplazo se sustituyen por una cadena vacía.

Ejemplo:

```
{{ITEM.VALUE}.iregsub("fail|error|fault|problem","ERROR")} - se resolverá como "ERROR" si se reciben las s
```

jsonpath(pattern,<default>)

Devuelve el resultado de JSONPath.
 Tipos de valores compatibles: *String*, *Text*, *Log*.

Parámetros:

- **pattern**: el patrón obligatorio;
- **default**: el valor alternativo opcional si la consulta JSONPath no devuelve datos. Ten en cuenta que se devuelve un error en caso de otros errores.

Comentarios:

- La función devuelve un error si:
  - El JSON no es válido.

- La consulta JSONPath falla.
- El resultado de JSONPath no contiene datos y no se proporciona el parámetro `default`.

Ejemplo:

```
{{ITEM.VALUE}.jsonpath("$.columns.servicename", "NA")}
```

minúsculas

Transformación de todos los caracteres del valor de la macro a minúsculas. Funciona con conjuntos de caracteres de un solo byte (como ASCII) y no admite UTF-8.

Ejemplo:

```
{{ITEM.VALUE}.lowercase()} - transformará un valor como "Zabbix SERVER" en "zabbix server" (minúsculas)
```

regrepl(pattern,replacement,<pattern2>,<replacement2>,...)

Reemplazo de carácter/subcadena en el valor de macro.

Parámetros:

- **pattern** - la expresión regular que se debe coincidir;
- **replacement** - la cadena de reemplazo. **\1 - \9** se admiten como marcadores de posición en las cadenas de reemplazo para capturar grupos.

Comentarios:

- Todas las apariciones que coincidan con el patrón se reemplazan.
- Los patrones y reemplazos se procesan secuencialmente, y cada par posterior se aplica de acuerdo con el resultado del reemplazo anterior.
- Las referencias a grupos de captura inexistentes en la cadena de reemplazo se reemplazan con una cadena vacía.

Ejemplos:

```
{{ITEM.VALUE}.regrepl("oldParam", "newParam")} - reemplazará "oldParam" por "newParam"
{{ITEM.VALUE}.regrepl("([^a-z])","\\\1")} - todos los caracteres que no sean letras se escaparán con una b
{$THRESHOLD:"{{#FSNAME}.regrepl(\"\\$\",\"\")}"} - eliminará una barra invertida final (por ejemplo, para
{{ITEM.VALUE}.regrepl("_v1\.0", "_v2.0", "\(final\)", "")} - reemplazará varias partes en el valor del ite
```

regsub(patrón,salida)

Extracción de subcadenas mediante una coincidencia de expresión regular (sensible a mayúsculas y minúsculas).

Parámetros:

- **patrón**: la expresión regular a coincidir;
- **salida**: las opciones de salida. Se admiten los marcadores de posición **\1 - \9** para capturar grupos. **\0** devuelve el texto coincidente.

Comentarios:

- Si no hay coincidencia para la expresión regular, la función devuelve una cadena vacía.
- Si el patrón de la función es una expresión regular incorrecta, entonces la macro se evalúa como 'UNKNOWN' (excepto para macros de bajo nivel, en cuyo caso la función será ignorada y la macro permanecerá sin resolver).
- Las referencias a grupos de captura inexistentes en la cadena de reemplazo se reemplazan por una cadena vacía.

Ejemplos:

```
{{ITEM.VALUE}.regsub("^([0-9]+)", ID de problema: \1)} - se resolverá como "ID de problema: 123" si se rec
{{ITEM.VALUE}.regsub("fail|error|fault|problem","ERROR")} - se resolverá como "ERROR" si se reciben las su
```

Vea más ejemplos.

tr(caracteres,reemplazo)

Transliteración de los caracteres del valor de la macro.

- **caracteres**: el conjunto de caracteres a reemplazar;
- **reemplazo**: el conjunto de caracteres de reemplazo correspondientes posicionalmente.

Ejemplos:

```
{{ITEM.VALUE}.tr(abc, xyz)} - reemplazará todas las apariciones de "a" por "x", "b" por "y", "c" por "z"
{{ITEM.VALUE}.tr(abc, xyzq)} - reemplazará todas las apariciones de "a" por "x", "b" por "y", "c" por "z"
{{ITEM.VALUE}.tr(abcde, xyz)} - reemplazará todas las apariciones de "a" por "x", "b" por "y", "c" por "z"
```

```
{{ITEM.VALUE}.tr("\\\'", "\/\"")} - reemplazará todas las apariciones de barra invertida por barra, comill
{{ITEM.VALUE}.tr(A-Z,a-z)} - convertirá todas las letras a minúsculas
{{ITEM.VALUE}.tr(0-9a-z,*)} - reemplazará todos los números y letras minúsculas por "*"
{{ITEM.VALUE}.tr(0-9,ab)} - reemplazará todas las apariciones de 0 por "a", y todas las apariciones de 1,
{{ITEM.VALUE}.tr(0-9abcA-L,*)} - reemplazará todos los números, los caracteres "abc" y el rango A-L por "*
{{ITEM.VALUE}.tr("\n","*")} - reemplazará las apariciones de fin de línea por *
{{ITEM.VALUE}.tr("e", "\n")} - reemplazará todas las "e" por fin de línea
```

Para incluir caracteres literales:

```
barra invertida - debe escaparse como \\
comilla simple - debe escaparse como \'
comilla doble - debe escaparse como \"
```

Secuencias de escape soportadas con barra invertida:

```
\\\\ => \\ - doble barra invertida a una sola barra invertida
\\a  => \a - alerta
\\b  => \b - retroceso
\\f  => \f - salto de formulario
\\n  => \n - nueva línea
\\r  => \r - retorno
\\t  => \t - tabulación horizontal
\\v  => \v - tabulación vertical
```

mayúsculas

Transformación de todos los caracteres del valor de la macro a mayúsculas. Funciona con conjuntos de caracteres de un solo byte (como ASCII) y no admite UTF-8.

Ejemplo:

```
{{ITEM.VALUE}.uppercase()} - transformará un valor como "Zabbix Server" en "ZABBIX SERVER" (mayúsculas)
```

urldecode

Decodifica un valor de macro desde la codificación de URL.

Ejemplo:

```
{{ITEM.VALUE}.urldecode()} - decodificará un valor como "%2F" en "/"
```

urlencode

Codifica el valor de una macro en codificación URL.

Ejemplo:

```
{{ITEM.VALUE}.urlencode()} - codificará un carácter como "/" en "%2F"
```

xmlxpath(pattern,<default>)

Devuelve el resultado de XPath del XML.
 Tipos de valores compatibles: *String*, *Text*, *Log*.

Parámetros:

- **pattern**: el patrón obligatorio;
- **default**: el valor alternativo opcional si la consulta XPath no devuelve datos. Tenga en cuenta que, en caso de otros errores, se devuelve un error.

Comentarios:

- La función devuelve un error si:
  - El XML no es válido.
  - La consulta XPath falla.
  - El resultado de XPath no contiene datos y no se proporciona el parámetro `default`.
  - Zabbix se compiló sin compatibilidad con XML.

Ejemplo:

```
{{ITEM.VALUE}.xmlxpath("/response/error/status","unknown")}
```

Ejemplos adicionales

La siguiente tabla muestra más ejemplos de uso de funciones de macros.

{#IFALIAS} es una macro de LLD y solo se define en contextos de descubrimiento de bajo nivel (reglas de descubrimiento, prototipos y los items/triggers creados a partir de ellos). Usarla fuera de LLD dejará el token sin expandir.

Función de macro	Valor recibido	Salida
{{ITEM.VALUE}.regsub(^[0-9]+, Problem)}	123Log line	Problem
{{ITEM.VALUE}.regsub("^([0-9]+)", "Problem")}	123 Log line	Problem
{{ITEM.VALUE}.regsub(".*", "Problem ID: \1")}	Log line	Problem ID:
{{ITEM.VALUE}.regsub("^(\w+).*?([0-9]+)", " Problem ID: \1_\2 ")}	MySQL crashed errno 123	Problem ID: MySQL_123
{{ITEM.VALUE}.regsub("([1-9]+", "Problem ID: \1")}	123 Log line	*UNKNOWN* (expresión regular no válida)
{{#IFALIAS}.regsub("(.*)_([0-9]+)", \1)}	customername_1	customername
{{#IFALIAS}.regsub("(.*)_([0-9]+)", \2)}	customername_1	1
{{#IFALIAS}.regsub("(.*)_([0-9]+", \1)}	customername_1	{{#IFALIAS}.regsub("(.*)_([0-9]+", \1)} (expresión regular no válida)
{$MACRO:"{{#IFALIAS}.regsub(\"(.*)_([0-9]+\", \1)}"}	customername_1	{$MACRO:"customername"}
{$MACRO:"{{#IFALIAS}.regsub(\"(.*)_([0-9]+\", \2)}"}	customername_1	{$MACRO:"1"}
{$MACRO:"{{#IFALIAS}.regsub(\"(.*)_([0-9]+\", \1)}"}	customername_1	{$MACRO:"{{#IFALIAS}.regsub(\"(.*)_([0-9]+\", \1)}"} (expresión regular no válida)
"{$MACRO:"\{{#IFALIAS}.regsub("(.*)_([0-9]+", \1)}\"}"	customername_1	"{$MACRO:"\customername\"}"
"{$MACRO:"\{{#IFALIAS}.regsub("(.*)_([0-9]+", \2)}\"}"	customername_1	"{$MACRO:"\1\"}"
"{$MACRO:\"{{#IFALIAS}.regsub(\\"(.*)_([0-9]+", \1)}\"}"	customername_1	"{$MACRO:\"{{#IFALIAS}.regsub(\\"(.*)_([0-9]+\\", \1)}\"}" (expresión regular no válida)

Ver valores completos de los items

Los valores largos de las macros resueltas {ITEM.VALUE} y {ITEM.LASTVALUE} para items de texto/log se truncan a 20 caracteres en algunas ubicaciones del frontend. Para ver los valores completos de estas macros puede usar funciones de macro, por ejemplo:

```
{{ITEM.VALUE}.regsub("(.*)", \1)}
{{ITEM.LASTVALUE}.regsub("(.*)", \1)}
```

Consulte también: detalles de las macros {ITEM.VALUE} y {ITEM.LASTVALUE} macro details.

Macros de usuario

Descripción general

Los macros de usuario son compatibles con Zabbix para ofrecer una mayor flexibilidad, además de los macros compatibles de forma predeterminada.

Los macros de usuario se pueden definir a nivel global, de template y de host. Estos macros tienen una sintaxis especial:

{$MACRO}

Zabbix resuelve los macros según la siguiente prioridad:

1. Macros a nivel de host (se comprueban primero).
2. Macros definidos para los templates de primer nivel del host (es decir, templates vinculados directamente al host), ordenados por ID del template.
3. Macros definidos para los templates de segundo nivel del host, ordenados por ID del template.
4. Macros definidos para los templates de tercer nivel del host, ordenados por ID del template, etc.
5. Macros globales (se comprueban en último lugar).

En otras palabras, si un macro no existe para un host, Zabbix intentará encontrarlo en los templates del host, aumentando progresivamente el nivel de profundidad. Si aún no se encuentra, se utilizará un macro global, si existe.

Warning:
Si existe un macro con el **mismo nombre** en varios templates vinculados del mismo nivel, se utilizará el macro del template con el ID más bajo. Por lo tanto, tener macros con el mismo nombre en varios templates supone un riesgo de configuración.

Si Zabbix no puede encontrar un macro, este no se resolverá.

Attention:
Los macros (incluidos los macros de usuario) se dejan sin resolver en la sección Configuration (por ejemplo, en la lista de triggers) de forma intencionada, para que las configuraciones complejas sean más transparentes.

Los macros de usuario se pueden utilizar en:

- nombre del item
- parámetro de la clave del item
- intervalos de actualización y intervalos flexibles del item
- nombre y descripción del trigger
- parámetros y constantes de la expresión del trigger (consulte los ejemplos)
- muchas otras ubicaciones; consulte la lista completa

Casos de uso comunes de macros globales y de equipo

- utilizar una macro global en varios lugares; luego cambiar el valor de la macro y aplicar los cambios de configuración a todos los lugares con un solo clic
- aprovechar las plantillas con atributos específicos del equipo: contraseñas, números de puerto, nombres de archivos, expresiones regulares, etc.

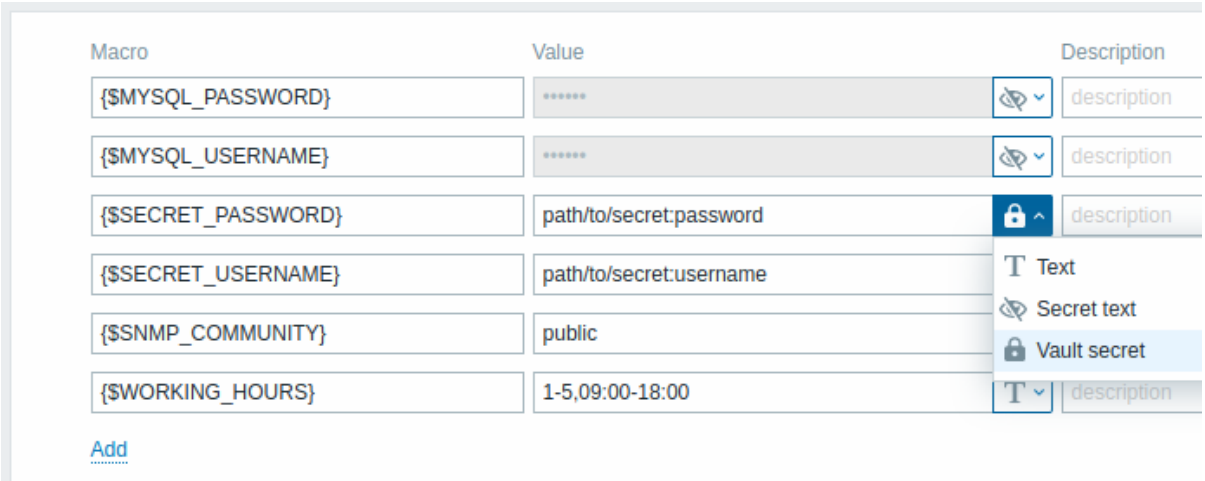
Note:
Se recomienda usar macros de host en lugar de macros globales, porque agregar, actualizar o eliminar macros globales obliga a una actualización incremental de la configuración para todos los hosts. Para obtener más información, consulte Passive and active agent checks.

Configuración

Para definir macros de usuario, vaya a la ubicación correspondiente en el frontend:

- Para macros globales, visite *Administración* > *Macros*.
- Para macros a nivel de host y template, abra las propiedades del host o del template y busque la pestaña *Macros*.

Una macro de usuario tiene los siguientes atributos:



Parameter	Description
Macro	Nombre de la macro. El nombre debe ir entre llaves y comenzar con un signo de dólar. Ejemplo: `{$FRONTEND_URL}`. Los siguientes caracteres están permitidos en los nombres de macro: **A-Z** (solo mayúsculas), **0-9**, _, **.**

Parameter	Description
Value	Valor de la macro. Se admiten tres tipos de valor: **Text** (predeterminado) - valor de texto sin formato **Secret text** - el valor se oculta con asteriscos **Vault secret** - el valor contiene una ruta/consulta a un vault secret. Para cambiar el tipo de valor, haga clic en el botón al final del campo de entrada del valor. La longitud máxima de un valor de macro de usuario es de 2048 caracteres.
Description	Campo de texto utilizado para proporcionar más información sobre esta macro.

Al configurar macros de usuario, **tenga en cuenta** los siguientes comportamientos específicos según el contexto:

- si las macros de usuario se usan en items o triggers de template, considere añadir esas macros también al template (aunque estén definidas globalmente); de ese modo, las macros de tipo *Text* funcionarán como se espera después de exportar el template a XML e importarlo en otro sistema (los valores de las secret macros no se exportan)
- si las macros de usuario se usan en expresiones de trigger, esas macros solo se resolverán si hacen referencia a un parámetro o constante; NO se resolverán si hacen referencia a un host, clave de item, función, operador u otra expresión de trigger (las secret macros no se pueden usar en expresiones de trigger)
- si las macros de usuario se usan en un host que tiene una regla de descubrimiento de bajo nivel con prototipos de host, los hosts descubiertos heredarán todas las macros de usuario definidas en ese host

Ejemplos

Ejemplo 1

Uso de una macro de nivel de host en la clave del item "Estado del daemon SSH":

```
net.tcp.service[ssh,,{$SSH_PORT}]
```

Este item se puede asignar a varios hosts, siempre que el valor de `{$SSH_PORT}` esté definido en esos hosts.

Ejemplo 2

Uso de una macro de nivel de host en el trigger "La carga de CPU es demasiado alta":

```
last(/ca_001/system.cpu.load[,avg1])>{$MAX_CPULOAD}
```

Este trigger se crearía en el template, no se editaría en hosts individuales.

> **Note:**
> Si desea utilizar la cantidad de valores como parámetro de la función (por ejemplo, `max(/host/key,#3)`), incluya el signo de almohadilla en la definición de la macro de esta manera: `SOME_PERIOD => #3`.

Ejemplo 3

Uso de dos macros en el trigger "La carga de la CPU es demasiado alta":

```
min(/ca_001/system.cpu.load[,avg1],{$CPULOAD_PERIOD})>{$MAX_CPULOAD}
```

Tenga en cuenta que una macro puede utilizarse como parámetro de una función de trigger, en este ejemplo, la función `min()`.

Ejemplo 4

Sincronice la condición de indisponibilidad del agent con el intervalo de actualización del item:

- defina la macro `{$INTERVAL}` y utilícela en el intervalo de actualización del item;
- utilice `{$INTERVAL}` como parámetro del trigger de indisponibilidad del agent:

```
nodata(/ca_001/agent.ping,{$INTERVAL})=1
```

Ejemplo 5

Centralice la configuración del horario laboral:

- cree una macro global `{$WORKING_HOURS}` con el valor `1-5,09:00-18:00`;
- utilícela en el campo *Horario laboral* de *Administración > General > Interfaz gráfica de usuario*;
- utilícela en el campo *Cuándo está activo* de *Usuarios > Usuarios*, pestaña *Medios* de un usuario;
- utilícela para configurar un sondeo más frecuente de los item durante el horario laboral:

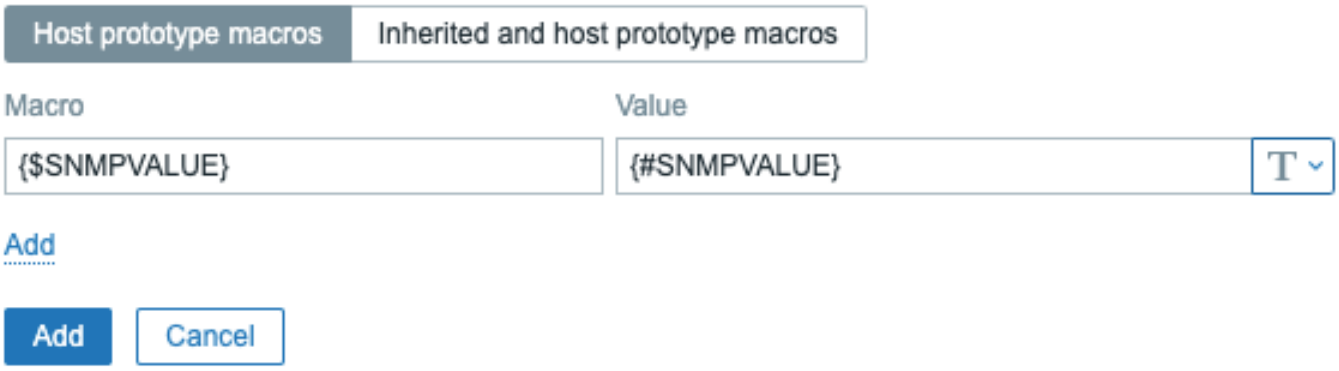


- utilícela en la condición de acción *Período de tiempo*;
- ajuste el horario laboral en *Administración* > *Macros*, si es necesario.

Ejemplo 6

Utilice una macro de prototipo de host para configurar items para los hosts descubiertos:

1. En un prototipo de host, defina la macro de usuario `{$SNMPVALUE}` con la macro de descubrimiento de bajo nivel `{#SNMPVALUE}` como valor:



2. Asigne el template *Generic SNMPv2* al prototipo de host.
3. Utilice `{$SNMPVALUE}` en el campo *SNMP OID* de los items del template *Generic SNMPv2*.

Contexto de las macros de usuario

Consulta macros de usuario con contexto.

Macros de usuario con contexto

Descripción general

Se puede utilizar un contexto opcional en las macros de usuario, lo que permite reemplazar el valor predeterminado por uno específico del contexto.

El contexto se añade al nombre de la macro; la sintaxis depende de si el contexto es un valor de texto estático:

```
{$MACRO:"static text"}
```

o una expresión regular:

```
{$MACRO:regex:"regular expression"}
```

Tenga en cuenta que una macro con contexto de expresión regular solo se puede definir en la configuración de macros de usuario. Si el prefijo `regex:` se utiliza en otro lugar como contexto de macro de usuario, por ejemplo, en una expresión de trigger, se tratará como un contexto estático.

Las comillas del contexto son opcionales (consulte también las notas importantes).

Ejemplos de contextos de macros:

Ejemplo	Descripción
`{$LOW_SPACE_LIMIT}`	Macro de usuario sin contexto.
`{$LOW_SPACE_LIMIT:/tmp}`	Macro de usuario con contexto (cadena estática).
`{$LOW_SPACE_LIMIT:regex:"^/tmp$"}`	Macro de usuario con contexto (expresión regular). Equivale a `{$LOW_SPACE_LIMIT:/tmp}`.
`{$LOW_SPACE_LIMIT:regex:"^/var/log/.*$"}`	Macro de usuario con contexto (expresión regular). Coincide con todas las cadenas que comienzan por /var/log/.

Casos de uso

Las macros de usuario con contexto se pueden definir para lograr umbrales más flexibles en las expresiones de trigger (basados en los valores recuperados por el descubrimiento de bajo nivel). Por ejemplo, puede definir las siguientes macros:

- {$LOW_SPACE_LIMIT} = 10
- {$LOW_SPACE_LIMIT:/home} = 20
- {$LOW_SPACE_LIMIT:regex:"^/[a-z]+$"} = 30

Entonces, una macro de descubrimiento de bajo nivel se puede usar como contexto de macro en un prototipo de trigger para el descubrimiento de sistemas de archivos montados:

```
last(/host/vfs.fs.size[{#FSNAME},pfree])<{$LOW_SPACE_LIMIT:"{#FSNAME}"}
```

Después del descubrimiento, se aplicarán diferentes umbrales de poco espacio en los triggers según los puntos de montaje o los tipos de sistema de archivos descubiertos. Se generarán eventos de problema si:

- la carpeta /home tiene menos del 20% de espacio libre en disco
- las carpetas que coinciden con el patrón regexp (como /etc, /tmp o /var) tienen menos del 30% de espacio libre en disco
- las carpetas que no coinciden con el patrón regexp y no son /home tienen menos del 10% de espacio libre en disco

Notas importantes

- Si existe más de una macro de usuario con contexto, Zabbix intentará hacer coincidir primero las macros de contexto simples y luego las macros de contexto con expresiones regulares en un orden indefinido.

Warning:
No cree diferentes macros de contexto que coincidan con la misma cadena para evitar un comportamiento indefinido.

- Si no se encuentra una macro con su contexto en el equipo, en las plantillas enlazadas o globalmente, entonces se busca la macro sin contexto.
- Solo se admiten macros de bajo nivel de descubrimiento en el contexto. Cualquier otra macro se ignora y se trata como texto plano.

Técnicamente, el contexto de la macro se especifica utilizando reglas similares a los parámetros de la clave de la métrica, excepto que el contexto de la macro no se analiza como varios parámetros si hay un carácter `,`:

- El contexto de la macro debe ir entre comillas " si el contexto contiene un carácter } o comienza con un carácter ". Las comillas dentro del contexto entre comillas deben escaparse con el carácter \.
- El propio carácter \ no se escapa, lo que significa que es imposible tener un contexto entre comillas que termine con el carácter \: la macro {$MACRO:"a:\b\c\"} no es válida.
- Se ignoran los espacios iniciales en el contexto, los espacios finales no:
  - Por ejemplo, {$MACRO:A} es lo mismo que {$MACRO: A}, pero no {$MACRO:A }.
- Todos los espacios antes de las comillas iniciales y después de las comillas finales se ignoran, pero todos los espacios dentro de las comillas no:
  - Las macros {$MACRO:"A"}, {$MACRO: "A"}, {$MACRO:"A" } y {$MACRO: "A" } son iguales, pero las macros {$MACRO:"A"} y {$MACRO:" A "} no lo son.

Las siguientes macros son todas equivalentes, porque tienen el mismo contexto: {$MACRO:A}, {$MACRO: A} y {$MACRO:"A"}. Esto contrasta con las claves de las métricas, donde 'key[a]', 'key[a]' y 'key["a"]' son semánticamente iguales, pero diferentes a efectos de unicidad.

Macros de usuario secretos

Descripción general

Zabbix proporciona dos opciones para proteger información sensible en los valores de las macros de usuario:

- Texto secreto
- Secreto de Vault

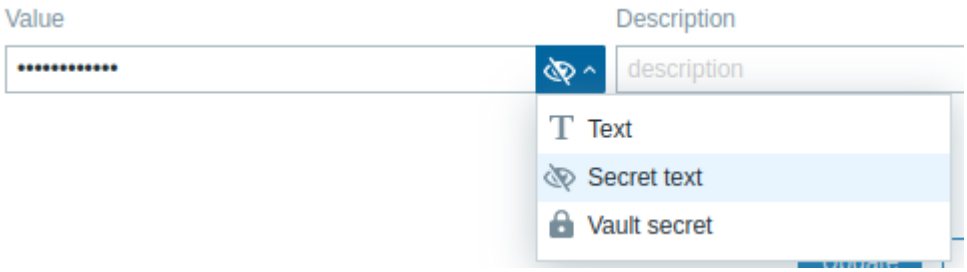
Note:
Aunque el valor de una macro secreta está oculto, puede ser revelado mediante su uso en métricas. Por ejemplo, en un script externo, se puede utilizar una instrucción `echo` que haga referencia a una macro secreta para revelar el valor de la macro en la interfaz, ya que el servidor Zabbix tiene acceso al valor real de la macro. Consulte las ubicaciones donde los valores de las macros secretas se muestran sin máscara.

Las macros secretas no pueden utilizarse en expresiones de disparadores.

Texto secreto

Con las macros de texto secreto, el valor de la macro se oculta con asteriscos.

Para convertir el valor de una macro en secreto, haga clic en el botón al final del campo *Value* y seleccione la opción *Secret text*:



Una vez guardada la configuración, ya no será posible ver el valor.

Para cambiar el valor de la macro, coloque el cursor sobre el campo *Value* y haga clic en el botón *Set new value* (aparece al pasar el cursor):



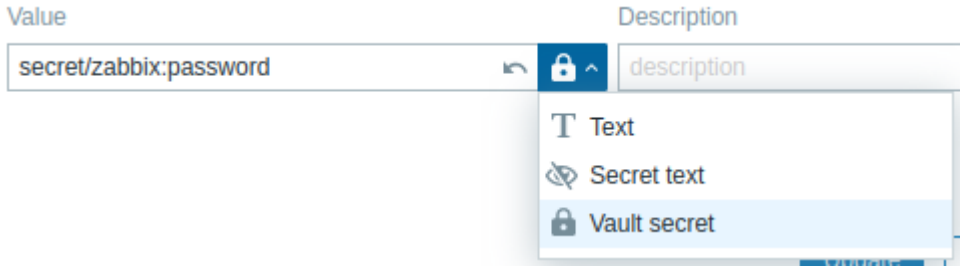
Cuando haga clic en el botón *Set new value* (o cambie el tipo de valor de la macro), el valor actual se borrará. Puede restaurar el valor original haciendo clic en la flecha al final del campo *Value* (solo disponible antes de guardar la nueva configuración). Tenga en cuenta que restaurar el valor original no lo mostrará.

Note:
Las URL que contengan una macro secreta no funcionarán, ya que la macro en ellas se resolverá como "******".

Secreto de Vault

Con las macros de secreto de Vault, el valor de la macro se almacena en un software externo de gestión de secretos (vault).

Para configurar una macro de secreto de Vault, haga clic en el botón al final del campo *Value* y seleccione la opción *Vault secret*:



El valor de la macro debe apuntar a un secreto de vault. El formato de entrada depende del proveedor de vault. Para ver ejemplos de configuración específicos de cada proveedor, consulte:

- HashiCorp
- CyberArk

Los valores de las macros secretas de Vault se recuperan del vault mediante Zabbix server (y Zabbix proxy, si *Resolve secret vault macros by* está configurado en *Zabbix server and proxy*) en cada actualización de los datos de configuración y luego se almacenan en la caché de configuración. Zabbix server y Zabbix proxy pueden usar distintos vaults.

Si *Resolve secret vault macros by* está configurado en *Zabbix server*, entonces los secretos del vault los recupera solo el server y Zabbix proxy recibe los valores de las macros secretas de Vault desde Zabbix server en cada sincronización de configuración y los almacena en su propia caché de configuración. Esto significa que un Zabbix proxy no puede iniciar la recopilación de datos después de un reinicio hasta que reciba la actualización de configuración desde Zabbix server.

Para actualizar manualmente los valores secretos desde el vault, utilice la opción `secrets_reload` de control en tiempo de ejecución (solo server).

El cifrado debe estar habilitado entre Zabbix server y proxy; de lo contrario, se registrará un mensaje de advertencia del server.

Warning:
Si no es posible recuperar correctamente el valor de una macro, el item correspondiente que use ese valor pasará a no ser compatible.

Ubicaciones sin enmascarar

Esta lista proporciona ubicaciones de parámetros donde los valores de macros secretas no están enmascarados.

Note:
Los valores de macros secretas permanecerán enmascarados en las ubicaciones siguientes si se referencian indirectamente. Por ejemplo, {ITEM.KEY}, {ITEM.KEY<1-9>}, {LLDRULE.KEY} macros integradas usadas en tipos de medio (parámetros de Script o webhook) se resolverán a claves de item que contienen macros secretas enmascaradas, como `net.tcp.port[******,******]` en lugar de `net.tcp.port[192.0.2.1,80]`.

Contexto	Parámetro
Items, prototipos de item, reglas LLD	
Item	*Parámetros de la clave del item*
Prototipo de item	*Parámetros de la clave del prototipo de item*
Regla de descubrimiento de bajo nivel	*Parámetros de la clave del item de descubrimiento*
agent SNMP	*Comunidad SNMP*
	Nombre de contexto (SNMPv3)
	Nombre de seguridad (SNMPv3)
	Frase de contraseña de autenticación (SNMPv3)
	Frase de contraseña de privacidad (SNMPv3)
agent HTTP	*URL*
	Campos de consulta
	Cuerpo de la solicitud
	Encabezados
	Nombre de usuario
	Contraseña
	Contraseña de la clave SSL
Script	*Parámetros*
	Script
Browser	*Parámetros*
	Script
Monitor de base de datos	*Consulta SQL*
agent TELNET	*Script*
	Nombre de usuario
	Contraseña
agent SSH	*Script*
	Nombre de usuario
	Contraseña
Comprobación simple	*Nombre de usuario*
	Contraseña
agent JMX	*Nombre de usuario*
	Contraseña
Preprocesamiento del valor del item	
Paso de preprocesamiento de JavaScript	*Script*

Contexto	Parámetro
Escenarios web	
Escenario web	*Valor de variable* *Valor de encabezado* *URL* *Valor del campo de consulta* *Valor del campo POST* *POST sin procesar*
Autenticación del escenario web	*Usuario* *Contraseña* *Contraseña de la clave SSL*
Conectores	
Conector	*URL* *Nombre de usuario* *Contraseña* *Token* *Proxy HTTP* *Archivo de certificado SSL* *Archivo de clave SSL* *Contraseña de la clave SSL*
Descubrimiento de red	
SNMP	*Comunidad SNMP* *Nombre de contexto* (SNMPv3) *Nombre de seguridad* (SNMPv3) *Frase de contraseña de autenticación* (SNMPv3) *Frase de contraseña de privacidad* (SNMPv3)
Scripts globales	
webhook	*Script de JavaScript* *Valor del parámetro del script de JavaScript*
Telnet	*Nombre de usuario* *Contraseña*
SSH	*Nombre de usuario* *Contraseña*
Script	*Script*
Tipos de medio	
Script	*Parámetros del script*
webhook	*Parámetros*
Gestión IPMI	
Host	*Nombre de usuario* *Contraseña*

Macros de descubrimiento de bajo nivel

Descripción general

Hay un tipo de macro que se utiliza dentro de la función de descubrimiento de bajo nivel (LLD):

```
{#MACRO}
```

Es una macro que se utiliza en una regla LLD y devuelve valores reales del nombre del sistema de archivos, interfaz de red, OID SNMP, etc.

Estas macros pueden utilizarse para crear prototipos de entidades (prototipos de item, trigger, gráfico, regla LLD, host y grupo de host). Durante el descubrimiento de sistemas de archivos, interfaces de red, máquinas virtuales, etc., estas macros se sustituyen

por valores reales y son la base para crear entidades reales.

Algunas macros de descubrimiento de bajo nivel vienen "preempaquetadas" con la función LLD en Zabbix - {#FSNAME}, {#FSTYPE}, {#IFNAME}, {#SNMPINDEX}, {#SNMPVALUE}. Sin embargo, no es obligatorio ceñirse a estos nombres al crear una regla de descubrimiento de bajo nivel personalizada. Entonces puede utilizar cualquier otro nombre de macro LLD y hacer referencia a ese nombre.

Tipos de datos soportados

Al definir reglas de descubrimiento personalizadas, los valores de las propiedades devueltos en objetos JSON para macros LLD deben ser uno de los siguientes tipos primitivos:

- string
- number
- boolean

No se admiten arrays, objetos ni valores nulos. Cualquier macro LLD que haga referencia a dicho valor permanecerá sin expandirse y aparecerá literalmente (por ejemplo, `'{#MY_MACRO}'`) durante el preprocesamiento y la creación del item.

Ubicaciones soportadas

Las macros LLD pueden usarse:

- en el filtro de la regla de bajo nivel de descubrimiento
- en reglas de bajo nivel de descubrimiento anidadas, en
  - parámetros de preprocesamiento JSONPath
  - campo JSONPath para macros LLD personalizadas
- para prototipos de item y prototipos de descubrimiento en
  - nombre
  - parámetros de clave
  - unidad
  - intervalo de actualización[1]
  - tiempo de espera[1]
  - período de almacenamiento de histórico[1]
  - período de almacenamiento de tendencias[1]
  - pasos de preprocesamiento del valor del item
  - SNMP OID
  - campo de sensor IPMI
  - expresión de item calculado/agregado, en:
    - constantes de expresión y parámetros de función
    - parámetros de clave de item
  - condiciones de filtro de item agregado (nombre de grupo de host y nombre de etiqueta)
  - script SSH y script Telnet
  - consulta SQL de monitorización de base de datos
  - campo endpoint de item JMX
  - descripción
  - campo URL de HTTP agent
  - campo de parámetros de consulta HTTP de HTTP agent
  - campo de cuerpo de la petición de HTTP agent
  - campo de códigos de estado requeridos de HTTP agent
  - clave y valor del campo de cabeceras de HTTP agent
  - campo de nombre de usuario de autenticación HTTP de HTTP agent
  - campo de contraseña de autenticación HTTP de HTTP agent
  - campo de proxy HTTP de HTTP agent
  - campo de archivo de certificado SSL de HTTP agent
  - campo de archivo de clave SSL de HTTP agent
  - campo de contraseña de clave SSL de HTTP agent
  - etiquetas
- para prototipos de trigger en
  - nombre
  - datos operativos
  - expresión (solo en constantes y parámetros de función)
  - URL
  - descripción
  - etiquetas
- para prototipos de gráficos en
  - nombre

- para prototipos de host en
  - nombre
  - nombre visible
  - campos personalizados de interfaz: IP, DNS, puerto, comunidad SNMP v1/v2, nombre de contexto SNMP v3, nombre de seguridad SNMP v3, frase de autenticación SNMP v3, frase de privacidad SNMP v3
  - nombre de prototipo de grupo de host
  - valor de etiqueta de host
  - valor de macro de host
  - (ver la lista completa)

En todos esos lugares, excepto en el filtro de la regla de bajo nivel de descubrimiento, las macros LLD pueden usarse dentro del contexto de macro de usuario estático.

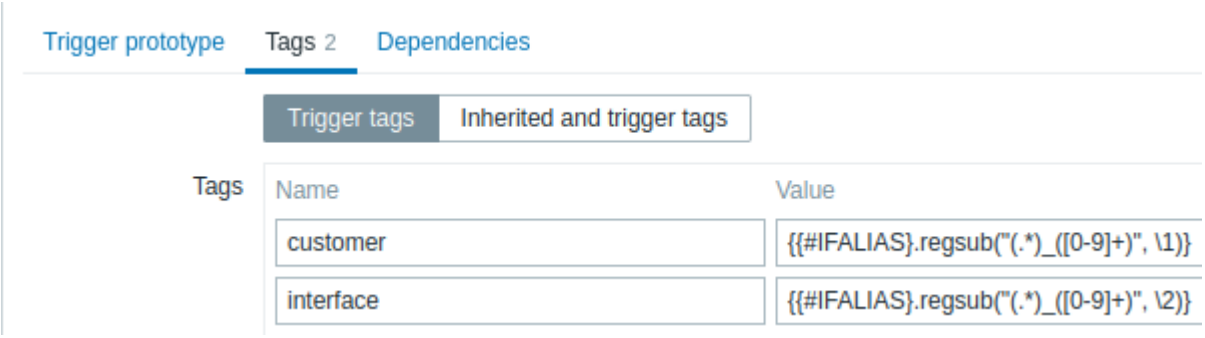
Uso de funciones de macros

Las funciones de macros son compatibles con las macros de descubrimiento de bajo nivel (excepto en el filtro de la regla de descubrimiento de bajo nivel), lo que permite extraer una parte determinada del valor de la macro mediante una expresión regular.

Por ejemplo, puede que desee extraer el nombre del cliente y el número de la interfaz de la siguiente macro de LLD para etiquetar eventos:

```
{#IFALIAS}=customername_1
```

Para ello, puede utilizar la función de macro `regsub` con la macro en el campo de valor de la etiqueta de evento de un prototipo de trigger:



Tenga en cuenta que no se permiten comas en los parámetros de la clave de un item sin comillas, por lo que el parámetro que contiene una función de macro debe estar entre comillas. El carácter de barra inversa (\) debe utilizarse para escapar las comillas dobles dentro del parámetro. Ejemplo:

```
net.if.in["{{#IFALIAS}.regsub(\"(.*)_([0-9]+)\", \1)}",bytes]
```

Para obtener más información sobre la sintaxis de las funciones de macros, consulte: Funciones de macros

Las funciones de macros son compatibles con las macros de descubrimiento de bajo nivel desde Zabbix 4.0.

Prototipos de item sin parámetros de clave

Cuando no se utilizan parámetros de clave de item, coloque la macro LLD dentro de los corchetes del parámetro clave de item `[...]` para que se trate como un parámetro y se sustituya durante el descubrimiento:

```
v_[{#MACRO}]
```

Notas a pie de página

[1] En los campos marcados con [1], una sola macro debe rellenar todo el campo. No se admiten varias macros en un campo ni macros mezcladas con texto.

Macros de expresión

Overview

Las macros de expresión le permiten realizar cálculos en campos.

Su valor se calcula primero resolviendo cualquier macro interna y luego evaluando la expresión resultante.

Sintaxis:

```
{?EXPRESSION}
```

EXPRESSION usa la misma sintaxis y admite las mismas functions que las trigger expressions.

Ejemplo:

```
{?trendavg(/host/item1,1M:now/M)/trendavg(/host/item1,1M:now/M-1y)*100}
```

Para las ubicaciones que admiten macros de expresión, busque "{?EXPRESSION}" en la tabla de supported macro.

Notas de uso:

- Use macros {FUNCTION.*} para referenciar valores de función de trigger expressions/recovery expressions.
- Use macros {HOST.HOST<1-9>} y {ITEM.KEY<1-9>} para referenciar hosts e items.
- En templates, use macros {HOST.HOST<1-9>} o omita por completo el host para el primer host; por ejemplo, {?avg(//item1,1h)}; en lugar de nombres de template, ya que los nombres de template no se reemplazan por nombres de host durante el template linking.

```
{?{FUNCTION.VALUE2} - {FUNCTION.VALUE3}}
{?max(/{HOST.HOST}/{ITEM.KEY},3h)}
```

Véase también Trigger expression examples para un ejemplo de uso de una macro de expresión en un nombre de evento.

Usuarios y grupos de usuarios

Resumen

Todos los usuarios en Zabbix acceden a la aplicación Zabbix a través del frontend basado en web. A cada usuario se le asigna un nombre de inicio de sesión único y una contraseña.

Todas las contraseñas de los usuarios están cifradas y almacenadas en la base de datos de Zabbix. Los usuarios no pueden usar su ID de usuario y contraseña para iniciar sesión directamente en el servidor UNIX, a menos que también se hayan configurado en consecuencia para UNIX. La comunicación entre el servidor web y el navegador del usuario puede protegerse mediante SSL.

Con un esquema flexible de permisos de usuario, puede restringir y diferenciar los derechos para:

- Acceder a las funciones administrativas del frontend de Zabbix.
- Realizar ciertas acciones en el frontend.
- Acceder a los hosts monitorizados en grupos de hosts.
- Usar métodos específicos de la API.

Configuración de un usuario

Descripción general

La instalación inicial de Zabbix tiene dos usuarios predefinidos:

- *Admin*: un superusuario de Zabbix con permisos completos.
- *guest*: un usuario especial de Zabbix.

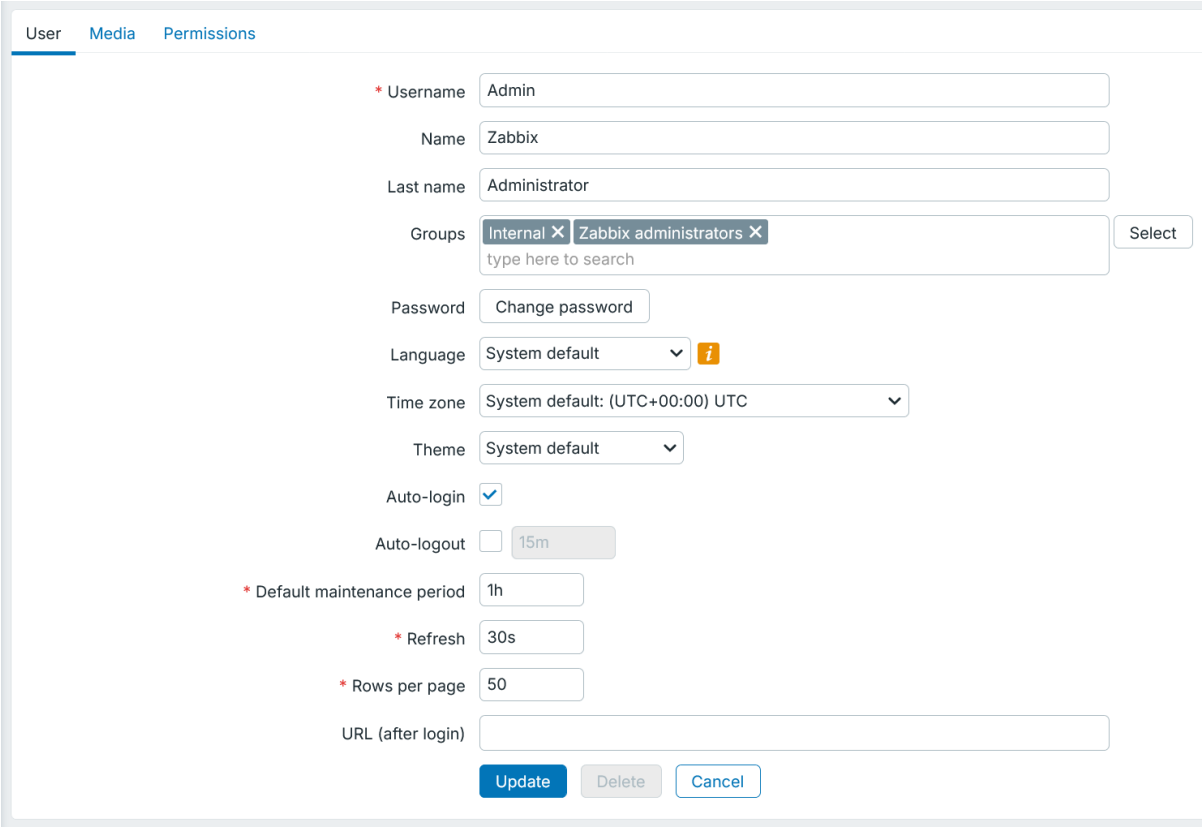
El usuario 'guest' está deshabilitado de forma predeterminada.
Para habilitarlo, elimine el grupo predefinido *Disabled* de los grupos de usuarios.
Puede iniciar sesión con este usuario y acceder a las páginas de monitorización de Zabbix.
Tenga en cuenta que, de forma predeterminada, 'guest' no tiene permisos sobre los objetos de Zabbix.

Para configurar un usuario:

1. Vaya a *Usuarios* > *Usuarios*.
2. Haga clic en *Crear usuario* (o en el nombre de un usuario para editar uno existente).
3. Edite los atributos del usuario en el formulario.

Atributos generales

La pestaña *Usuario* contiene los atributos generales del usuario:



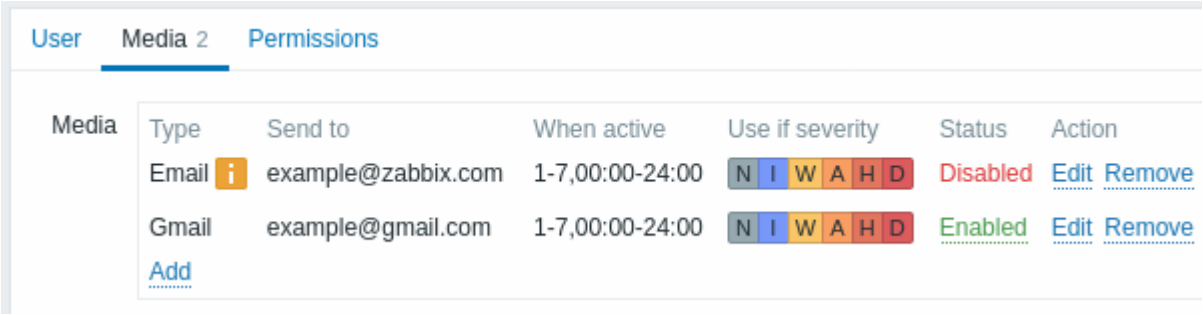
Todos los campos de entrada obligatorios están marcados con un asterisco rojo.

Parámetro	Descripción
Nombre de usuario	Nombre de usuario único, utilizado como nombre de inicio de sesión.
Nombre	Nombre del usuario (opcional). Si no está vacío, se muestra en la información de reconocimiento y en la información del destinatario de la notificación.
Apellidos	Apellidos del usuario (opcional). Si no está vacío, se muestra en la información de reconocimiento y en la información del destinatario de la notificación.
Grupos	Seleccione los grupos de usuarios a los que pertenece el usuario. Este campo se completa automáticamente; al comenzar a escribir el nombre de un grupo de usuarios, se mostrará una lista desplegable con los grupos coincidentes. Desplácese hacia abajo para seleccionar. También puede hacer clic en *Seleccionar* para añadir grupos. Haga clic en «x» para eliminar el elemento seleccionado. La pertenencia a grupos de usuarios determina a qué grupos de hosts y hosts tiene el usuario acceso.
Contraseña	Dos campos para introducir la contraseña del usuario, o un botón *Cambiar contraseña* si el usuario ya existe. Al hacer clic en el botón *Cambiar contraseña*, se abren dos campos para introducir una nueva contraseña. Para un usuario con el *rol de superadministrador* que cambia su propia contraseña, al hacer clic en el botón *Cambiar contraseña* se abre un campo adicional para introducir la contraseña actual (anterior). Cuando la contraseña se cambia correctamente, el usuario cuya contraseña se modificó cerrará sesión en todas las sesiones activas. Tenga en cuenta que la contraseña solo se puede cambiar para usuarios que utilizan la autenticación interna de Zabbix.
Idioma	Idioma del frontend de Zabbix. Se requiere la extensión php gettext para que funcionen las traducciones.
Zona horaria	Seleccione la zona horaria para anular la zona horaria global a nivel de usuario, o seleccione **Predeterminada del sistema** para utilizar la configuración de zona horaria global.

Parámetro	Descripción
Tema	Define el aspecto del frontend: **Predeterminado del sistema**: utiliza la configuración predeterminada del sistema **Azul**: tema azul estándar **Azul (clásico)**: tema azul estándar con fuentes heredadas **Oscuro**: tema oscuro alternativo **Azul oscuro**: tema azul oscuro **Oscuro (clásico)**: tema oscuro alternativo con fuentes heredadas **Claro de alto contraste**: tema claro con alto contraste **Oscuro de alto contraste**: tema oscuro con alto contraste
Inicio de sesión automático	Marque esta casilla para que Zabbix recuerde al usuario e inicie sesión automáticamente durante 30 días. Al iniciar sesión con *Recordarme durante 30 días*: - El cierre de sesión automático se restablece (la sesión persiste durante 30 días). - El inicio de sesión automático se habilita para una reautenticación fluida. Al iniciar sesión sin *Recordarme durante 30 días*: - El inicio de sesión automático se habilita para una reautenticación fluida. - El cierre de sesión automático sigue regido por la configuración de tiempo de espera estándar. Para ello se utilizan cookies del navegador.
Cierre de sesión automático	Con esta casilla marcada, el usuario cerrará sesión automáticamente después del número de segundos establecido (mínimo 90 segundos, máximo 1 día). Tenga en cuenta que esta configuración se anula si *Recordarme durante 30 días* está habilitado, ya que la sesión se prolonga durante todo el período. Se admiten sufijos de tiempo, por ejemplo, 90s, 5m, 2h, 1d. Tenga en cuenta que esta opción no funcionará: * Si la opción de configuración global «Mostrar una advertencia si el server de Zabbix está inactivo» está habilitada y el frontend de Zabbix permanece abierto. * Cuando las páginas del menú Supervisión realizan actualizaciones de información en segundo plano. * Si se inicia sesión con la opción *Recordarme durante 30 días* marcada.
Período de mantenimiento predeterminado	Establezca el período de mantenimiento predeterminado para el mantenimiento ad hoc.
Actualización	Establezca la frecuencia de actualización utilizada para gráficos, datos de texto sin formato, etc. Se puede establecer en 0 para deshabilitarla. Se admiten sufijos de tiempo, por ejemplo, 90s, 5m, 1h.
Filas por página	Puede determinar cuántas filas por página se mostrarán en las listas.
URL (después de iniciar sesión)	Puede hacer que Zabbix transfiera al usuario a una URL específica después de iniciar sesión correctamente, por ejemplo, a la página *Problemas*.

Medios del usuario

La pestaña *Medios* contiene una lista de todos los medios definidos para el usuario.
Los medios se utilizan para enviar notificaciones.



Haga clic en *Añadir* para asignar medios al usuario.

Si el tipo de medio se ha deshabilitado:

- Se muestra un icono de información amarillo después del nombre.
- Se muestra *Deshabilitado* en la columna Estado.

Note:
Los permisos del usuario para cambiar sus propios detalles de medios se pueden conceder o revocar según su rol de usuario (consulte la opción *Crear y editar medios propios*). Los permisos del superadministrador para cambiar los detalles de medios de otros usuarios también se pueden conceder o revocar según su rol de usuario (consulte la opción *Crear y editar medios de usuario*).

Consulte la sección Tipos de medios para obtener información detallada sobre la configuración de los medios del usuario.

Permisos

La pestaña *Permisos* contiene información sobre los siguientes elementos:

- Rol de usuario (obligatorio para cualquier usuario recién creado), que solo puede ser cambiado por un usuario de tipo *Super admin*.

Attention:
No se pueden crear usuarios sin un rol de usuario (excepto mediante la API de usuarios de Zabbix). Los usuarios creados anteriormente que no tengan un rol aún pueden editarse sin asignarles un rol. Sin embargo, una vez asignado un rol, este solo puede cambiarse, no eliminarse.

 Tenga en cuenta que los usuarios sin un rol solo pueden iniciar sesión en Zabbix mediante la autenticación LDAP o SAML, siempre que su información de LDAP/SAML coincida con las asignaciones de grupos de usuarios configuradas en Zabbix.

- Tipo de usuario (*User*, *Admin*, *Super admin*) definido en la configuración del rol de usuario.
- Grupos de hosts y templates a los que el usuario tiene acceso.
  - De forma predeterminada, los usuarios de tipo *User* y *Admin* no tienen acceso a ningún grupo, template ni host. Para concederles dicho acceso, los usuarios deben incluirse en grupos de usuarios configurados con permisos para las entidades correspondientes.
- Derechos de acceso a las secciones y elementos del frontend de Zabbix, los módulos y los métodos de la API.
  - Los elementos cuyo acceso está permitido se muestran en verde, mientras que aquellos cuyo acceso está denegado se muestran en gris claro.
  - Si el acceso a la API está habilitado, los métodos de la API del rol de usuario se muestran en *Allowed methods* o *Denied methods*, según si se selecciona *Allow list* o *Deny list* en la configuración del rol de usuario. Si no se especifican métodos de la API, se muestra *None*. *None* en *Allowed methods* significa que no se permite ningún método de la API, mientras que *None* en *Denied methods* significa que se permiten todos los métodos de la API disponibles para el tipo de usuario.
- Derechos para realizar acciones específicas.
  - Las acciones que el usuario puede realizar se muestran en verde, mientras que aquellas que no puede realizar se muestran en gris claro.

Consulte la página Permisos para obtener más información.

Permisos

Descripción general

Los permisos en Zabbix dependen del tipo de usuario, los roles de usuario personalizados y el acceso a los equipos, que se especifica en función del grupo de usuarios.

Tipos de usuario

Los permisos en Zabbix dependen principalmente del tipo de usuario:

- *User*: tiene derechos de acceso limitados a las secciones del menú (consulte más abajo) y, de forma predeterminada, no tiene acceso a ningún recurso. Los permisos para los grupos de hosts o templates deben asignarse explícitamente.
- *Admin*: tiene derechos de acceso incompletos a las secciones del menú (consulte más abajo). De forma predeterminada, el usuario no tiene acceso a ningún grupo de hosts. Los permisos para los grupos de hosts o templates deben otorgarse explícitamente.
- *Super admin*: tiene acceso a todas las secciones del menú. El usuario tiene acceso de lectura y escritura a todos los grupos de hosts y templates. Los permisos no pueden revocarse denegando el acceso a grupos específicos.

Acceso al menú

Note:
El acceso restringido a algunos elementos de la interfaz de usuario solo impide abrir esa página; no elimina la posibilidad de acceder a los datos subyacentes desde otras partes de la interfaz.

La siguiente tabla muestra el acceso a las secciones del menú de Zabbix según el tipo de usuario:

Sección del menú		User	Admin	Super admin
Dashboards		+	+	+
Monitoring		+	+	+
	Problems	+	+	+
	Hosts	+	+	+
	Latest data	+	+	+
	Maps	+	+	+
	Discovery		+	+
APM		+	+	+
	Traces	+	+	+
	Metrics	+	+	+
	Logs	+	+	+
Services		+	+	+
	Services	+	+	+
	SLA		+	+
	SLA report	+	+	+
Inventory		+	+	+
	Overview	+	+	+
	Hosts	+	+	+
Reports		+	+	+
	System information		+	+
	Scheduled reports		+	+
	Availability report	+	+	+
	Top 100 triggers	+	+	+
	Audit log			+
	Action log			+
	Notifications		+	+
Data collection			+	+
	Template groups		+	+
	Host groups		+	+
	Templates		+	+
	Hosts		+	+
	Maintenance		+	+
	Event correlation			+
	Discovery		+	+
Alerts			+	+
	Trigger actions		+	+
	Service actions		+	+
	Discovery actions		+	+
	Autoregistration actions		+	+
	Internal actions		+	+
	Media types			+
	Scripts			+
Users				+
	User groups			+
	User roles			+
	Users			+
	API tokens			+
	Authentication			+
	Devices			+
Administration				+
	Data source			+
	General			+
	Audit log			+
	Housekeeping			+
	Proxy groups			+
	Proxies			+
	Macros			+
	Queue			+

Roles de usuario

Los roles de usuario permiten realizar ajustes personalizados en los permisos definidos por el tipo de usuario.
Aunque no se pueden añadir permisos (ya que eso superaría los permisos del tipo de usuario), algunos permisos se pueden revocar.

Además, un rol de usuario determina el acceso no solo a las secciones del menú, sino también a los servicios, módulos, métodos de la API y diversas acciones en el frontend.

Los roles de usuario se configuran en la sección *Usuarios* > *Roles de usuario* por usuarios Superadministrador.

Los roles de usuario se asignan a los usuarios en el formulario de configuración del usuario, en la pestaña *Permisos*, por usuarios Superadministrador.

Users

User | Media | Permissions

* Role: Admin role ✕ | Select

User type: Admin

Permissions

User group permissions

Group	Type	Permissions
All groups	Hosts	None
All groups	Templates	None

Proxies: None

Proxy groups: None

Permissions can be assigned for user groups only.

Access to UI elements

Dashboards: Dashboards

Monitoring: Problems | Hosts | Latest data | Maps | Discovery

APM: Traces | Metrics | Logs

Services: Services | SLA | SLA report

Inventory: Overview | Hosts

Reports: System information | Scheduled reports | Availability report | Top 100 triggers | Notifications

Data collection: Template groups | Host groups | Templates | Hosts | Maintenance | Discovery

Alerts: Trigger actions | Service actions | Discovery actions | Autoregistration actions | Internal actions

User profile settings

Redirect URL after login

Access to services

Read-write access to services: None

Read-only access to services: All

Access to modules

Action log | Clock | Discovery status | Favorite graphs | Favorite maps | Gauge | Geomap | Graph | Graph (classic) | Graph prototype | Honeycomb | Host availability | Host card | Host navigator | Item card | Item history | Item navigator | Item value | Map | Map navigation tree | Pie chart | Problem hosts | Problems | Problems by severity | Scatter plot | SLA report | System information | Top hosts | Top items | Top triggers | Trigger overview | URL | Web monitoring

Access to API

Disabled

Devices

Enabled | Manage own devices

Access to actions

Create and edit dashboards | Create and edit maps | Create and edit maintenance | Add problem comments | Change severity | Acknowledge problems | Suppress problems | Close problems | Execute scripts | Manage API tokens | Manage scheduled reports | Manage SLA | Invoke "Execute now" on read-only hosts | Change problem ranking | Create and edit own media | Select "Server" for monitoring and discovery

Add | Cancel

Acceso a hosts

El acceso a los datos de cualquier host y template en Zabbix se concede a los grupos de usuarios únicamente en el nivel de grupo de host/template.

Esto significa que a un usuario individual no se le puede conceder acceso directamente a un host (o grupo de hosts). Solo se le puede conceder acceso a un host formando parte de un grupo de usuarios al que se le haya concedido acceso al grupo de hosts que contiene el host.

De forma similar, a un usuario solo se le puede conceder acceso a un template formando parte de un grupo de usuarios al que se le haya concedido acceso al grupo de templates que contiene el template.

Grupos de usuarios

Descripción general

Los grupos de usuarios permiten agrupar usuarios tanto con fines organizativos como para asignar permisos sobre los datos. Los permisos para ver y configurar los datos de los grupos de hosts y de los grupos de templates se asignan a los grupos de usuarios, no a usuarios individuales.

A menudo puede ser conveniente separar qué información está disponible para un grupo de usuarios y cuál para otro. Esto se puede lograr agrupando a los usuarios y asignando después distintos permisos a los grupos de hosts y de templates.

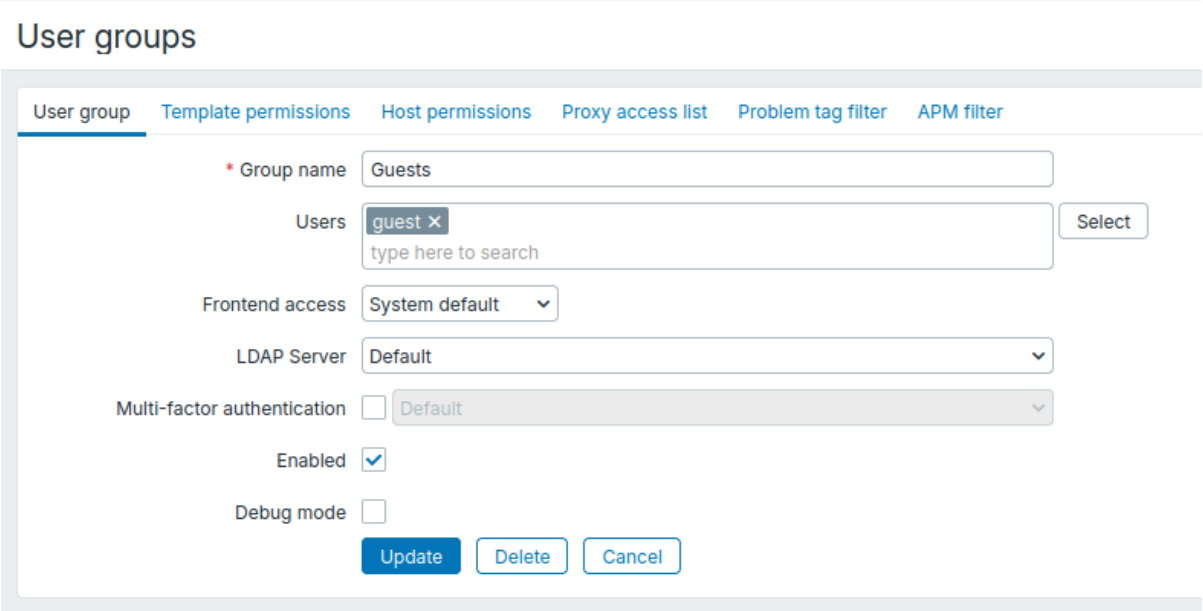
Un usuario puede pertenecer a cualquier número de grupos.

Configuración

Para configurar un grupo de usuarios:

1. Vaya a *Usuarios* > *Grupos de usuarios*.
2. Haga clic en *Crear grupo de usuarios* (o en el nombre del grupo para editar un grupo existente).
3. Edite los atributos del grupo en el formulario.

La pestaña **Grupo de usuarios** contiene los atributos generales del grupo:

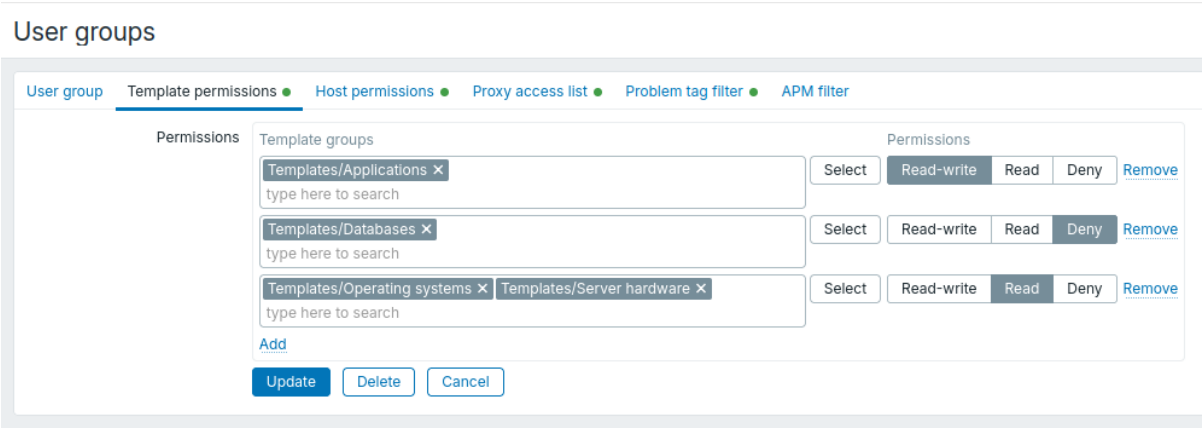


Todos los campos de entrada obligatorios están marcados con un asterisco rojo.

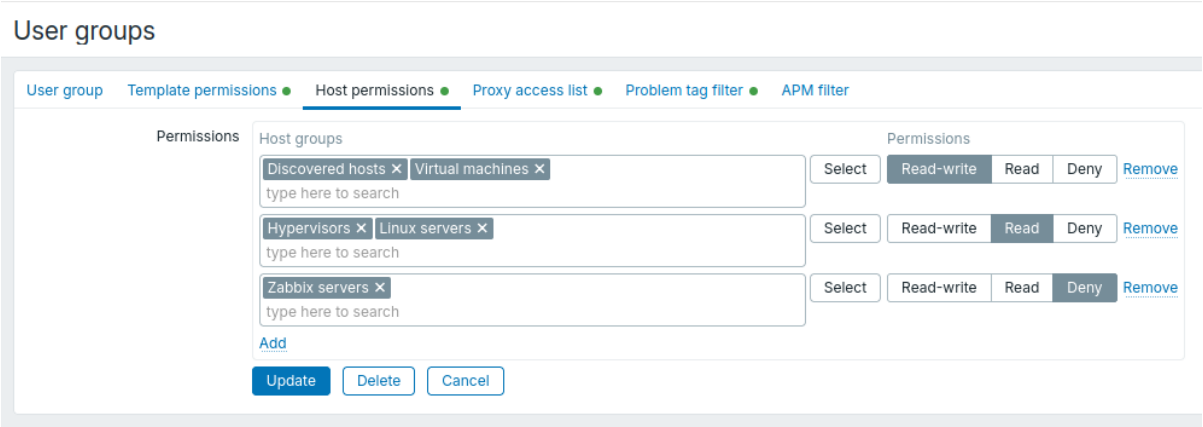
Parámetro	Descripción
Nombre del grupo	Nombre único del grupo.
Usuarios	Para añadir usuarios al grupo, comience a escribir el nombre de un usuario existente. Cuando aparezca el menú desplegable con los nombres de usuario coincidentes, desplácese hacia abajo para seleccionar uno. También puede hacer clic en el botón *Seleccionar* para seleccionar usuarios en una ventana emergente.

Parámetro	Descripción
Acceso al frontend	Cómo se autentican los usuarios del grupo. **Predeterminado del sistema**: utiliza el método de autenticación predeterminado (establecido globalmente) **Interno**: utiliza la autenticación interna de Zabbix (incluso si la autenticación LDAP se utiliza globalmente). Se ignora si la autenticación HTTP es la predeterminada global. **LDAP**: utiliza la autenticación LDAP (incluso si la autenticación interna se utiliza globalmente). Se ignora si la autenticación HTTP es la predeterminada global. **Deshabilitado**: se prohíbe el acceso al frontend de Zabbix para este grupo
Servidor LDAP	Seleccione qué servidor LDAP utilizar para autenticar al usuario. Este campo está habilitado solo si *Acceso al frontend* está establecido en LDAP o Predeterminado del sistema.
Autenticación multifactor	Seleccione qué método de autenticación multifactor utilizar para autenticar al usuario: **Predeterminado**: utiliza el método establecido como predeterminado en la configuración de MFA; esta opción está seleccionada de forma predeterminada para los nuevos grupos de usuarios si MFA está habilitada; **<Nombre del método>**: utiliza el método seleccionado (por ejemplo, "Zabbix TOTP"); **Deshabilitado**: MFA está deshabilitada para este grupo; esta opción está seleccionada de forma predeterminada para los nuevos grupos de usuarios si MFA está deshabilitada. Tenga en cuenta que, si un usuario pertenece a varios grupos de usuarios con MFA habilitada (o al menos un grupo tiene MFA habilitada), se aplican las siguientes reglas de autenticación: si algún grupo utiliza el método de MFA "Predeterminado", autenticará al usuario; de lo contrario, se utilizará para la autenticación el primer método (ordenado alfabéticamente).
Habilitado	Estado del grupo de usuarios y de sus miembros. *Marcado*: el grupo de usuarios y los usuarios están habilitados *Sin marcar*: el grupo de usuarios y los usuarios están deshabilitados
Modo de depuración	Marque esta casilla para activar el modo de depuración para los usuarios.

La pestaña **Permisos de template** permite especificar el acceso del grupo de usuarios a los grupos de templates y, a través de ellos, a los datos de los templates:



La pestaña **Permisos de host** permite especificar el acceso del grupo de usuarios a los grupos de hosts y, a través de ellos, a los datos de los hosts:



Haga clic en Add para elegir los grupos de templates/hosts (ya sean grupos principales o anidados) y asignarles permisos. Comience a escribir el nombre del grupo (aparecerá un menú desplegable con los grupos coincidentes) o haga clic en *Seleccionar* para abrir una ventana emergente con todos los grupos.

A continuación, utilice los botones de opción para asignar permisos a los grupos seleccionados. Los permisos posibles son los siguientes:

- **Lectura y escritura**: acceso de lectura y escritura a un grupo
- **Lectura**: acceso de solo lectura a un grupo
- **Denegar**: acceso denegado a un grupo

Si el mismo grupo de templates/hosts se añade en varias filas con diferentes permisos, se aplicará el permiso más restrictivo.

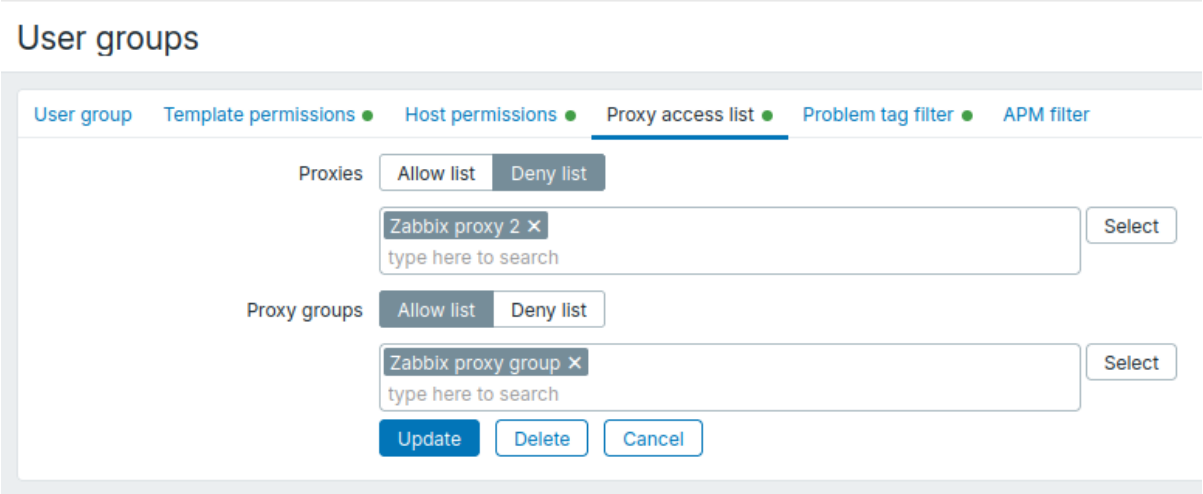
Tenga en cuenta que un usuario *Superadministrador* puede imponer que los grupos anidados tengan el mismo nivel de permisos que el grupo principal; esto se puede hacer en el formulario de configuración del grupo de hosts/templates.

Las pestañas **Permisos de template** y **Permisos de host** admiten el mismo conjunto de parámetros.

Los permisos actuales para los grupos se muestran en el bloque *Permisos* y se pueden modificar o eliminar.

Note:
Si un grupo de usuarios tiene permisos de **Lectura y escritura** para un host y permisos de **Denegar** o ningún permiso para un template vinculado a este host, los usuarios de dicho grupo no podrán editar los items basados en templates en el host, y el nombre del template se mostrará como *Template inaccesible*.

La pestaña **Lista de acceso a proxies** permite especificar el acceso del grupo de usuarios a los proxies, individualmente o mediante grupos de proxies:



Utilice el campo *Grupos de proxies* para aplicar el permiso a los grupos seleccionados. Un proxy que pertenece a un grupo solo se puede permitir o denegar a través de ese grupo. Utilice el campo *Proxies* para añadir proxies individuales que no pertenecen a ningún grupo de proxies.

Proxies y *Grupos de proxies* admiten el mismo conjunto de parámetros.

Haga clic en el botón *Seleccionar* junto al campo *Proxies* o *Grupos de proxies* para seleccionar proxies o grupos de proxies, respectivamente. También puede comenzar a escribir el nombre de un proxy o grupo de proxies en el campo correspondiente. Aparecerá una lista desplegable con los proxies o grupos de proxies coincidentes.

Seleccione *Lista de permitidos* para permitir que el grupo de usuarios acceda únicamente a los proxies y grupos de proxies seleccionados.
 Si la selección de *Proxies* está vacía para la *Lista de permitidos*, el grupo de usuarios no tendrá acceso a ningún proxy que no pertenezca a un grupo de proxies. Si la selección de *Grupos de proxies* está vacía para la *Lista de permitidos*, el grupo de usuarios no tendrá acceso a ningún grupo de proxies.

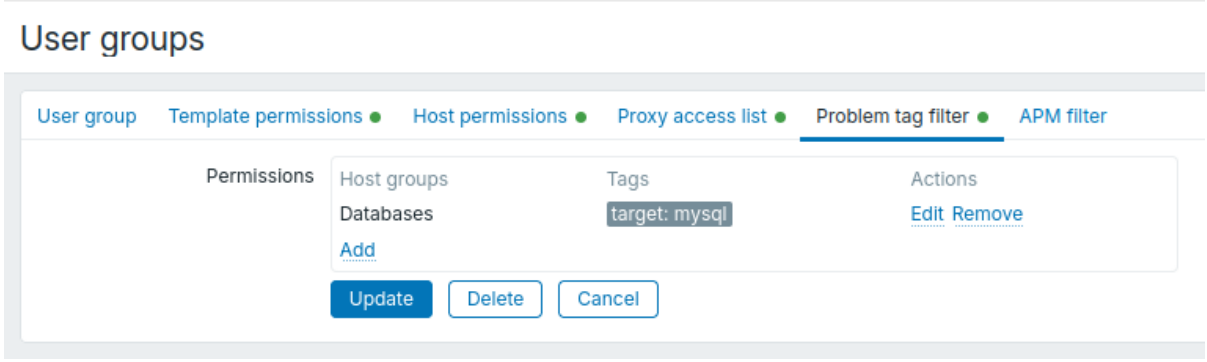
Seleccione *Lista de denegados* para denegar al grupo de usuarios el acceso a los proxies y grupos de proxies seleccionados. Se permite el acceso a todos los demás proxies y grupos de proxies.
 Si la selección de *Proxies* está vacía para la *Lista de denegados*, el grupo de usuarios tendrá acceso completo a todos los proxies que no pertenezcan a un grupo de proxies. Si la selección de *Grupos de proxies* está vacía para la *Lista de denegados*, el grupo de usuarios tendrá acceso completo a todos los grupos de proxies.

Reglas de precedencia de permisos para usuarios que pertenecen a varios grupos de usuarios:

- La *Lista de denegados* prevalece sobre la *Lista de permitidos*. Si algún grupo deniega el acceso a un proxy/grupo de proxies, el usuario no podrá acceder a él.

- Los permisos de la *Lista de permitidos* son acumulativos. Si varios grupos permiten el acceso, el usuario podrá acceder a todos los proxies permitidos por esos grupos.

La pestaña **Filtro de etiquetas de problemas** permite establecer permisos basados en etiquetas para que los grupos de usuarios vean problemas filtrados por nombre y valor de etiqueta:

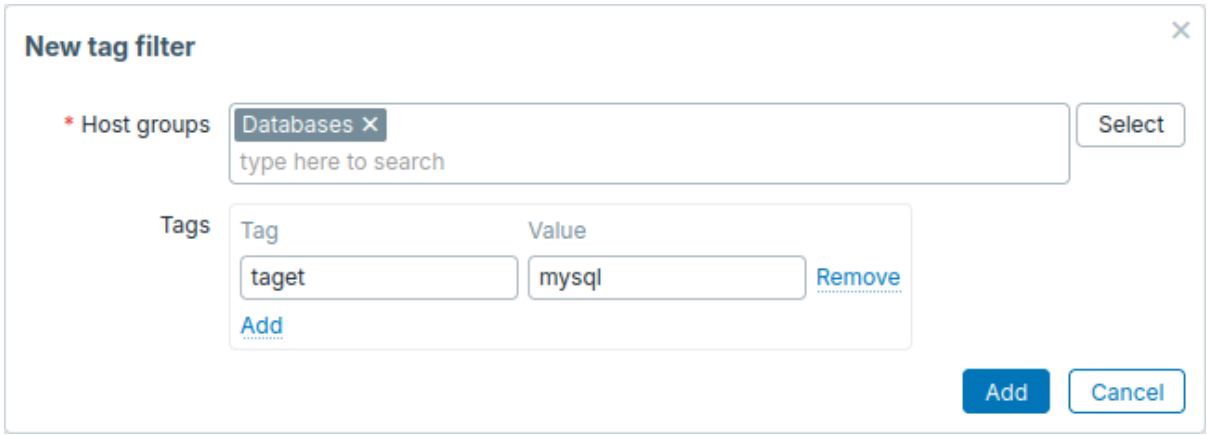


Haga clic en Add para elegir los grupos de hosts. Para seleccionar un grupo de hosts al que aplicar un filtro de etiquetas, haga clic en *Seleccionar* para obtener la lista completa de grupos de hosts existentes o comience a escribir el nombre de un grupo de hosts para obtener un menú desplegable con los grupos coincidentes. Solo se mostrarán grupos de hosts, porque el filtro de etiquetas de problemas no se puede aplicar a grupos de templates.

Se pueden añadir nombres de etiquetas sin valores, pero no valores sin nombres. Solo las tres primeras etiquetas (con valores, si los tienen) se muestran en el bloque *Permisos*; si hay más, se pueden ver haciendo clic en el icono ••• o pasando el cursor sobre él.

El filtro de etiquetas permite separar el acceso al grupo de hosts de la posibilidad de ver problemas.

Por ejemplo, si un administrador de bases de datos necesita ver solo los problemas de bases de datos "MySQL", primero es necesario crear un grupo de usuarios para administradores de bases de datos y, a continuación, especificar el nombre de etiqueta "target" y el valor "mysql".



Si se especifica el nombre de etiqueta "target" y se deja vacío el campo de valor, el grupo de usuarios verá todos los problemas con el nombre de etiqueta "target" para el grupo de hosts seleccionado.

Asegúrese de especificar correctamente el nombre y el valor de la etiqueta; de lo contrario, el grupo de usuarios no verá ningún problema.

Revisemos un ejemplo en el que un usuario es miembro de varios grupos de usuarios seleccionados. En este caso, el filtrado utilizará la condición OR para las etiquetas.

Grupo de usuarios A			**Grupo de usuarios B**			**Resultado visible para un usuario (miembro) de ambos grupos**
Filtro de etiquetas						
Grupo de hosts	*Nombre de etiqueta*	*Valor de etiqueta*	*Grupo de hosts*	*Nombre de etiqueta*	*Valor de etiqueta*	

Bases de datos	target	mysql	Bases de datos	target	oracle	problemas target:mysql o target:oracle visibles
No configurado en **Filtro de etiquetas de problemas**			Bases de datos	target	oracle	problemas target:oracle visibles

Acceso desde varios grupos de usuarios

Un usuario puede pertenecer a cualquier número de grupos de usuarios. Estos grupos pueden tener diferentes permisos de acceso a hosts o templates.

Por lo tanto, es importante saber a qué entidades podrá acceder un usuario sin privilegios como resultado. En el siguiente ejemplo, considere cómo se verá afectado el acceso al host **X** (en el Grupo de hosts 1) en varias situaciones para un usuario que está en los grupos de usuarios A y B.

- Si el Grupo A tiene solo acceso de *Lectura* al Grupo de hosts 1, pero el Grupo B tiene acceso de *Lectura-escritura* al Grupo de hosts 1, el usuario obtendrá acceso de **Lectura-escritura** a 'X'.

Attention:
Los permisos de "Lectura-escritura" tienen prioridad sobre los permisos de "Lectura".

- En el mismo escenario anterior, si 'X' está simultáneamente también en el Grupo de hosts 2 que está **denegado** al Grupo A o B, el acceso a 'X' será **no disponible**, a pesar de un acceso de *Lectura-escritura* al Grupo de hosts 1.
- Si el Grupo A no tiene permisos definidos y el Grupo B tiene acceso de *Lectura-escritura* al Grupo de hosts 1, el usuario obtendrá acceso de **Lectura-escritura** a 'X'.
- Si el Grupo A tiene acceso de *Denegado* al Grupo de hosts 1 y el Grupo B tiene acceso de *Lectura-escritura* al Grupo de hosts 1, el usuario tendrá el acceso a 'X' **denegado**.

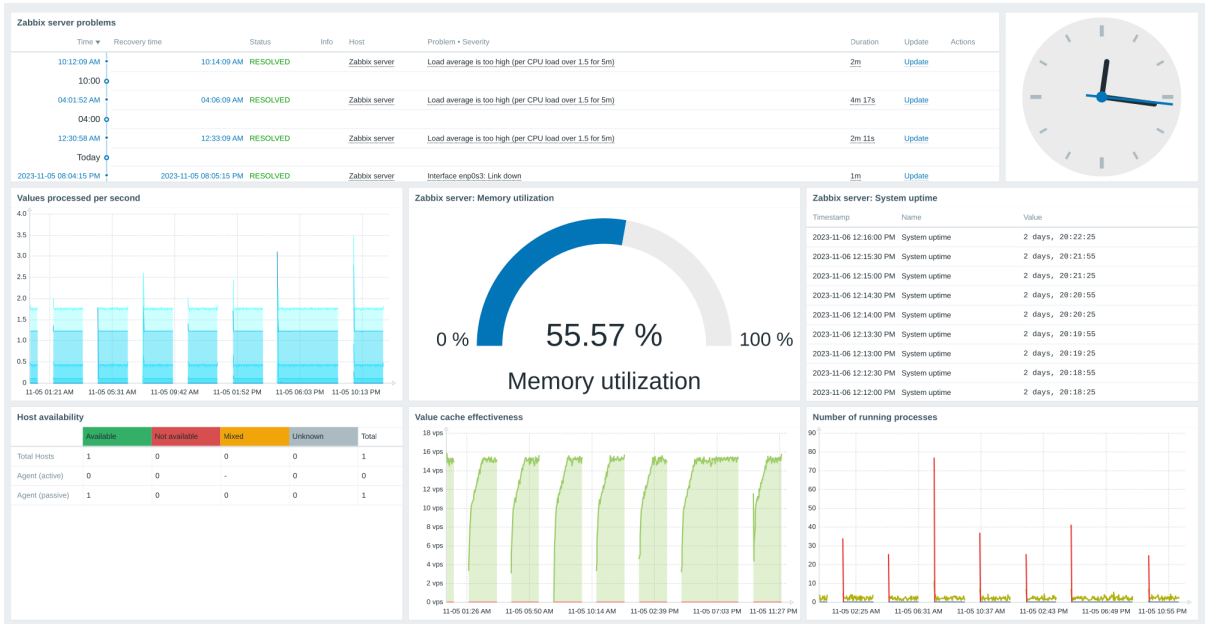
Otros detalles

- Un usuario de nivel Admin con acceso *Lectura-escritura* a un host no podrá vincular/desvincular templates si no tiene acceso al grupo de templates al que pertenecen. Con acceso *Lectura* al grupo de templates podrá vincular/desvincular templates al host, sin embargo, no verá ningún template en la lista de templates y no podrá operar con templates en otros lugares.
- Un usuario de nivel Admin con acceso *Lectura* a un host no verá el host en la lista de hosts de la sección de configuración; sin embargo, los triggers del host serán accesibles en la configuración del servicio IT.
- Cualquier usuario que no sea Super Admin (incluido 'guest') puede ver mapas de red siempre que el mapa esté vacío o solo tenga imágenes. Cuando se agregan hosts, grupos de hosts o triggers al mapa, se respetan los permisos.
- El servidor Zabbix no enviará notificaciones a los usuarios definidos como destinatarios de la operación de acción si el acceso al host en cuestión está explícitamente "denegado".

Informes programados

Descripción general

Con la función *Informes programados*, puede configurar el envío de una versión en PDF de un dashboard determinado a los destinatarios especificados en intervalos recurrentes.



Requisitos previos:

- El Zabbix web service debe estar instalado y configurado correctamente para habilitar la generación de informes programados; consulte Configuración de informes programados para obtener instrucciones.
- Un usuario debe tener un rol de usuario de tipo *Admin* o *Super admin* con los siguientes permisos:
  - *Informes programados* en el bloque *Acceso a los elementos de la interfaz de usuario* (para ver la configuración de los informes)
  - *Gestionar informes programados* en el bloque *Acceso a las acciones* (para crear o editar informes)

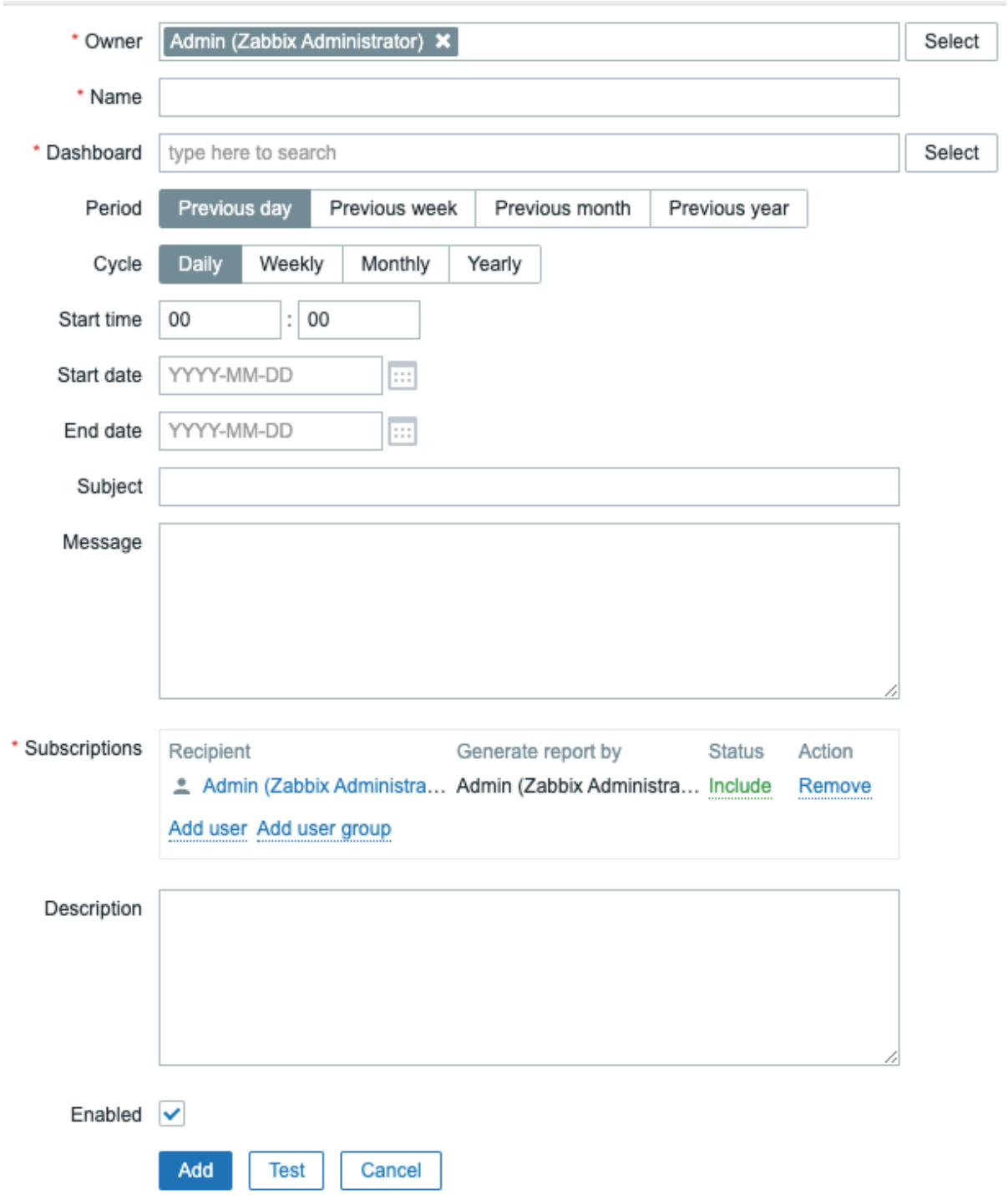
Para crear un informe programado en el frontend de Zabbix, haga lo siguiente:

1. Vaya a *Informes* > *Informes programados*.
2. Haga clic en *Crear informe* en la esquina superior derecha de la pantalla.
3. Introduzca los parámetros del informe en el formulario.

Como alternativa, puede crear un informe abriendo uno existente, haciendo clic en el botón *Clonar* y guardándolo con otro nombre.

Configuración

La pestaña *Scheduled reports* contiene los atributos generales del informe.



Todos los campos de entrada obligatorios están marcados con un asterisco rojo.

Parámetro	Descripción
Owner	Usuario que crea un informe. Los usuarios con nivel *Super admin* pueden cambiar el propietario. Para los usuarios con nivel *Admin*, este campo es de solo lectura.
Name	Nombre del informe; debe ser único.
Dashboard	Dashboard en el que se basa el informe; solo se puede seleccionar un dashboard a la vez. Para seleccionar un dashboard, comience a escribir el nombre; aparecerá una lista de dashboards coincidentes. Desplácese hacia abajo para seleccionar uno. También puede hacer clic en *Select*, junto al campo, y seleccionar un dashboard de la lista mostrada.
Period	Período de tiempo para el que se preparará el informe. Seleccione el día, la semana, el mes o el año anterior.
Cycle	Frecuencia de generación del informe. Los informes se pueden enviar diariamente, semanalmente, mensualmente o anualmente. El modo "Weekly" permite seleccionar los días de la semana en los que se enviará el informe.
Start time	Hora del día, en el formato hh:mm, a la que se preparará el informe. Tenga en cuenta que se utilizará la zona horaria del Zabbix server.

Parámetro	Descripción
Repeat on	Días de la semana en los que se enviará el informe. Este campo solo está disponible si *Cycle* está establecido en ”Weekly”.
Start date	Fecha en la que debe comenzar la generación periódica de informes.
End date	Fecha en la que debe detenerse la generación periódica de informes.
Subject	Asunto del correo electrónico del informe. Macros compatibles: {TIME}, {TIMESTAMP}.
Message	Cuerpo del correo electrónico del informe. Macros compatibles: {TIME}, {TIMESTAMP}.
Subscriptions	Lista de destinatarios del informe. De forma predeterminada, incluye solo al propietario del informe. Se puede especificar como destinatario del informe cualquier usuario de Zabbix que tenga configurado un medio de correo electrónico. Haga clic en *Add user* o *Add user group* para añadir más destinatarios. Haga clic en el nombre de usuario para editar la configuración: *Generate report by*: indica si los datos del informe deben generarse según los permisos del dashboard del usuario actual o del destinatario. *Status*: seleccione ”Include” para enviar el informe al usuario o ”Exclude” para impedir que se envíe a este usuario. Al menos un usuario debe tener el estado ”Include”. El estado ”Exclude” se puede utilizar para excluir usuarios específicos de un grupo de usuarios incluido. Tenga en cuenta que los usuarios con permisos insuficientes (es decir, usuarios con un rol basado en el tipo de usuario *Admin* que no son miembros del mismo grupo de usuarios que el destinatario o el propietario del informe) verán ”Inaccessible user” o ”Inaccessible user group” en lugar de los nombres reales en los campos *Recipient* y *Generate report by*; los campos *Status* y *Action* se mostrarán como de solo lectura.
Enabled	Estado del informe. Al desmarcar esta casilla, se deshabilitará el informe.
Description	Descripción opcional del informe. Esta descripción es para uso interno y no se enviará a los destinatarios del informe.

Botones del formulario

Los botones situados en la parte inferior del formulario permiten realizar varias operaciones.

Add	Añadir un informe. Este botón solo está disponible para informes nuevos.
Update	Actualizar las propiedades de un informe.
Clone	Crear otro informe basándose en las propiedades del informe actual.
Test	Comprobar si la configuración del informe es correcta enviando un informe al usuario actual.
Delete	Eliminar el informe.
Cancel	Cancelar la edición de las propiedades del informe.

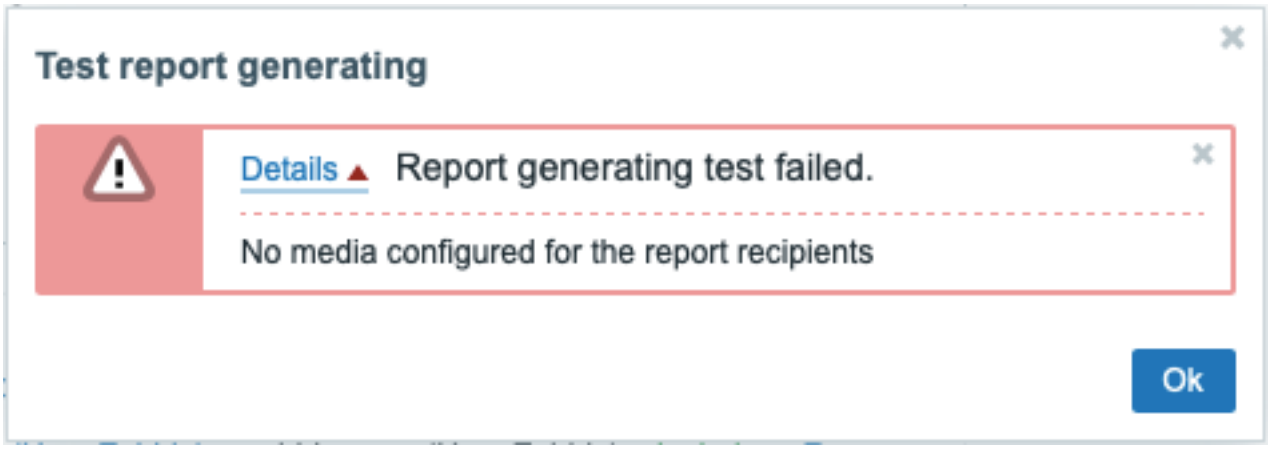
Pruebas

Para probar un informe, haga clic en el botón *Probar* situado en la parte inferior del formulario de configuración del informe.

> **Note:**
> El botón *Probar* no está disponible si el formulario de configuración del informe se ha abierto desde el menú de acciones del dashboard.

Si la configuración es correcta, el informe de prueba se envía inmediatamente al usuario actual. En los informes de prueba, se ignoran los suscriptores y la configuración de usuario *Generar informe por*.

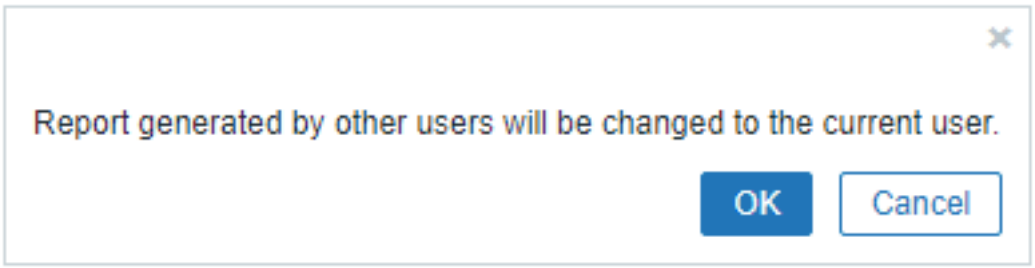
Si la configuración es incorrecta, se muestra un mensaje de error que describe la posible causa.



Actualizar un informe

Para actualizar un informe existente, haga clic en el nombre del informe, realice los cambios de configuración necesarios y luego haga clic en el botón *Actualizar*.

Si otro usuario actualiza un informe existente y este usuario cambia el Panel, al hacer clic en el botón *Actualizar*, se mostrará un mensaje de advertencia "El informe generado por otros usuarios se cambiará al usuario actual".



Hacer clic en *OK* en este paso provocará los siguientes cambios:

- La configuración *Generar informe por* se actualizará para mostrar el usuario que editó el informe por última vez (a menos que *Generar informe por* esté configurado como destinatario).
- Los usuarios que se hayan mostrado como "Usuario inaccesible" o "Grupo de usuarios inaccesible" se eliminarán de la lista de suscriptores del informe.

Hacer clic en *Cancelar* cerrará el formulario de configuración y cancelará la actualización del informe.

Clonación de un informe

Para clonar rápidamente un informe existente, haga clic en el botón *Clonar* en la parte inferior del formulario de configuración del informe existente. Al clonar un informe creado por otro usuario, el usuario actual se convierte en el propietario del nuevo informe.

La configuración del informe se copiará al nuevo formulario de configuración respetando los permisos de usuario:

- Si el usuario que clona un informe no tiene permisos para un panel, el campo *Panel* se borrará.
- Si el usuario que clona un informe no tiene permisos para algunos usuarios o grupos de usuarios en la lista de *Suscripciones*, los destinatarios inaccesibles no se clonarán.
- La configuración de *Generar informe por* se actualizará para mostrar al usuario actual (a menos que *Generar informe por* esté configurado para el destinatario).

Cambie la configuración necesaria y el nombre del informe, luego haga clic en *Agregar*.

Exportación de datos

Descripción general

Zabbix admite la exportación de datos en tiempo real de dos maneras:

- exportar a archivos
- transmisión a sistemas externos

Se pueden exportar las siguientes entidades:

- eventos de disparadores

- valores de métricas
- tendencias (solo exportar a archivos)

Exportar a archivos

Resumen

Es posible configurar la exportación en tiempo real de eventos de trigger, valores de item y tendencias en un formato JSON delimitado por saltos de línea.

La exportación se realiza en archivos, donde cada línea del archivo de exportación es un objeto JSON. No se aplican asignaciones de valores.

En caso de errores (los datos no se pueden escribir en el archivo de exportación, o el archivo de exportación no se puede renombrar, o no se puede crear uno nuevo después de renombrarlo), el elemento de datos se descarta y nunca se escribe en el archivo de exportación. Solo se escribe en la base de datos de Zabbix. La escritura de datos en el archivo de exportación se reanuda cuando se resuelve el problema de escritura.

El archivo de exportación se crea con permisos de lectura y escritura solo para el propietario del archivo. Además, el archivo es legible por el grupo propietario. Todos los demás permisos están denegados.

Tenga en cuenta que host/item puede no tener metadatos (grupos de host, nombre de host, nombre de item) si el host/item se eliminó después de que se recibieron los datos, pero antes de que el server exportara los datos.

Configuración

La exportación en tiempo real de eventos de trigger, valores de item y tendencias se configura especificando un directorio para los archivos de exportación; consulte el parámetro ExportDir en la configuración del server.

Hay otros dos parámetros disponibles:

- `ExportFileSize` se puede utilizar para establecer el tamaño máximo permitido de cada archivo de exportación. Cuando un proceso necesita escribir en un archivo, primero comprueba su tamaño. Si supera el límite de tamaño configurado, el archivo cambia de nombre añadiendo `.old` a su nombre y se crea un archivo nuevo con el nombre original.

Attention:
Se creará un archivo por cada proceso que escriba datos (es decir, aproximadamente entre 4 y 30 archivos). Como el tamaño predeterminado de cada archivo de exportación es de 1G, conservar archivos de exportación grandes puede agotar rápidamente el espacio en disco.

- `ExportType` permite especificar qué tipos de entidades (eventos, historial, tendencias) se exportarán.

Protocolo de exportación JSON delimitado por saltos de línea

Esta sección presenta detalles del protocolo de exportación en un formato JSON delimitado por saltos de línea, utilizado en:

- exportación de datos a archivos
- transmisión a sistemas externos

Todos los archivos tienen la extensión `.ndjson`. Cada línea del archivo de exportación es un objeto JSON.

Exportación de eventos de trigger

La siguiente información se exporta para un evento de problema:

Field	Type	Description
clock	number	Número de segundos transcurridos desde la Época hasta el momento en que se detectó el problema (parte entera).
ns	number	Número de nanosegundos que se añadirán a `clock` para obtener una hora precisa de detección del problema.
value	number	1 (siempre).
eventid	number	ID del evento de problema.
name	string	Nombre del evento de problema.
severity	number	Gravedad del evento de problema (0 - No clasificado, 1 - Información, 2 - Advertencia, 3 - Promedio, 4 - Alto, 5 - Desastre).
hosts	array	Lista de hosts involucrados en la expresión del trigger; debe haber al menos un elemento en el array.
-	object	
host	string	Nombre del host.

Field		Type	Description
	name	string	Nombre visible del host.
groups		array	Lista de grupos de hosts de todos los hosts involucrados en la expresión del trigger; debe haber al menos un elemento en el array.
-		string	Nombre del grupo de hosts.
tags		array	Lista de etiquetas del problema (puede estar vacía).
-		object	
	tag	string	Nombre de la etiqueta.
	value	string	Valor de la etiqueta (puede estar vacío).

La siguiente información se exporta para un evento de recuperación:

Field	Type	Description
clock	number	Número de segundos transcurridos desde la Época hasta el momento en que se resolvió el problema (parte entera).
ns	number	Número de nanosegundos que se añadirán a `clock` para obtener una hora precisa de resolución del problema.
value	number	0 (siempre).
eventid	number	ID del evento de recuperación.
p_eventid	number	ID del evento de problema.

Ejemplos de exportación de eventos de trigger

Problema:

```
{"clock":1519304285,"ns":123456789,"value":1,"name":"Either Zabbix agent is unreachable on Host B or polle
```

Recuperación:

```
{"clock":1519304345,"ns":987654321,"value":0,"eventid":43,"p_eventid":42}
```

Problema (generación de múltiples eventos de problema):

```
{"clock":1519304286,"ns":123456789,"value":1,"eventid":43,"name":"Either Zabbix agent is unreachable on Ho
```

```
{"clock":1519304286,"ns":123456789,"value":1,"eventid":43,"name":"Either Zabbix agent is unreachable on Ho
```

Recuperación:

```
{"clock":1519304346,"ns":987654321,"value":0,"eventid":44,"p_eventid":43}
```

```
{"clock":1519304346,"ns":987654321,"value":0,"eventid":44,"p_eventid":42}
```

Exportación del valor del item

La siguiente información se exporta para un valor de item recopilado:

Field		Type	Description
host		object	Nombre del host del item.
	host	string	Nombre del host.
	name	string	Nombre visible del host.
groups		array	Lista de grupos de host del host del item; debe haber al menos un elemento en el array.
	-	string	Nombre del grupo de host.
item_tags		array	Lista de etiquetas del item (puede estar vacía).
	-	object	
	tag	string	Nombre de la etiqueta.
	value	string	Valor de la etiqueta (puede estar vacío).
itemid		number	ID del item.
name		string	Nombre visible del item.
clock		number	Número de segundos desde la época Unix hasta el momento en que se recopiló el valor (parte entera).

Field	Type	Description
ns	number	Número de nanosegundos que se añadirán a `clock` para obtener un momento de recopilación del valor más preciso.
timestamp (*Log* only)	number	0 si no está disponible.
source (*Log* only)	string	Cadena vacía si no está disponible.
severity (*Log* only)	number	0 si no está disponible.
eventid (*Log* only)	number	0 si no está disponible.
value	number (for numeric items) or string (for text items)	Valor del item recopilado.
type	number	Tipo de valor recopilado: 0 - flotante numérico, 1 - carácter, 2 - log, 3 - entero sin signo, 4 - texto, 5 - binario, 6 - JSON

Ejemplo de exportación del valor de item

Valor numérico (sin signo):


```
{"host":{"host":"Host B","name":"Host B visible"},"groups":["Group X","Group Y","Group Z"],"item_tags":[{"
```


Valor numérico (decimal):


```
{"host":{"host":"Host B","name":"Host B visible"},"groups":["Group X","Group Y","Group Z"],"item_tags":[{"
```


Valor de carácter, texto:


```
{"host":{"host":"Host B","name":"Host B visible"},"groups":["Group X","Group Y","Group Z"],"item_tags":[{"
```


Valor de log:


```
{"host":{"host":"Host A","name":"Host A visible"},"groups":["Group X","Group Y","Group Z"],"item_tags":[{"
```


Exportación de tendencias

La siguiente información se exporta para un valor de tendencia calculado:

Field	Type	Description
host	object	Nombre del host del host del item.
host	string	Nombre del host.
name	string	Nombre visible del host.
groups	array	Lista de grupos de host del host del item; debe haber al menos un elemento en el array.
-	string	Nombre del grupo de host.
item_tags	array	Lista de etiquetas del item (puede estar vacía).
-	object	
tag	string	Nombre de la etiqueta.
value	string	Valor de la etiqueta (puede estar vacío).
itemid	number	ID del item.
name	string	Nombre visible del item.
clock	number	Número de segundos transcurridos desde la época Unix hasta el momento en que se recopiló el valor (parte entera).
count	number	Número de valores recopilados para una hora determinada.
min	number	Valor mínimo del item para una hora determinada.
avg	number	Valor promedio del item para una hora determinada.
max	number	Valor máximo del item para una hora determinada.

Field	Type	Description
type	number	Tipo de valor: 0 - flotante numérico, 3 - entero numérico sin signo

Ejemplos de exportación de tendencias

Valor numérico (sin signo):

```
{"host":{"host":"Host B","name":"Host B visible"},"groups":["Group X","Group Y","Group Z"],"item_tags":[{"
```

Valor numérico (decimal):

```
{"host":{"host":"Host B","name":"Host B visible"},"groups":["Group X","Group Y","Group Z"],"item_tags":[{"
```

Transmisión a sistemas externos

Descripción general

Es posible transmitir valores de item y eventos desde Zabbix a sistemas externos mediante HTTP (consulte los detalles del protocolo).

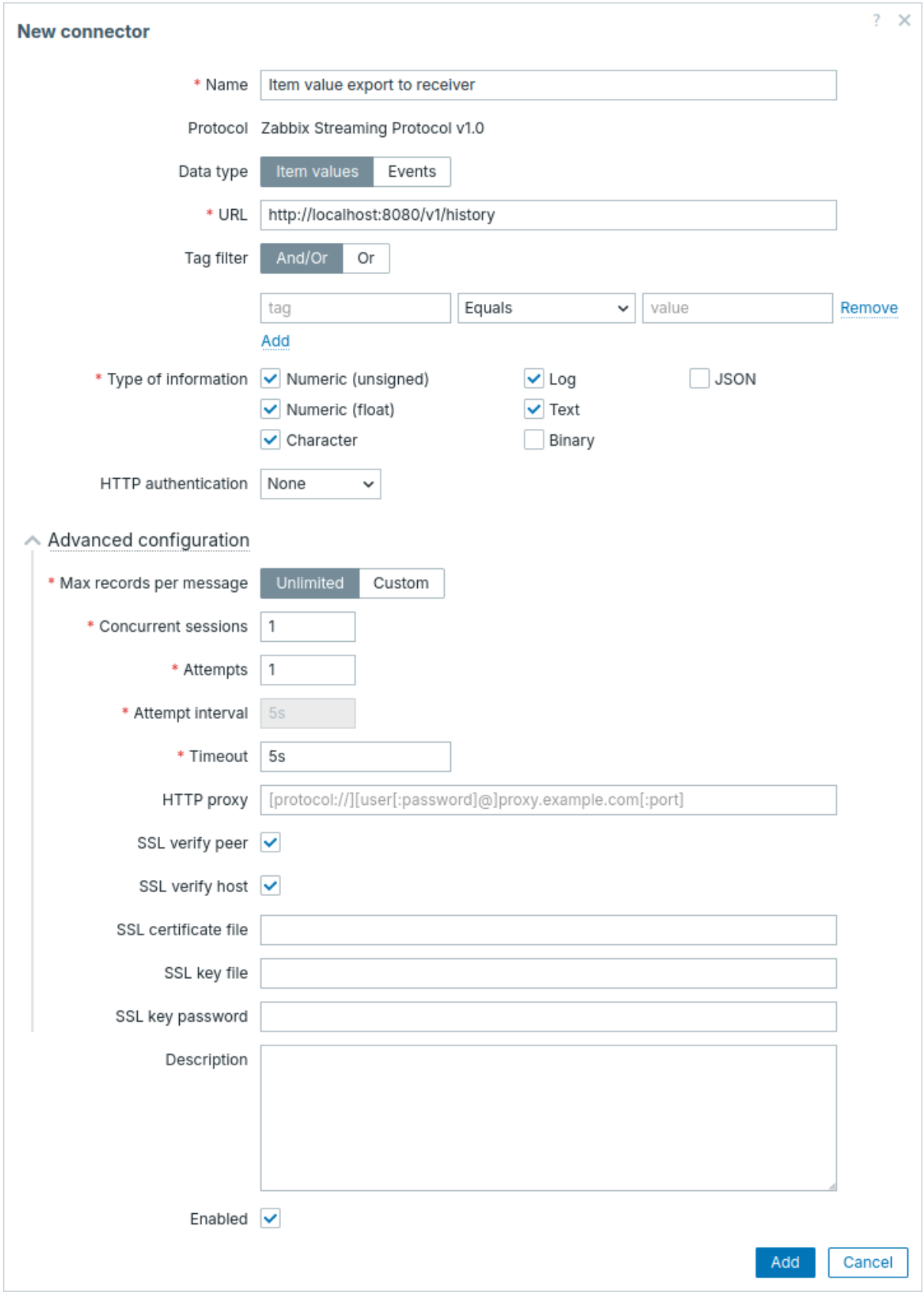
El filtro de etiquetas se puede utilizar para transmitir subconjuntos de valores de item o eventos.

Dos tipos de procesos del server de Zabbix son responsables de la transmisión de datos: `connector manager` y `connector worker`. Un item interno de Zabbix, `zabbix[connector_queue]`, permite supervisar el número de valores encolados en la cola del conector.

Configuración

Se requieren los siguientes pasos para configurar la transmisión de datos a un sistema externo:

1. Configure un sistema remoto para recibir datos de Zabbix. Para este propósito, están disponibles las siguientes herramientas:
   - Un ejemplo de un receptor simple que registra la información recibida en los archivos `events.ndjson` y `history.ndjson`.
   - Conector de Kafka para Zabbix server: un server ligero escrito en Go, diseñado para reenviar valores de item y eventos desde un Zabbix server a un broker de Kafka.
2. Establezca el número necesario de workers de connector en Zabbix ajustando el parámetro `StartConnectors` en `zabbix_server.conf`. El número de workers de connector debe coincidir con el número de connectors configurado en el frontend de Zabbix (o superarlo si hay más de una sesión simultánea). A continuación, reinicie Zabbix server.
3. Configure un nuevo connector en el frontend de Zabbix (*Administración* > *General* > *Connectors*) y recargue la caché del server con el comando `zabbix_server -R config_cache_reload`.



Los campos obligatorios están marcados con un asterisco.

Parámetro	Descripción
Nombre	Introduzca el nombre del connector.
Tipo de datos	Seleccione el tipo de datos que se transmitirá: **Valores de item**: transmite valores de item desde Zabbix a sistemas externos; **Eventos**: transmite eventos desde Zabbix a sistemas externos.
URL	Introduzca la URL del receptor. Se admiten macros de usuario.

Parámetro	Descripción
Filtro de tags	Exporte únicamente los valores de item o eventos que coincidan con el filtro de tags. Si no se establece, se exportará todo. Es posible incluir o excluir tags y valores de tags específicos. Se pueden establecer varias condiciones. La coincidencia de nombres de tag siempre distingue entre mayúsculas y minúsculas. Hay varios operadores disponibles para cada condición: **Existe**: incluye los nombres de tag especificados; **Es igual a**: incluye los nombres y valores de tag especificados (distingue entre mayúsculas y minúsculas); **Contiene**: incluye los nombres de tag especificados cuyos valores contienen la cadena introducida (coincidencia de subcadena, sin distinguir entre mayúsculas y minúsculas); **No existe**: excluye los nombres de tag especificados; **No es igual a**: excluye los nombres y valores de tag especificados (distingue entre mayúsculas y minúsculas); **No contiene**: excluye los nombres de tag especificados cuyos valores contienen la cadena introducida (coincidencia de subcadena, sin distinguir entre mayúsculas y minúsculas). Hay dos tipos de cálculo para las condiciones: **Y/O**: deben cumplirse todas las condiciones; las condiciones que tienen el mismo nombre de tag se agruparán mediante la condición O; **O**: basta con que se cumpla una condición.
Tipo de información	Seleccione el tipo de información (numérico (sin signo), numérico (flotante), carácter, etc.) por el que se filtrarán los valores de item que el connector debe transmitir. Este campo está disponible si *Tipo de datos* está establecido en ”Valores de item”.
Autenticación HTTP	Seleccione la opción de autenticación: **Ninguna**: no se utiliza autenticación; **Básica**: se utiliza autenticación básica; **NTLM**: se utiliza autenticación NTLM (Windows NT LAN Manager); **Kerberos**: se utiliza autenticación Kerberos (consulte también: Configuración de Kerberos con Zabbix); **Digest**: se utiliza autenticación Digest; **Bearer**: se utiliza autenticación Bearer.
Nombre de usuario	Introduzca el nombre de usuario (hasta 255 caracteres). Se admiten macros de usuario. Este campo está disponible si *Autenticación HTTP* está establecido en ”Básica”, ”NTLM”, ”Kerberos” o ”Digest”.
Contraseña	Introduzca la contraseña del usuario (hasta 255 caracteres). Se admiten macros de usuario. Este campo está disponible si *Autenticación HTTP* está establecido en ”Básica”, ”NTLM”, ”Kerberos” o ”Digest”.
Token Bearer	Introduzca el token Bearer. Se admiten macros de usuario. Este campo está disponible y es obligatorio si *Autenticación HTTP* está establecido en ”Bearer”.
Configuración avanzada	Haga clic en el encabezado *Configuración avanzada* para mostrar las opciones de configuración avanzada (consulte más abajo).
Máximo de registros por mensaje	Especifique el número máximo de valores o eventos que se pueden transmitir en un mensaje.
Sesiones simultáneas	Seleccione el número de procesos emisores que se ejecutarán para este connector. Se pueden especificar hasta 100 sesiones; el valor predeterminado es ”1”.
Intentos	Número de intentos para transmitir datos. Se pueden especificar hasta 5 intentos; el valor predeterminado es ”1”.
Intervalo entre intentos	Especifique cuánto tiempo debe esperar el connector después de un intento fallido de transmitir datos. Se pueden especificar hasta 10s; el valor predeterminado es ”5s”. Este campo está disponible si *Intentos* está establecido en ”2” o más. Los intentos fallidos son aquellos en los que no se ha podido establecer una conexión o en los que el código de respuesta HTTP no es 200, 201, 202, 203 o 204. Los reintentos se **activan** en caso de errores de comunicación o cuando el código de respuesta HTTP no es 200, 201, 202, 203, 204, 400, 401, 403, 404, 405, 415 o 422. Se siguen las redirecciones, por lo que 302 -> 200 es una respuesta positiva, mientras que 302 -> 503 activará un reintento.
Tiempo de espera	Especifique el tiempo de espera del mensaje (1-60 segundos, valor predeterminado: 5 segundos). Se admiten sufijos de tiempo (por ejemplo, 30s, 1m). Se admiten macros de usuario.

Parámetro	Descripción
Proxy HTTP	Puede especificar un proxy HTTP con el siguiente formato: `[protocol://][username[:password]@]proxy.example.com[:port]` Se admiten macros de usuario. El prefijo opcional `protocol://` puede utilizarse para especificar protocolos de proxy alternativos (la compatibilidad con el prefijo de protocolo se añadió en cURL 7.21.7). Si no se especifica ningún protocolo, el proxy se tratará como un proxy HTTP. De forma predeterminada, se utilizará el puerto 1080. Si se especifica *Proxy HTTP*, el proxy sobrescribirá las variables de entorno relacionadas con el proxy, como `http_proxy` y `HTTPS_PROXY`. Si no se especifica, el proxy no sobrescribirá las variables de entorno relacionadas con el proxy. El valor introducido se transmite tal cual, sin realizar comprobaciones de validez. También puede introducir una dirección de proxy SOCKS. Si especifica el protocolo incorrecto, el connector no podrá transmitir valores de item o eventos desde Zabbix. Tenga en cuenta que con el proxy HTTP solo se admite la autenticación simple.
Verificar par SSL	Marque la casilla para verificar el certificado SSL del web server. El certificado del server se verifica con los certificados de la autoridad certificadora (CA) de confianza de la ubicación de CA de todo el sistema. Puede sobrescribir la ubicación de los archivos CA mediante el parámetro de configuración de Zabbix server o proxy `SSLCALocation`.
Verificar host SSL	Marque la casilla para verificar que el campo *Common Name* o el campo *Subject Alternate Name* del certificado del web server coincida. Esto establece la opción de cURL `CURLOPT_SSL_VERIFYHOST`.
Archivo de certificado SSL	Nombre del archivo de certificado SSL utilizado para la autenticación del cliente. El archivo de certificado debe estar en formato PEM[1]. Se admiten macros de usuario. Si el archivo de certificado también contiene la clave privada, deje vacío el campo *Archivo de clave SSL*. Si la clave está cifrada, especifique la contraseña en el campo *Contraseña de clave SSL*. El directorio que contiene este archivo se especifica mediante el parámetro de configuración de Zabbix server o proxy `SSLCertLocation`.
Archivo de clave SSL	Nombre del archivo de clave privada SSL utilizado para la autenticación del cliente. El archivo de clave privada debe estar en formato PEM[1]. Se admiten macros de usuario. El directorio que contiene este archivo se especifica mediante el parámetro de configuración de Zabbix server o proxy `SSLKeyLocation`.
Contraseña de clave SSL	Contraseña del archivo de clave privada SSL. Se admiten macros de usuario.
Descripción	Introduzca la descripción del connector.
Habilitado	Marque la casilla para habilitar el connector.

Cuando el conector de Kafka se configura con una lista de direcciones de brokers de arranque separadas por comas (por ejemplo, `Kafka.URL=kafka1.example.com:9093,kafka2.example.com:9093`), el cliente de Kafka se conecta al broker o brokers que responden primero y utiliza sus metadatos del clúster. Si la lista contiene direcciones de distintos clústeres de Kafka, solo se utilizará el clúster que responda más rápido y las demás direcciones se registrarán como no disponibles; como resultado, pueden aparecer advertencias de inicio como la siguiente aunque el connector esté conectado:

```
kafka cluster connected, but broker(s) "kafka1.example.com:9093, kafka2.example.com:9093" unavailable; wil
```

En algunos entornos (redes privadas, redes de contenedores o configuraciones de DNS/hosts no estándar), los nombres de host o las IP pueden resolverse como direcciones de loopback (por ejemplo, `127.0.0.1/localhost`) o ser normalizados por el cliente, lo que puede hacer que estas advertencias resulten engañosas. Para reducir la confusión, asegúrese de que todas las direcciones de `Kafka.URL` pertenezcan al mismo clúster de Kafka, verifique la resolución DNS desde el host del connector y los `advertised.listeners` de los brokers, y prefiera las direcciones que se resuelvan en la dirección anunciada por el broker.

Protocolo

La comunicación entre el server y el receptor se realiza a través de HTTP mediante REST API, NDJSON, "Content-Type: application/x-ndjson".

Para obtener más detalles, consulte Protocolo de exportación JSON delimitado por saltos de línea.

Solicitud del servidor

Ejemplo de transmisión de valores de métricas:

```
POST /v1/history HTTP/1.1
Host: localhost:8080
Accept: */*
Accept-Encoding: deflate, gzip, br, zstd
Content-Length: 628
Content-Type: application/x-ndjson

{"host":{"host":"Zabbix server","name":"Zabbix server"},"groups":["Zabbix servers"],"item_tags":[{"tag":"f
{"host":{"host":"Zabbix server","name":"Zabbix server"},"groups":["Zabbix servers"],"item_tags":[{"tag":"f
{"host":{"host":"Zabbix server","name":"Zabbix server"},"groups":["Zabbix servers"],"item_tags":[{"tag":"b
```

Ejemplo de transmisión de eventos:

```
POST /v1/events HTTP/1.1
Host: localhost:8080
Accept: */*
Accept-Encoding: deflate, gzip, br, zstd
Content-Length: 333
Content-Type: application/x-ndjson

{"clock":1673454303,"ns":800155804,"value":1,"eventid":5,"name":"trigger for foo being 0","severity":0,"ho
{"clock":1673454303,"ns":832290669,"value":0,"eventid":6,"p_eventid":5}
```

Respuesta del receptor

La respuesta consta del código de estado de la respuesta HTTP y de la carga útil JSON. El código de estado de la respuesta HTTP debe ser ”200”, ”201”, ”202”, ”203” o ”204” para las solicitudes que se hayan procesado correctamente; cualquier otro indica solicitudes fallidas.

Ejemplo de éxito:

```
HTTP/1.1 200 OK
Content-Type: application/json
X-Content-Type-Options: nosniff
Date: Tue, 21 Apr 2026 10:13:04 GMT
Content-Length: 23

{"response":"success"}
```

Ejemplo con errores:

```
HTTP/1.1 422 Unprocessable Entity
Content-Type: application/json
X-Content-Type-Options: nosniff
Date: Tue, 21 Apr 2026 12:15:01 GMT
Content-Length: 55

{"error":"invalid character '{' after top-level value"}
```

SNMP gateway

Descripción general

Zabbix SNMP gateway es una extensión AgentX para snmpd que admite tanto sondeo como trapping SNMP.

Con Zabbix SNMP gateway es posible utilizar el protocolo SNMP para recuperar:

- datos de trigger;
- datos de trigger en problema;
- estado de grupos de hosts (conteo de triggers por estado de trigger por grupo)

Los datos se recuperan por el OID, que es una combinación de una base común y un sufijo específico. La **base** común se establece en el archivo de configuración de SNMP gateway, por ejemplo:

- `BaseOID=1.3.6.1.4.1.3043.7.55` - para cualquier dato de trigger;
- `ProblemBaseOID=1.3.6.1.4.1.3047.7.55` - para datos de trigger en problema;
- `BaseOID=1.3.6.1.4.1.3046.7.55` - para el estado de grupos de hosts.

El **sufijo** del OID se establece en la configuración de los triggers del host como una **etiqueta** (por ejemplo, `OIDSuffix:3`) en el frontend.

En este caso, toda la información para el trigger estará disponible bajo `OID=1.3.6.1.4.1.3043.7.55.X.3`. "X" aquí será el número de campos de datos del trigger (es decir, 1 - sufijo, 2 - ID, 3 - expresión, 4 - descripción, etc.).

Para una descripción más detallada y un ejemplo de archivo de configuración, consulte el archivo readme de SNMP gateway.

Instalación y configuración

Consulte el archivo readme de SNMP gateway para obtener instrucciones sobre:

- instalar y configurar snmpd;
- habilitar el soporte AgentX;
- configurar Zabbix SNMP gateway;
- configurar traps SNMP para cambios de estado de trigger.

Recuperando datos

Con todo configurado correctamente, puede utilizar los comandos `snmpwalk` y `snmpget` para recuperar datos:

```
[user@localhost ~]# snmpget -v2c -c public 127.0.0.1 1.3.6.1.4.1.3043.7.55.2.3
SNMPv2-SMI::enterprises.3043.7.55.2.3 = INTEGER: 15247
```

```
[user@localhost ~]# snmpwalk -v2c -c public 127.0.0.1 1.3.6.1.4.1.3043.7.55
SNMPv2-SMI::enterprises.3043.7.55.1.1 = INTEGER: 1
SNMPv2-SMI::enterprises.3043.7.55.1.3 = INTEGER: 3
SNMPv2-SMI::enterprises.3043.7.55.1.4 = INTEGER: 4
SNMPv2-SMI::enterprises.3043.7.55.1.5 = INTEGER: 5
SNMPv2-SMI::enterprises.3043.7.55.1.6 = INTEGER: 6
SNMPv2-SMI::enterprises.3043.7.55.1.10 = INTEGER: 10
SNMPv2-SMI::enterprises.3043.7.55.2.1 = INTEGER: 15367
SNMPv2-SMI::enterprises.3043.7.55.2.3 = INTEGER: 15247
SNMPv2-SMI::enterprises.3043.7.55.2.4 = INTEGER: 15365
SNMPv2-SMI::enterprises.3043.7.55.2.5 = INTEGER: 15366
SNMPv2-SMI::enterprises.3043.7.55.2.6 = INTEGER: 13493
SNMPv2-SMI::enterprises.3043.7.55.2.10 = INTEGER: 13503
...
```

Opciones de filtrado

Puede limitar la información de los triggers de problema en la configuración del gateway SNMP:

- por severidad (por defecto `ProblemMinSeverity=-1`)
- ocultando los problemas reconocidos (por defecto `ProblemHideAck=false`)

Puede limitar el recuento de problemas por grupo de hosts en la configuración del gateway SNMP:

- por triggers en estado desconocido (por defecto `CountUnknown=false`)
- por triggers con problemas reconocidos/no reconocidos/todos (por defecto `CountAcknowledgeStatus=all`)

Exportación/importación de configuración

Descripción general

La funcionalidad de exportación/importación de Zabbix permite intercambiar diversas entidades de configuración entre un sistema Zabbix y otro.

Casos de uso típicos de esta funcionalidad:

- compartir templates o mapas de red: los usuarios de Zabbix pueden compartir sus parámetros de configuración
- cargar un template en Zabbix Community templates. Luego, otros pueden descargar el template e importar el archivo en Zabbix.
- integrarse con herramientas de terceros: los formatos universales YAML, XML y JSON hacen posible la integración y la importación/exportación de datos con herramientas y aplicaciones de terceros

Qué se puede exportar/importar

Los objetos que se pueden exportar/importar son:

- Grupos de host (*solo a través de la API de Zabbix*)

- Grupos de template (*solo a través de la API de Zabbix*)
- Templates
- Hosts
- Mapas de red
- Tipos de medios
- Imágenes
- Paneles
- Expresiones regulares

Formato de exportación

Los datos se pueden exportar usando el frontend web de Zabbix o Zabbix API. Los formatos de exportación compatibles son YAML, XML y JSON.

Detalles sobre la exportación

- Todos los elementos compatibles se exportan en un solo archivo.
- Las entidades de host y template (items, triggers, gráficos, reglas de descubrimiento) que se heredan de templates vinculadas no se exportan. Cualquier cambio realizado en esas entidades a nivel de host (como un intervalo de item modificado, una expresión regular modificada o prototipos añadidos a la regla de descubrimiento de bajo nivel) se perderá al exportar; al importar, todas las entidades de las templates vinculadas se recrean tal como en la template vinculada original.
- Las entidades creadas por el descubrimiento de bajo nivel y cualquier entidad que dependa de ellas no se exportan. Por ejemplo, un trigger creado para un item generado por una regla LLD no se exportará.
- Cuando el host/template exportado contiene entidades que admiten timeouts, los valores de timeout se exportarán si estas entidades tienen sus propios timeouts configurados.

Detalles sobre la importación

- La importación se detiene ante el primer error.
- Al actualizar imágenes existentes durante la importación de imágenes, el campo ”imagetype” se ignora; es decir, no es posible cambiar el tipo de imagen mediante la importación.
- Al importar hosts/templates usando la opción ”Delete missing”, las macros de host/template que no estén presentes en el archivo de importación se eliminarán del host/template después de la importación.
- Las etiquetas vacías para items, triggers, graphs, discoveryRules, itemPrototypes, triggerPrototypes, graphPrototypes no tienen significado; es decir, es lo mismo que si faltaran.
- Si las entidades del host/template importado tienen sus propios timeouts configurados, se aplicarán; de lo contrario, se aplicarán los timeouts de proxy/global.
- La importación admite YAML, XML y JSON; el archivo de importación debe tener una extensión de archivo correcta: .yaml y .yml para YAML, .xml para XML y .json para JSON. Consulte la información de compatibilidad sobre las versiones XML admitidas.
- La importación solo admite archivos de configuración en codificación UTF-8 (con o sin BOM); otras codificaciones (UTF16LE, UTF16BE, UTF32LE, UTF32BE, etc.) darán como resultado un error de conversión de importación.

Formato base de YAML

El formato de exportación YAML contiene los siguientes nodos:

- Nodo raíz para la exportación YAML de Zabbix
- Versión de exportación

```
zabbix_export:
  version: '8.0'
```

Los demás nodos dependen de los objetos exportados.

Formato XML

El formato de exportación XML contiene las siguientes etiquetas:

- Encabezado predeterminado para documentos XML
- Etiqueta raíz para la exportación XML de Zabbix
- Versión de exportación

```
<?xml version="1.0" encoding="UTF-8"?>
<zabbix_export>
    <version>8.0</version>
</zabbix_export>
```

Las demás etiquetas dependen de los objetos exportados.

Formato JSON

El formato de exportación JSON contiene los siguientes objetos:

- Objeto raíz para la exportación JSON de Zabbix
- Versión de exportación

```
{
    "zabbix_export": {
        "version": "8.0"
    }
}
```

Los demás objetos dependen de los objetos exportados.

Grupos de template

Resumen

En el frontend, los grupos de template solo se pueden exportar con la exportación de template. Cuando se exporta un template, todos los grupos a los que pertenece se exportan automáticamente junto con él.

La API permite exportar grupos de template de forma independiente de los templates.

Formato de exportación

```
  template_groups:
    - uuid: 36bff6c29af64692839d077febfc7079
      name: 'Network devices'
```

Elementos exportados

Element	Type	Description
uuid	string	Identificador único de este grupo de template.
name	string	Nombre del grupo.

Grupos de host

Resumen

En el frontend, los grupos de host solo se pueden exportar junto con la exportación de host. Cuando se exporta un host, todos los grupos a los que pertenece se exportan automáticamente con él.

La API permite exportar grupos de host de forma independiente de los hosts.

Formato de exportación

```
   host_groups:
    - uuid: 6f6799aa69e844b4b3918f779f2abf08
      name: 'Zabbix servers'
```

Elementos exportados

Element	Type	Description
uuid	string	Identificador único de este grupo de hosts.
name	string	Nombre del grupo.

Plantillas

Resumen

Los templates se exportan con muchos objetos relacionados y relaciones entre objetos.

La exportación de templates contiene:

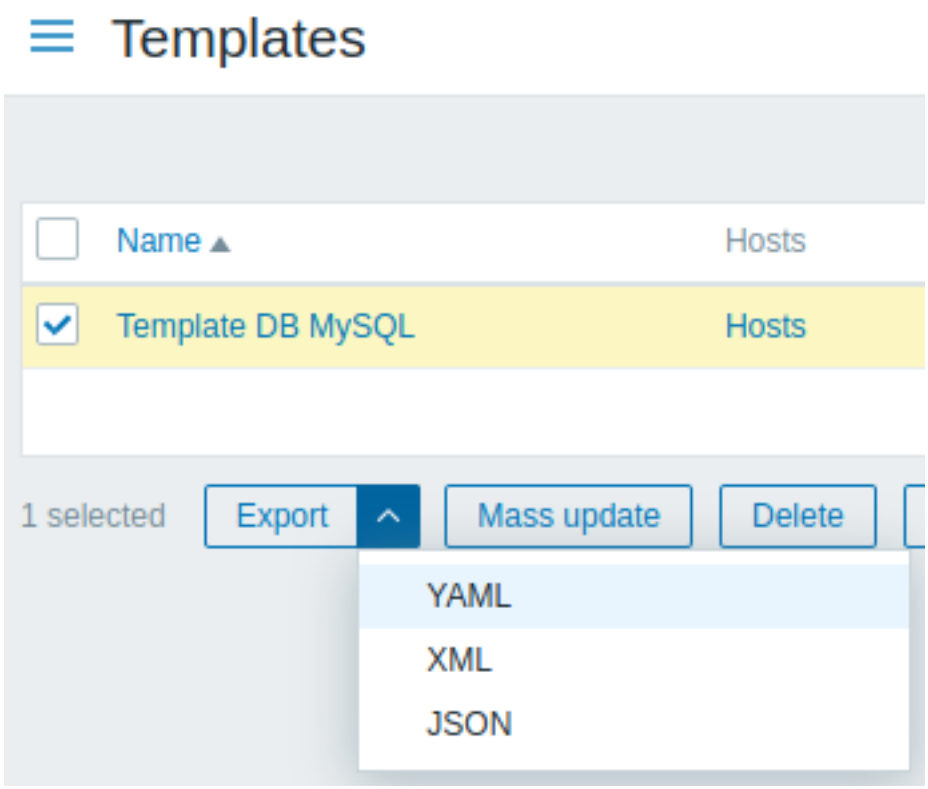
- Grupos de templates vinculados
- Grupos de hosts vinculados (si se usan en la configuración de host prototype)
- Datos del template

- Vinculación con otros templates
- Vinculación con grupos de templates
- items vinculados directamente
- triggers vinculados directamente
- gráficos vinculados directamente
- paneles vinculados directamente
- reglas de descubrimiento vinculadas directamente con todos los prototipos
- escenarios web vinculados directamente
- Mapas de valores

Exportación

Para exportar templates, haga lo siguiente:

1. Vaya a *Data collection* > *Templates*.
2. Marque las casillas de verificación de los templates que desea exportar.
3. Haga clic en *Export* debajo de la lista para exportar en el formato YAML predeterminado, o haga clic en la flecha para seleccionar un formato de exportación diferente.



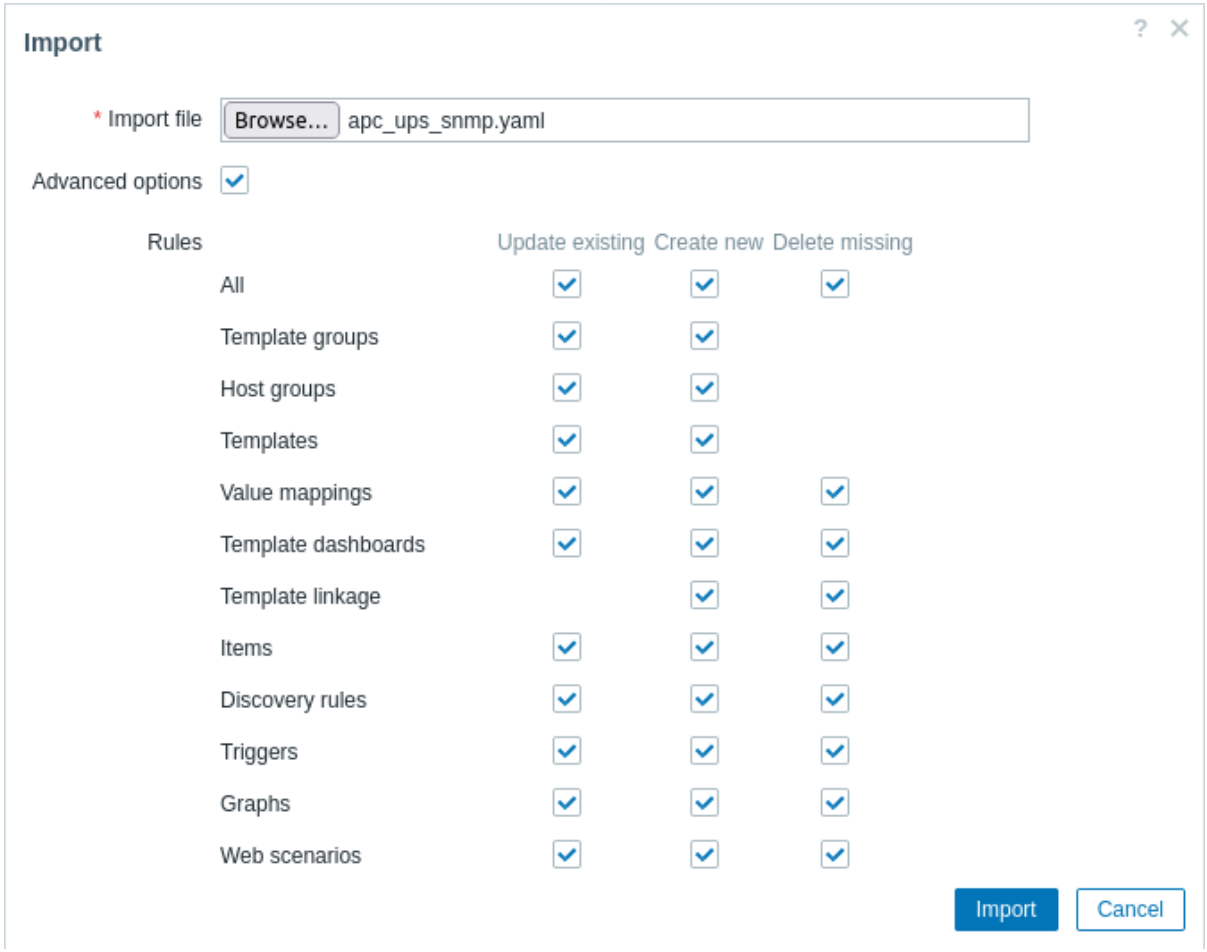
Según el formato seleccionado, los templates se exportan a un archivo local con un nombre predeterminado:

- `zabbix_export_templates.yaml` - en la exportación YAML;
- `zabbix_export_templates.xml` - en la exportación XML;
- `zabbix_export_templates.json` - en la exportación JSON.

Importación

Para importar templates, haga lo siguiente:

1. Vaya a *Recopilación de datos* > *Templates*.
2. Haga clic en *Importar* en la esquina superior derecha.
3. Seleccione el archivo de importación.
4. Marque las casillas de las reglas de importación que desea aplicar (consulte más abajo).
5. Haga clic en *Importar* en la esquina inferior derecha del formulario de configuración.



Si marca la casilla *Opciones avanzadas*, se mostrará una lista detallada de todos los elementos que se pueden importar; marque o desmarque cada regla de importación según sea necesario.

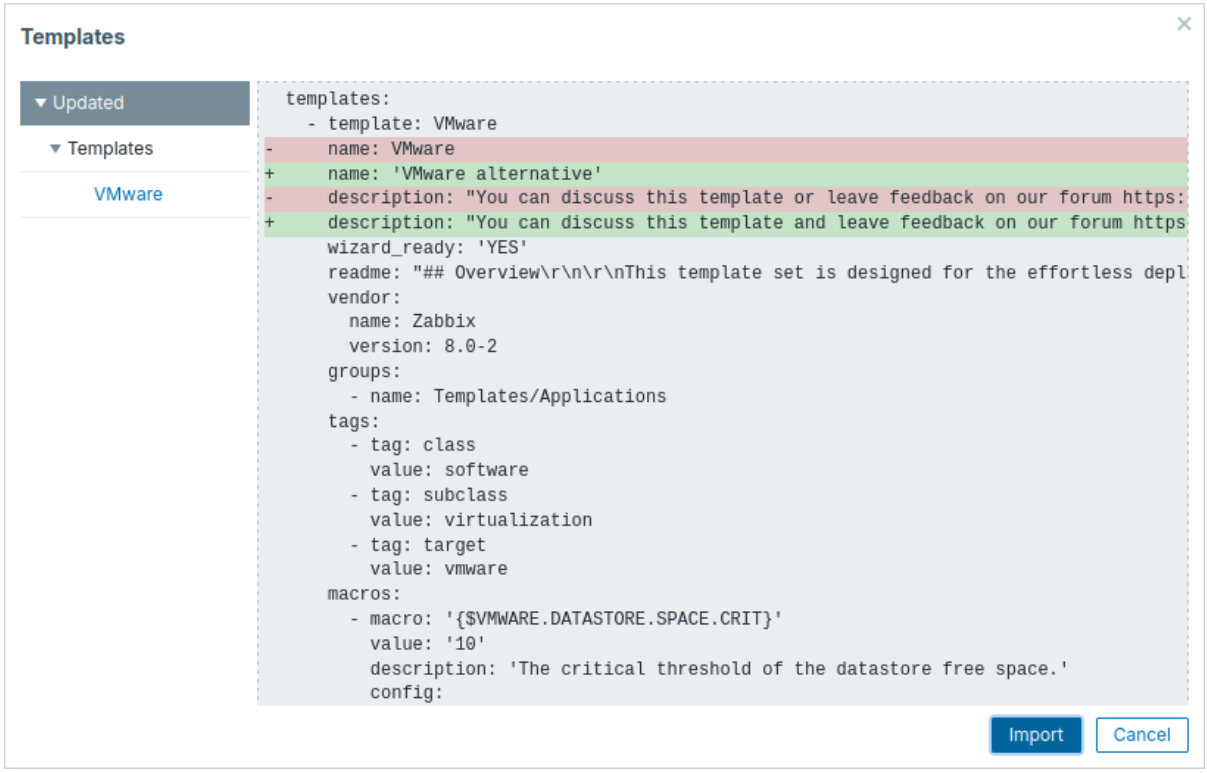
Si hace clic en la casilla de la fila *Todos*, se marcarán o desmarcarán todos los elementos que aparecen debajo.

Reglas de importación:

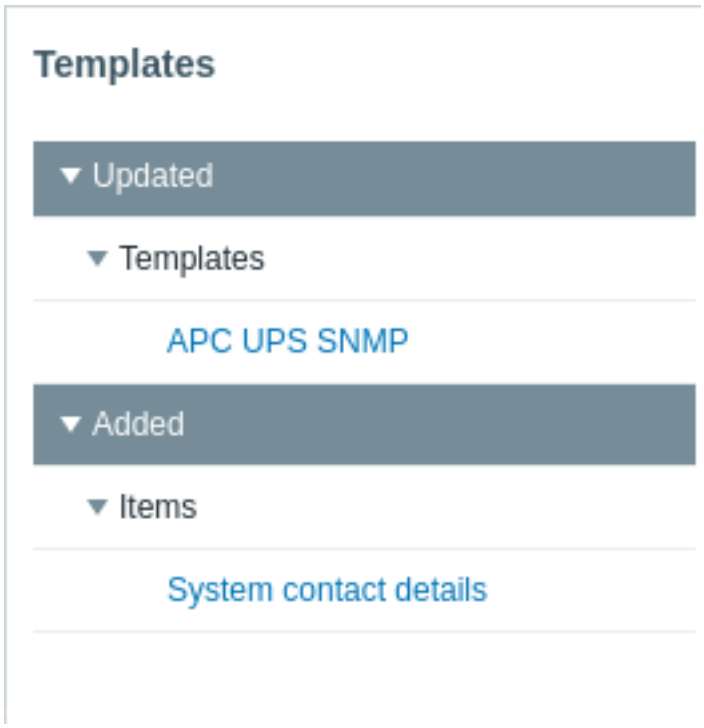
Regla	Descripción
Actualizar existentes	Los elementos existentes se actualizarán utilizando los datos del archivo de importación. De lo contrario, no se actualizarán.
Crear nuevos	Los elementos nuevos se crearán utilizando los datos del archivo de importación. De lo contrario, no se crearán.
Eliminar ausentes	Los elementos existentes que no estén presentes en el archivo de importación se eliminarán. De lo contrario, no se eliminarán. Si se marca *Eliminar ausentes* para *Vinculación de templates*, se desvincularán las vinculaciones actuales de templates que no estén presentes en el archivo de importación. Las entidades (items, triggers, gráficos, etc.) heredadas de los templates desvinculados no se eliminarán (a menos que también se seleccione la opción *Eliminar ausentes* para cada entidad).

Attention:
Si ya existen templates con los mismos nombres, se deben marcar las opciones *Eliminar ausentes* al importar para lograr una importación limpia. De esta forma, se eliminarán los items antiguos que ya no estén en el template actualizado (tenga en cuenta que esto implicará perder el historial de estos items antiguos).

En la siguiente pantalla, podrá ver el contenido del template que se está importando. Si se trata de un template nuevo, todos los elementos aparecerán en verde. Si se está actualizando un template existente, los elementos nuevos del template se resaltarán en verde; los elementos eliminados del template se resaltarán en rojo; los elementos que no hayan cambiado aparecerán sobre un fondo gris.



El menú de la izquierda se puede utilizar para navegar por la lista de cambios. La sección *Actualizados* resalta todos los cambios realizados en los elementos existentes del template. La sección *Añadidos* muestra los elementos nuevos del template. Los elementos de cada sección están agrupados por tipo de elemento; haga clic en la flecha gris para expandir o contraer el grupo de elementos.



Revise los cambios del template y, a continuación, haga clic en *Importar* para realizar la importación del template. En el frontend se mostrará un mensaje indicando si la importación se realizó correctamente o si falló.

Formato de exportación Cuando se exporta un template, Zabbix genera un formato estructurado YAML, JSON o XML. La exportación incluye elementos del template, como metadatos del template, items, macros, triggers, dashboards y más.

Cada elemento cumple una función específica y puede contener elementos anidados.

Las siguientes secciones describen cada elemento del formato de exportación. Los ejemplos usan el template Linux by Zabbix agent.

Una elipsis (`...`) indica elementos omitidos por brevedad. La nota (`see table below`) se usa cuando el elemento se explica

con más detalle en secciones posteriores.

```
zabbix_export:
  version: '8.0'
  template_groups:
    - uuid: 846977d1dfed4968bc5f8bdb363285bc
      name: 'Templates/Operating systems'
  templates: (see table below)
  triggers: (see table below)
  graphs: (see table below)
```

Metadatos de template

Element	Type	Description
version	string	(required) Versión de exportación.
template_groups		(required) Elemento raíz para los grupos de template.
uuid	string	(required) Identificador único para este grupo de template.
name	string	(required) Nombre del grupo de template.
host_groups		Elemento raíz para los grupos de host que se usan en los prototipos de host.
uuid	string	(required) Identificador único para este grupo de host.
name	string	(required) Nombre del grupo de host.
templates		Elemento raíz para templates.
triggers		Elemento raíz para los elementos de trigger de template, que son los mismos que para template item triggers.
graphs		Elemento raíz para los elementos de gráfico de template, que son los mismos que para host graphs.

Templates

```
zabbix_export:
  (...)
  templates:
    - uuid: f8f7908280354f2abeed07dc788c3747
      template: 'Linux by Zabbix agent'
      name: 'Linux by Zabbix agent'
      description: |
        Esta es una template oficial de Linux. Requiere Zabbix agent 8.0 o una versión posterior. (...)
      wizard_ready: 'YES'
      readme: |
        ## Descripción general

        Esta es una template oficial de Linux. Requiere Zabbix agent 8.0 o una versión posterior. (...)
      vendor:
        name: Zabbix
        version: 8.0-2
      groups:
        - name: 'Templates/Sistemas operativos'
      items: (see table below)
      discovery_rules: (see table below)
      tags:
        - tag: class
          value: os
        - tag: target
          value: linux
      macros:
        - macro: '{$AGENT.TIMEOUT}'
          value: 3m
          description: 'Tiempo de espera tras el cual el agent se considera no disponible. Solo funciona p
          config: (see table below)
        - macro: '{$CPU.UTIL.CRIT}'
          value: '90'
```

```
        description: 'Umbral crítico de utilización de CPU expresado en %.'
        config: (see table below)
      (...)
    dashboards: (see table below)
    valuemaps: (see table below)
  (...)
```

Element	Type	Description
uuid	string	(required) Identificador único de esta template.
template	string	(required) Nombre único de la template.
name	string	Nombre visible de la template.
description	text	Descripción de la template.
wizard_ready	text	Indica si la template está disponible para su selección en el Host Wizard. Valores posibles:[1] NO (0, predeterminado), YES (1). Véase también: Template object (`wizard_ready`).
readme	text	Instrucciones de configuración específicas de la template para mostrar en el Host Wizard. Admite formato Markdown.
vendor		Elemento raíz para el proveedor de la template (presente si la template exportada contiene datos del proveedor).
name	string	(required) Nombre del proveedor de la template.
version	string	(required) Versión de la template. Para las templates incluidas de serie, la versión se muestra de la siguiente manera: versión principal de Zabbix, delimitador (”-”), número de revisión (aumenta con cada nueva versión de la template y se reinicia con cada versión principal de Zabbix). Por ejemplo, 7.0-0, 7.0-3, 8.0-0, 8.0-3.
templates		Elemento raíz para las templates vinculadas.
name	string	(required) Nombre de la template.
groups		Elemento raíz para los grupos de templates.
name	string	(required) Nombre del grupo de templates.
items		Elemento raíz para los template items.
discovery_rules		Elemento raíz para las template low-level discovery rules.
httptests		Elemento raíz para los template web scenarios.
tags		Elemento raíz para las etiquetas de la template.
tag	string	(required) Nombre de la etiqueta.
value	string	Valor de la etiqueta.
macros		Elemento raíz para las macros de usuario de la template.
macro	string	(required) Nombre de la macro de usuario.
type	string	Tipo de macro de usuario. Valores posibles:[1] TEXT (0, predeterminado), SECRET_TEXT (1), VAULT (2). Véase también: User macro object (`type`).
value	string	Valor de la macro de usuario.
description	string	Descripción de la macro de usuario.
config		Elemento raíz para la template macro configuration responsable de cómo se muestra la macro en el Host Wizard.
dashboards		Elemento raíz para los template dashboards.
valuemaps		Elemento raíz para los template value maps.

Items de template

```
zabbix_export:
  (...)
  templates:
    (...)
    items:
      - uuid: f94f9f4699e94c369e6c98b2a2f485ce
        name: 'Zabbix agent ping'
        key: agent.ping
        description: 'El agent siempre devuelve "1" para este item. Puede utilizarse junto con `nodata()
        valuemap:
          name: 'Zabbix agent ping status'
        tags:
          - tag: component
```

```
            value: system
      (...)
      - uuid: 58818005e76d46dda14d6592f601ab00
        name: 'Number of installed packages'
        key: system.sw.packages.get
        delay: 1h
        preprocessing: (consulte la tabla siguiente)
        tags:
          - tag: component
            value: os
        triggers: (consulte la tabla siguiente)
      - uuid: 403cebed115441369e94d35d070ca7b8
        name: 'Memory utilization'
        type: DEPENDENT
        key: vm.memory.utilization
        value_type: FLOAT
        units: '%'
        description: 'El porcentaje de memoria utilizada se calcula como `100-pavailable`.'
        preprocessing: (consulte la tabla siguiente)
        master_item:
          key: 'vm.memory.size[pavailable]'
        tags:
          - tag: component
            value: memory
        triggers: (consulte la tabla siguiente)
      (...)
```

Elemento	Tipo	Descripción
uuid	string	(obligatorio) Identificador único de este item.
name	string	(obligatorio) Nombre del item.
type	string	Tipo de item. Valores posibles:[1] ZABBIX_PASSIVE (0, predeterminado), TRAP (2), SIMPLE (3), INTERNAL (5), ZABBIX_ACTIVE (7), EXTERNAL (10), ODBC (11), IPMI (12), SSH (13), TELNET (14), CALCULATED (15), JMX (16), SNMP_TRAP (17), DEPENDENT (18), HTTP_AGENT (19), SNMP_AGENT (20), ITEM_TYPE_SCRIPT (21), ITEM_TYPE_BROWSER (22), TELEMETRY_QUERY (24).
snmp_oid	string	(obligatorio para items SNMP_AGENT) Identificador de objeto SNMP.
key	string	(obligatorio) Clave del item.
delay	string	Intervalo de actualización del item. Predeterminado: 1m. El valor siempre será 0 para los items TRAP.
history	string	Periodo de tiempo (mediante un sufijo de tiempo, una macro de usuario o una macro de LLD) durante el que se deben conservar los datos históricos. Predeterminado: 31d.
trends	string	Periodo de tiempo (mediante un sufijo de tiempo, una macro de usuario o una macro de LLD) durante el que se deben conservar los datos de tendencias. Predeterminado: 365d.
status	string	Estado del item. Valores posibles:[1] ENABLED (0, predeterminado), DISABLED (1).
value_type	string	Tipo de valor recibido. Valores posibles:[1] FLOAT (0), CHAR (1), LOG (2), UNSIGNED (3, predeterminado), TEXT (4), BINARY (5), JSON (6).
allowed_hosts	string	Lista de direcciones IP, separadas por comas, de los hosts autorizados a enviar datos para el item. Compatible con items TRAP y HTTP_AGENT.
units	string	Unidades del valor recibido (bps, B, etc.).
params	text	Parámetros adicionales según el tipo de item (script ejecutado para items SSH y TELNET; consulta SQL para items ODBC; fórmula para items CALCULATED; script para items ITEM_TYPE_SCRIPT y ITEM_TYPE_BROWSER).
ipmi_sensor	string	Sensor IPMI. Compatible con items IPMI.

Elemento	Tipo	Descripción
authtype	string	Tipo de autenticación. Compatible con items SSH y HTTP_AGENT. Valores posibles para items SSH:[1] PASSWORD (0, predeterminado), PUBLIC_KEY (1). Valores posibles para items HTTP_AGENT:[1] NONE (0, predeterminado), BASIC (1), NTLM (2), Kerberos (3) o Digest (4).
username	string	(obligatorio para items SSH y TELNET) Nombre de usuario para la autenticación. Compatible con items SIMPLE, ODBC, JMX y HTTP_AGENT. Cuando se utiliza para items JMX, también debe especificarse `password` (consulte más abajo), o ambos elementos deben dejarse en blanco.
password	string	(obligatorio para items SSH y TELNET) Contraseña para la autenticación. Compatible con items SIMPLE, ODBC, JMX y HTTP_AGENT. Cuando se utiliza para items JMX, también debe especificarse `username` (consulte más arriba), o ambos elementos deben dejarse en blanco.
publickey	string	(obligatorio para items SSH) Nombre del archivo de clave pública.
privatekey	string	(obligatorio para items SSH) Nombre del archivo de clave privada.
description	text	Descripción del item.
inventory_link	string	Campo del inventario del host que se completa mediante el item. Valores posibles:[1] NONE (0), ALIAS (4), etc. (consulte Inventario del host para conocer los campos compatibles).
valuemap		Elemento raíz para los mapas de valores del item.
name	string	(obligatorio) Nombre del mapa de valores que se utilizará para el item.
logtimefmt	string	Formato de la hora en las entradas del registro. Compatible con items de tipo de valor LOG.
preprocessing		Elemento raíz para el preprocesamiento de valores del item.
step		Elemento raíz para los pasos de preprocesamiento de valores de items de template.
jmx_endpoint	string	Endpoint JMX. Compatible con items JMX.
master_item		(obligatorio para items DEPENDENT) Elemento raíz del master item del item dependiente.
key	string	(obligatorio) Clave del master item del item dependiente.
timeout	string	Tiempo de espera de la solicitud de sondeo de datos del item. Compatible con la lista de tipos de item de Timeouts.
url	string	(obligatorio para items HTTP_AGENT) Cadena de URL.
query_fields		Elemento raíz para los parámetros de consulta. Compatible con items HTTP_AGENT.
name	string	(obligatorio para items HTTP_AGENT) Nombre del parámetro de consulta.
value	string	Valor del parámetro de consulta. Compatible con items HTTP_AGENT.
parameters		Elemento raíz para los parámetros definidos por el usuario. Compatible con items ITEM_TYPE_SCRIPT y ITEM_TYPE_BROWSER.
name	string	(obligatorio para items ITEM_TYPE_SCRIPT y ITEM_TYPE_BROWSER) Nombre del parámetro definido por el usuario.
value	string	Valor del parámetro definido por el usuario. Compatible con items ITEM_TYPE_SCRIPT y ITEM_TYPE_BROWSER.
posts	string	Datos del cuerpo de la solicitud HTTP(S). Compatible con items HTTP_AGENT.
status_codes	string	Rangos de códigos de estado HTTP requeridos, separados por comas. Compatible con items HTTP_AGENT.
follow_redirects	string	Seguir las redirecciones de la respuesta durante el sondeo de datos. Compatible con items HTTP_AGENT. Valores posibles:[1] NO (0), YES (1, predeterminado).
post_type	string	Tipo de cuerpo de los datos enviados. Compatible con items HTTP_AGENT. Valores posibles:[1] RAW (0, predeterminado), JSON (2), XML (3).
http_proxy	string	Cadena de conexión del proxy HTTP(S). Compatible con items HTTP_AGENT.
headers		Elemento raíz para los encabezados de la solicitud HTTP(S). Compatible con items HTTP_AGENT.
name	string	(obligatorio para items HTTP_AGENT) Nombre del encabezado.
value	string	(obligatorio para items HTTP_AGENT) Valor del encabezado.

Elemento	Tipo	Descripción
retrieve_mode	string	Parte de la respuesta que se debe almacenar. Compatible con items HTTP_AGENT. Valores posibles:[1] BODY (0, predeterminado), HEADERS (1), BOTH (2).
request_method	string	Tipo de método de solicitud. Compatible con items HTTP_AGENT. Valores posibles:[1] GET (0, predeterminado), POST (1), PUT (2), HEAD (3).
output_format	string	Cómo procesar la respuesta. Compatible con items HTTP_AGENT. Valores posibles:[1] RAW (0, predeterminado), JSON (1).
allow_traps	string	Permitir que el valor se complete de forma similar al item trapper. Compatible con items HTTP_AGENT. Valores posibles:[1] NO (0, predeterminado), YES (1).
ssl_cert_file	string	Ruta del archivo de clave SSL pública. Compatible con items HTTP_AGENT.
ssl_key_file	string	Ruta del archivo de clave SSL privada. Compatible con items HTTP_AGENT.
ssl_key_password	string	Contraseña del archivo de clave SSL. Compatible con items HTTP_AGENT.
verify_peer	string	Indica si se debe validar que el certificado del host sea auténtico. Compatible con items HTTP_AGENT. Valores posibles:[1] NO (0, predeterminado), YES (1).
verify_host	string	Indica si se debe validar que el nombre del host de la conexión coincida con el del certificado del host. Compatible con items HTTP_AGENT. Valores posibles:[1] NO (0, predeterminado), YES (1).
time_shift	string	Cuánto tiempo hacia atrás (antes de la hora actual) termina el intervalo de tiempo que consulta el item. Compatible con items TELEMETRY_QUERY.
lookback_limit	string	Cuánto tiempo hacia atrás (antes de la hora actual) consulta el item en busca de datos cuando no existe una consulta anterior que continuar (por ejemplo, en la primera ejecución del item), o cuando la consulta anterior es más antigua que `delay` (por ejemplo, porque el Zabbix server estaba inactivo). Compatible con items TELEMETRY_QUERY.
granularity	string	Duración de cada segmento en el que se divide el intervalo de tiempo completo de la consulta. Compatible con items TELEMETRY_QUERY.
query		(obligatorio para items TELEMETRY_QUERY) Elemento raíz de la consulta de telemetría. Define qué consulta el item en el origen de datos, cómo filtra los datos y cómo agrega el resultado.
signal_type	string	(obligatorio) Tipo de señal que consulta el item. Compatible con items TELEMETRY_QUERY. Valores posibles:[1] TRACES (0), METRICS (1), LOGS (2).
metric_point_type	string	(obligatorio para el tipo de señal METRICS) Tipo de punto de métrica que consulta el item. Compatible con items TELEMETRY_QUERY. Valores posibles:[1] SUM (0), GAUGE (1), HISTOGRAM (2), EXPONENTIAL_HISTOGRAM (3).
columns		(obligatorio) Elemento raíz para las columnas de datos de telemetría que consulta el item. Este elemento contiene los siguientes elementos: `column (string)` - (obligatorio) Nombre de la columna de datos de telemetría que consulta el item. `attribute_key (string)` - (obligatorio para columnas que contienen pares clave-valor) Nombre de la clave que consulta el item dentro de la columna que contiene pares clave-valor.

Elemento	Tipo	Descripción
aggregated_columns		(obligatorio) Elemento raíz para las columnas agregadas de datos de telemetría que consulta el item. Este elemento contiene los siguientes elementos: `column` (`string`) - (obligatorio para las funciones MIN, MAX, AVG, SUM y PERCENTILE) Nombre de la columna de datos de telemetría que consulta el item. `function` (`string`) - (obligatorio) Función que el item aplica a los registros de la columna. Valores posibles:[1] MIN (1), MAX (2), AVG (3), COUNT (4), SUM (5), PERCENTILE (8). `parameters` (`array`) - (obligatorio para las funciones PERCENTILE) Percentil que calcula el item a partir de los registros de la columna. `alias` (`string`) - (obligatorio) Nombre único del valor calculado, utilizado para identificarlo en el valor del item.
filter		(obligatorio) Elemento raíz del filtro de consulta de telemetría para los registros de columnas de la consulta. Este elemento contiene los siguientes elementos: `evaltype` (`string`) - (obligatorio) Método de evaluación de las condiciones del filtro. Valores posibles:[1] AND_OR (0), AND (1), OR (2), FORMULA (3). `formula` (`string`) - (obligatorio para los métodos de evaluación FORMULA) Expresión definida por el usuario que se utilizará para evaluar las condiciones de los filtros con una expresión personalizada. `conditions` (`string`) - (obligatorio) Elemento raíz de las condiciones del filtro de datos de telemetría que debe cumplir un registro de una columna de datos de telemetría. El elemento `conditions` contiene los siguientes elementos: `column` (`string`) - (obligatorio) Nombre de la columna de datos de telemetría a la que se aplican las condiciones del filtro. `attribute_key` (`string`) - (obligatorio para columnas que contienen pares clave-valor) Nombre de la clave a la que se aplica la condición dentro de la columna que contiene pares clave-valor. `operator` (`string`) - (obligatorio) Operador de la condición del filtro. Valores posibles para columnas de un solo valor:[1] EQUAL (0), NOT_EQUAL (1), LIKE (2), NOT_LIKE (3). Valores posibles para columnas que contienen pares clave-valor:[1] EQUAL (0), NOT_EQUAL (1), EXISTS (12). `formulaid` (`string`) - (obligatorio para los métodos de evaluación FORMULA) ID único arbitrario utilizado para hacer referencia a la condición desde una expresión personalizada. `value` (`string`) - (obligatorio para las operaciones EQUAL, NOT_EQUAL, LIKE y NOT LIKE) Valor con el que se debe comparar.
tags		Elemento raíz para las etiquetas del item.
tag	string	(obligatorio) Nombre de la etiqueta.
value	string	Valor de la etiqueta.
triggers		Elemento raíz para los triggers de items de template.

Note:
Véase también: Objeto item (consulte la propiedad correspondiente con un nombre coincidente).

Pasos de preprocesamiento del valor del item de template

```
zabbix_export:
  (...)
  templates:
    (...)
      items:
        (...)
        - uuid: 58818005e76d46dda14d6592f601ab00
          (...)
          preprocessing:
            - type: JSONPATH
```

```
            parameters:
              - $.length()
          - type: DISCARD_UNCHANGED_HEARTBEAT
            parameters:
              - 12h
        (...)
```

Elemento	Tipo	Descripción
type	string	(obligatorio) El tipo de paso de preprocesamiento del valor del item. Valores posibles:[1] MULTIPLIER (1), RTRIM (2), LTRIM (3), TRIM (4), REGEX (5), BOOL_TO_DECIMAL (6), OCTAL_TO_DECIMAL (7), HEX_TO_DECIMAL (8), SIMPLE_CHANGE (9, calculado: valor recibido - valor anterior), CHANGE_PER_SECOND (10, calculado: (valor recibido - valor anterior)/(hora actual - hora de la última comprobación)), XMLPATH (11), JSONPATH (12), IN_RANGE (13), MATCHES_REGEX (14), NOT_MATCHES_REGEX (15), CHECK_JSON_ERROR (16), CHECK_XML_ERROR (17), CHECK_REGEX_ERROR (18), DISCARD_UNCHANGED (19), DISCARD_UNCHANGED_HEARTBEAT (20), JAVASCRIPT (21), PROMETHEUS_PATTERN (22), PROMETHEUS_TO_JSON (23), CSV_TO_JSON (24), STR_REPLACE (25), CHECK_NOT_SUPPORTED (26), XML_TO_JSON (27), SNMP_WALK_VALUE (28), SNMP_WALK_TO_JSON (29), SNMP_GET_VALUE (30).
parameters		(obligatorio) Elemento raíz para los parámetros del paso de preprocesamiento del valor del item.
parameter	string	Parámetro individual del paso de preprocesamiento del valor del item.
error_handler	string	Tipo de acción utilizado en caso de fallo del paso de preprocesamiento. Valores posibles:[1] ORIGINAL_ERROR (0, predeterminado), DISCARD_VALUE (1), CUSTOM_VALUE (2), CUSTOM_ERROR (3).
error_handler_params	string	Parámetros del controlador de errores.

Note:
Véase también: Objeto de preprocesamiento de item (consulte la propiedad correspondiente con el mismo nombre).

Triggers de item de template

```
zabbix_export:
  (...)
  templates:
    (...)
      items:
        (...)
        - uuid: 58818005e76d46dda14d6592f601ab00
          (...)
          triggers:
            - uuid: b950c306394f4b3c902060a8273cbcde
              expression: 'change(/Linux by Zabbix agent/system.sw.packages.get)<>0'
              name: 'Linux: El número de paquetes instalados ha cambiado'
              priority: WARNING
              manual_close: 'YES'
              tags:
                - tag: scope
                  value: notice
        (...)
        - uuid: 403cebed115441369e94d35d070ca7b8
          (...)
          triggers:
            - uuid: cfd395b1cde74ef18a5e5f840bd5142a
              expression: 'min(/Linux by Zabbix agent/vm.memory.utilization,5m)>{$MEMORY.UTIL.MAX}'
              name: 'Linux: Alto uso de memoria'
              event_name: 'Linux: Alto uso de memoria (>{$MEMORY.UTIL.MAX}% durante 5m)'
              priority: AVERAGE
              description: 'El sistema se está quedando sin memoria libre.'
              dependencies:
```

```
              - name: 'Linux: Falta de memoria disponible'
                expression: 'max(/Linux by Zabbix agent/vm.memory.size[available],5m)<{$MEMORY.AVAILABLE
            tags:
              - tag: scope
                value: capacity
              - tag: scope
                value: performance
      (...)
```

Element	Type	Description
uuid	string	(required) Identificador único de este trigger.
expression	string	(required) Expresión del trigger.
recovery_mode	string	Base para generar eventos OK. Valores posibles:[1] EXPRESSION (0, predeterminado), RECOVERY_EXPRESSION (1), NONE (2).
recovery_expression	string	Expresión de recuperación del trigger.
correlation_mode	string	Modo de correlación (sin correlación de eventos o correlación de eventos por etiqueta). Valores posibles:[1] DISABLED (0, predeterminado), TAG_VALUE (1).
correlation_tag	string	Nombre de la etiqueta que se utilizará para la correlación de eventos.
name	string	(required) Nombre del trigger.
event_name	string	Nombre del evento.
opdata	string	Datos operativos.
url_name	string	Etiqueta para la URL asociada al trigger.
url	string	URL asociada al trigger.
status	string	Estado del trigger. Valores posibles:[1] ENABLED (0, predeterminado), DISABLED (1).
priority	string	Severidad del trigger. Valores posibles:[1] NOT_CLASSIFIED (0, predeterminado), INFO (1), WARNING (2), AVERAGE (3), HIGH (4), DISASTER (5).
description	text	Descripción del trigger.
type	string	Tipo de generación de eventos (un solo evento de problema o varios eventos de problema). Valores posibles:[1] SINGLE (0, predeterminado), MULTIPLE (1).
manual_close	string	Cierre manual de eventos de problema. Valores posibles:[1] NO (0, predeterminado), YES (1).
dependencies		Elemento raíz para las dependencias.
name	string	(required) Nombre del trigger dependiente.
expression	string	(required) Expresión del trigger dependiente.
recovery_expression	string	Expresión de recuperación del trigger dependiente.
tags		Elemento raíz para las etiquetas del trigger.
tag	string	(required) Nombre de la etiqueta.
value	string	Valor de la etiqueta.

Note:
Véase también: Trigger object (consulte la propiedad correspondiente con el mismo nombre).

Reglas de descubrimiento de bajo nivel de template

```
zabbix_export:
  (...)
  templates:
    (...)
      discovery_rules:
        - uuid: acfdea9c46ef48c68e6636f43b8f96a2
          name: 'Network interface discovery'
          key: net.if.discovery
          delay: 1h
          filter: (see table below)
          description: 'The discovery of network interfaces.'
          item_prototypes: (see table below)
          trigger_prototypes: (see table below)
```

```
        graph_prototypes: (see table below)
    (...)
```

Attention:
La mayoría de los elementos de las reglas de descubrimiento de bajo nivel de template son los mismos que los de template items. La siguiente tabla describe aquellos elementos que difieren de template items.

Elemento	Tipo	Descripción
type	string	Tipo de item. Valores posibles:[1] ZABBIX_PASSIVE (0, predeterminado), TRAP (2), SIMPLE (3), INTERNAL (5), ZABBIX_ACTIVE (7), EXTERNAL (10), ODBC (11), IPMI (12), SSH (13), TELNET (14), JMX (16), DEPENDENT (18), HTTP_AGENT (19), SNMP_AGENT (20), ITEM_TYPE_SCRIPT (21), ITEM_TYPE_BROWSER (22).
key	string	(obligatorio) La clave de la regla de descubrimiento de bajo nivel.
filter		Elemento raíz para template low-level discovery rule filters.
lifetime	string	Período de tiempo (usando segundos, time suffix o user macro), tras el cual se eliminarán los recursos descubiertos que ya no existan. Predeterminado: 7d.
lifetime_type	string	Escenario para eliminar recursos LLD perdidos. Valores posibles: DELETE_NEVER, DELETE_IMMEDIATELY, DELETE_AFTER.
enabled_lifetime	string	Período de tiempo (usando segundos, time suffix o user macro), tras el cual se deshabilitarán los recursos descubiertos que ya no existan.
enabled_lifetime_type	string	Escenario para deshabilitar recursos LLD perdidos. Valores posibles: DISABLE_NEVER, DISABLE_IMMEDIATELY, DISABLE_AFTER.
item_prototypes		Elemento raíz para los elementos de prototipo de item de template, que son los mismos que para template items.
trigger_prototypes		Elemento raíz para los elementos de prototipo de trigger de template, que son los mismos que para template item triggers.
graph_prototypes		Elemento raíz para los elementos de prototipo de gráfico de template, que son los mismos que para host graphs.
host_prototypes		Elemento raíz para los elementos de prototipo de host de template, que son los mismos que para hosts.
parent_discovery_rule		Elemento raíz para la regla de descubrimiento de bajo nivel principal (o prototipo de regla) del prototipo de regla de descubrimiento de bajo nivel. Esta propiedad indica que se trata de un prototipo de regla LLD, hijo directo de la regla/prototipo de regla referenciado.
key	string	(obligatorio) La clave de la regla de descubrimiento de bajo nivel principal (o prototipo de regla).
master_item	string	(obligatorio para reglas DEPENDENT) Elemento raíz para el item maestro de la regla dependiente.
lld_macro_paths		Elemento raíz para las rutas de macros de la regla de descubrimiento de bajo nivel.
lld_macro	string	(obligatorio) Nombre de la macro de la regla de descubrimiento de bajo nivel.
path	string	(obligatorio) Selector del valor que se asignará a la macro correspondiente.
preprocessing		Elemento raíz para el preprocesamiento del valor de la regla de descubrimiento de bajo nivel.
step		Elemento raíz para los elementos de paso de preprocesamiento del valor de la regla de descubrimiento de bajo nivel, que son los mismos que para template item value preprocessing steps, excepto que tienen menos valores posibles. Consulte también: LLD rule preprocessing object.
overrides		Elemento raíz para las reglas de anulación de la regla de descubrimiento de bajo nivel.
name	string	(obligatorio) Nombre único de la anulación.
step	string	(obligatorio) Número de orden único de la anulación.
stop	string	Detener el procesamiento de las siguientes anulaciones si coincide.
filter		Elemento raíz para los elementos de filtro de la regla de anulación de la regla de descubrimiento de bajo nivel de template, que son los mismos que para template low-level discovery rule filters.
operations		Elemento raíz para template low-level discovery rule override operations.

Note:
Consulte también: LLD rule object (consulte la propiedad correspondiente con el nombre coincidente).

Filtros de la regla de descubrimiento de bajo nivel de template

```
zabbix_export:
  (...)
  templates:
    (...)
      discovery_rules:
        - uuid: acfdea9c46ef48c68e6636f43b8f96a2
          (...)
          filter:
            evaltype: AND
            conditions:
              - macro: '{#IFNAME}'
                value: '{$NET.IF.IFNAME.MATCHES}'
              - macro: '{#IFNAME}'
                value: '{$NET.IF.IFNAME.NOT_MATCHES}'
                operator: NOT_MATCHES_REGEX
          (...)
        - uuid: 2bbdc79f082d4c618e01bec625e9c90a
          (...)
          filter:
            evaltype: AND
            conditions:
              - macro: '{#DEVNAME}'
                value: '{$VFS.DEV.DEVNAME.MATCHES}'
              - macro: '{#DEVNAME}'
                value: '{$VFS.DEV.DEVNAME.NOT_MATCHES}'
                operator: NOT_MATCHES_REGEX
              - macro: '{#DEVTYPE}'
                value: disk
          (...)
```

Element	Type	Description
evaltype	string	Anula el método de evaluación de las condiciones del filtro. Valores posibles:[1] AND_OR (0, predeterminado), AND (1), OR (2), FORMULA (3).
formula	string	Fórmula de cálculo personalizada para las condiciones del filtro.
conditions		Elemento raíz para las condiciones del filtro.
macro	string	(obligatorio) El nombre de la macro de descubrimiento de bajo nivel sobre la que realizar la comprobación.
value	string	Valor con el que comparar.
operator	string	Operador de la condición. Valores posibles:[1] MATCHES_REGEX (8, predeterminado), NOT_MATCHES_REGEX (9).
formulaid	string	(obligatorio) ID único arbitrario que se usa para referenciar una condición desde la expresión personalizada. Solo puede contener letras mayúsculas. El ID debe ser definido por el usuario al modificar las condiciones del filtro, pero se generará de nuevo al solicitarlas posteriormente.

Note:
Véase también: Objeto de filtro de la regla LLD (consulte la propiedad correspondiente con el mismo nombre).

Operaciones de anulación de la regla de descubrimiento de bajo nivel de template

Element	Type	Description
operationobject	string	Objeto al que se aplica la operación. Valores posibles:[1] ITEM_PROTOTYPE (0), TRIGGER_PROTOTYPE (1), GRAPH_PROTOTYPE (2), HOST_PROTOTYPE (3).
operator	string	Operador de la condición de anulación. Valores posibles:[1] EQUAL (1), NOT_EQUAL (2), LIKE (3), NOT_LIKE (4), REGEXP (5), NOT_REGEXP (6).
value	string	Una expresión regular o una cadena para el operador de condición de anulación.
status	string	Estado del objeto tras la operación de anulación.

Element	Type	Description
discover	string	Indica si el objeto se añade como resultado del descubrimiento.
delay	string	Intervalo de actualización establecido para el item prototype tras la operación de anulación.
history	string	Período de almacenamiento del historial establecido para el item prototype tras la operación de anulación.
trends	string	Período de almacenamiento de tendencias establecido para el item prototype tras la operación de anulación.
severity	string	Severidad del trigger prototype establecida tras la operación de anulación.
tags		Elemento raíz para las etiquetas establecidas para el objeto tras la operación de anulación.
tag	string	(obligatorio) Nombre de la etiqueta.
value	string	Valor de la etiqueta.
templates		Elemento raíz para los templates vinculados al host prototype tras la operación de anulación.
name	string	(obligatorio) Nombre del template.
inventory_mode	string	Modo de inventario del host prototype establecido tras la operación de anulación.

Note:
Véase también: Objeto de operación de anulación de la regla LLD (consulte la propiedad correspondiente con el mismo nombre).

Escenarios web de template

Element	Type	Description
uuid	string	(required) Identificador único de este escenario web.
name	string	(required) Nombre del escenario web.
delay	string	Frecuencia (usando segundos, time suffix o user macro) de ejecución del escenario web. Default: 1m.
attempts	integer	Número de intentos para ejecutar los pasos del escenario web. Possible values:[1] 1-10 (default: 1).
agent	string	Agent del cliente. Zabbix se hará pasar por el navegador seleccionado. Esto es útil cuando un sitio web devuelve contenido diferente para distintos navegadores. Default: Zabbix.
http_proxy	string	Proxy que se usará para el escenario web, indicado como: `http://[username[:password]@]proxy.example.com[:port]`
variables		Elemento raíz para las variables del escenario web que pueden usarse en los pasos del escenario.
name	string	(required) Nombre de la variable.
value	text	(required) Valor de la variable.
headers		Elemento raíz para los encabezados HTTP que se enviarán al realizar una solicitud. Los encabezados deben enumerarse usando la misma sintaxis con la que aparecerían en el protocolo HTTP.
name	string	(required) Nombre del encabezado.
value	text	(required) Valor del encabezado.
status	string	Estado del escenario web. Possible values:[1] ENABLED (0, default), DISABLED (1).
authentication	string	Método de autenticación. Possible values:[1] NONE (0, default), BASIC (1), NTLM (2), Kerberos (3), or Digest (4).
http_user	string	Nombre de usuario usado para autenticación BASIC (HTTP), NTLM, Kerberos o Digest.
http_password	string	Contraseña usada para autenticación BASIC (HTTP), NTLM, Kerberos o Digest.
verify_peer	string	Verificar el certificado SSL del servidor web. Possible values:[1] NO (0, default), YES (1).
verify_host	string	Verificar que el campo *Common Name* o el campo *Subject Alternate Name* del certificado del servidor web coincida. Possible values:[1] NO (0, default), YES (1).
ssl_cert_file	string	Nombre del archivo de certificado SSL usado para la autenticación del cliente (debe estar en formato PEM).

Element	Type	Description
ssl_key_file	string	Nombre del archivo de clave privada SSL usado para la autenticación del cliente (debe estar en formato PEM).
ssl_key_password	string	Contraseña del archivo de clave privada SSL.
steps		(required) Elemento raíz para template web scenario steps.
tags		Elemento raíz para las etiquetas del escenario web.
tag	string	(required) Nombre de la etiqueta.
value	string	Valor de la etiqueta.

Note:
See also: Web scenario object (refer to the relevant property with a matching name).

Pasos del escenario web de template

Element	Type	Description
name	string	(required) Nombre del paso del escenario web.
url	string	(required) URL para la monitorización.
query_fields		Elemento raíz para los parámetros de consulta (un array de campos HTTP que se añadirán a la URL al realizar una solicitud).
name	string	(required) Nombre del parámetro de consulta.
value	string	Valor del parámetro de consulta.
posts		Elemento raíz para las variables HTTP POST (una cadena (datos POST sin procesar) o un array de campos HTTP (datos de campos de formulario)).
name	string	(required) Nombre del campo POST.
value	string	(required) Valor del campo POST.
variables		Elemento raíz de las variables a nivel de paso (macros) que deben aplicarse después de este paso. Si el valor de la variable tiene el prefijo 'regex:', su valor se extrae de los datos devueltos por este paso según el patrón de expresión regular que sigue al prefijo 'regex:'
name	string	(required) Nombre de la variable.
value	text	(required) Valor de la variable.
headers		Elemento raíz para las cabeceras HTTP que se enviarán al realizar una solicitud.
name	string	(required) Nombre de la cabecera.
value	text	(required) Valor de la cabecera.
follow_redirects	string	Seguir redirecciones HTTP. Valores posibles:[1] NO (0), YES (1, predeterminado).
retrieve_mode	string	Modo de recuperación de la respuesta HTTP. Valores posibles:[1] BODY (0, predeterminado), HEADERS (1), BOTH (2).
timeout	string	Tiempo de espera (en segundos, sufijo de tiempo o macro de usuario) de la ejecución del paso. Predeterminado: 15s.
required	string	Texto que debe estar presente en la respuesta (se ignora si está vacío).
status_codes	string	Lista separada por comas de códigos de estado HTTP aceptados (por ejemplo, 200-201,210-299; se ignora si está vacío).

Note:
Véase también: Objeto de paso del escenario web (consulte la propiedad correspondiente con el mismo nombre).

Configuración de macros de template

```
zabbix_export:
  (...)
  templates:
    - uuid: f8f7908280354f2abeed07dc788c3747
      (...)
      macros:
        - macro: '{$AGENT.TIMEOUT}'
          (...)
```

```
        config:
          type: TEXT
          priority: '1'
          label: 'Segundos desde la última vez que se detectó el agent de Zabbix'
          description: 'Tiempo de espera tras el cual se considera que el agent no está disponible.'
      - macro: '{$CPU.UTIL.CRIT}'
        (...)
        config:
          type: TEXT
          priority: '2'
          section_name: Thresholds
          label: 'Umbral de utilización de CPU expresado'
          description: 'Umbral crítico de utilización de CPU expresado en %. En el rango de 0 a 100 incl
          regex: '^-?([0-9]+|([0-9]+)\.([0-9]+))$'
      (...)
      - macro: '{$IFCONTROL}'
        (...)
        config:
          type: CHECKBOX
          priority: '19'
          label: 'Control de interfaces'
          description: 'Genera un trigger cuando el estado operativo de la interfaz cambia a "Link down"
          options:
            - checked: '1'
              unchecked: '0'
      (...)
```

Elemento	Tipo	Descripción
type	string	(obligatorio) Tipo del campo de entrada de la macro. Valores posibles:[1] NOCONF (0), TEXT (1), LIST (2), CHECKBOX (3).
priority	string	Posición de la macro en la lista de macros.
section_name	string	Etiqueta de la sección plegable donde se agrupa la macro.
label	string	(obligatorio para macros TEXT, LIST, CHECKBOX) Etiqueta de la macro.
description	text	Texto de ayuda de la macro. Admite formato Markdown.
required	string	Marca la macro como obligatoria. Valores posibles:[1] NO (0), YES (1). Compatible con macros TEXT y LIST.
regex	string	Expresión regular para validar la entrada del usuario en un campo de texto. Compatible con macros TEXT.
options		Elemento raíz para elementos LIST o valores CHECKBOX.
value	string	(obligatorio para macros LIST) Valor del elemento LIST.
text	string	(obligatorio para macros LIST) Texto del elemento LIST.
checked	string	(obligatorio para macros CHECKBOX) Valor que representa el estado marcado. Valores posibles:[1] 0 (false), 1 (true). Compatible con macros CHECKBOX.
unchecked	string	(obligatorio para macros CHECKBOX) Valor que representa el estado desmarcado. Valores posibles:[1] 0 (false), 1 (true). Compatible con macros CHECKBOX.

Note:
Véase también: Objeto de configuración de macros (consulte la propiedad correspondiente con el mismo nombre).

Paneles de template

```
zabbix_export:
  (...)
  templates:
    (...)
      dashboards:
        - uuid: c689ad3115fd46a4b927d1f70ee2e5a4
          name: Filesystems
```

```
        pages:
          - name: Overview
            widgets: (see table below)
      (...)
```

Elemento	Tipo	Descripción
uuid	string	(obligatorio) Identificador único de este panel.
name	string	(obligatorio) Nombre del panel de template.
display pe- riod	integer	Periodo de visualización de las páginas del panel.
auto_start	string	Inicio automático de la presentación de diapositivas. Valores posibles:[1] NO (0), YES (1, predeterminado).
pages		Elemento raíz para las páginas del panel de template.
name	string	Nombre de la página.
display period	integer	Intervalo de la presentación de diapositivas.
sortorder	integer	Orden de clasificación de la página.
widgets		Elemento raíz para widgets del panel de template.

Note:
Véase también: Objeto de panel de template (consulte la propiedad correspondiente con el mismo nombre).

Widgets del dashboard de template

```
zabbix_export:
  (...)
  templates:
    (...)
      dashboards:
        - uuid: c689ad3115fd46a4b927d1f70ee2e5a4
          name: Filesystems
          pages:
            - name: Overview
              widgets:
                - type: graphprototype
                  width: '72'
                  height: '5'
                  fields:
                    - type: INTEGER
                      name: columns
                      value: '3'
                    - type: GRAPH_PROTOTYPE
                      name: graphid.0
                      value:
                        host: 'Linux by Zabbix agent'
                        name: 'FS [{#FSTYPE}({#FSNAME})]: Space utilization chart (relative to total)'
                    - type: STRING
                      name: reference
                      value: AAAEB
                - type: graphprototype
                  'y': '5'
                  width: '72'
                  height: '5'
                  fields:
                    - type: INTEGER
                      name: columns
                      value: '1'
                    - type: GRAPH_PROTOTYPE
                      name: graphid.0
                      value:
```

```
                        host: 'Linux by Zabbix agent'
                        name: 'FS [{#FSTYPE}({#FSNAME})]: Space usage graph, in % (relative to max availab
                    - type: STRING
                      name: reference
                      value: AAAEC
        (...)
```

Element	Type	Description
type	string	(required) Tipo de widget.
name	string	Nombre del widget.
x	integer	Posición horizontal desde el lado izquierdo del dashboard de template. Valores posibles:[1] 0-71.
y	integer	Posición vertical desde la parte superior del dashboard de template. Valores posibles:[1] 0-63.
width	integer	Ancho del widget. Valores posibles:[1] 1-72.
height	integer	Altura del widget. Valores posibles:[1] 1-64.
hide_header	string	Ocultar el encabezado del widget. Valores posibles:[1] NO (0, predeterminado), YES (1).
fields		Elemento raíz para los campos del widget del dashboard de template.
type	string	(required) Tipo de campo del widget. Valores posibles:[1] INTEGER (0), STRING (1), ITEM (4), ITEM_PROTOTYPE (5), GRAPH (6), GRAPH_PROTOTYPE (7), MAP (8), SERVICE (9), SLA (10), USER (11), ACTION (12), MEDIA_TYPE (13).
name	string	(required) Nombre del campo del widget. Valores posibles:[1]
value	mixed	(required) Valor del campo del widget, según el tipo de campo.

Note:
Véase también: Objeto de widget del dashboard de template (consulte la propiedad correspondiente con un nombre coincidente).

Mapas de valores de template

```
zabbix_export:
  (...)
  templates:
    (...)
      valuemaps:
        - uuid: 8c048c6cca8248f2860c208e8db0f59e
          name: ifOperStatus
          mappings:
            - value: '0'
              newvalue: unknown
            - value: '1'
              newvalue: notpresent
            - value: '2'
              newvalue: down
            - value: '3'
              newvalue: lowerlayerdown
            - value: '4'
              newvalue: testing
            - value: '5'
              newvalue: dormant
            - value: '6'
              newvalue: up
        (...)
```

Element	Type	Description
uuid	string	(obligatorio) Identificador único para este mapa de valores.
name	string	(obligatorio) Nombre del mapa de valores.
mapping		Elemento raíz para las asignaciones.
type	string	Tipo de coincidencia de la asignación. Valores posibles:[1] EQUAL (0, predeterminado), GREATER_OR_EQUAL (2), LESS_OR_EQUAL (3), IN_RANGE (4), REGEXP (5), DEFAULT (6).
value	string	Valor original.
newvalue	string	(obligatorio) Valor al que se asigna el valor original.

Note:
Véase también: Objeto de mapa de valores (consulte la propiedad correspondiente con un nombre coincidente).

Notas al pie [1] Los valores enteros de la API entre paréntesis, por ejemplo, ENABLED (0), se mencionan solo como referencia. Para obtener más información, consulte la página del objeto de API enlazada en la entrada de la tabla o al final de cada sección.

Hosts

Resumen

Los hosts se exportan con muchos objetos relacionados y relaciones entre objetos.

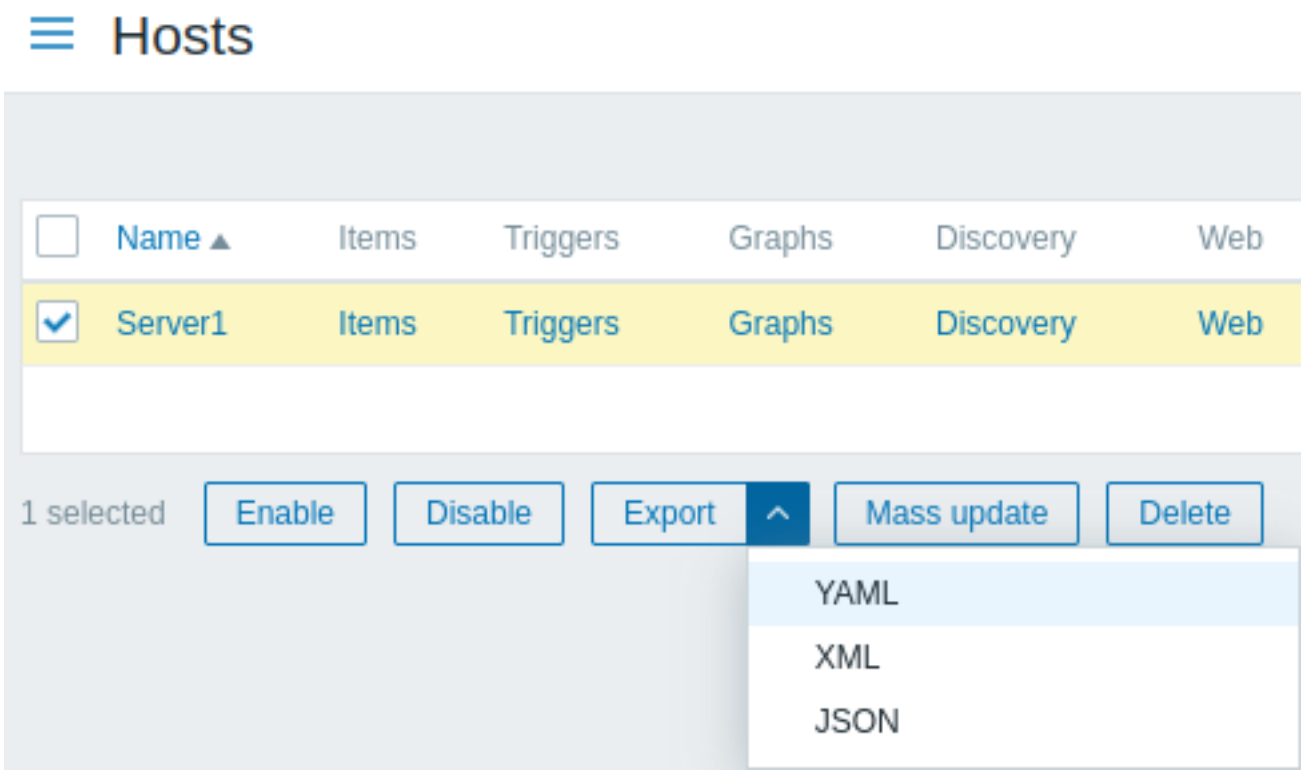
La exportación de host contiene:

- Grupos de host vinculados
- Datos del host
- Vinculación de template
- Vinculación de grupo de host
- Interfaces del host
- Items vinculados directamente
- Triggers vinculados directamente
- Reglas de descubrimiento vinculadas directamente con todos los prototipos
- Escenarios web vinculados directamente
- Macros del host
- Datos de inventario del host
- Mapas de valores
- Gráficos vinculados

Exportación

Para exportar hosts, haga lo siguiente:

1. Vaya a *Data collection* > *Hosts*.
2. Marque las casillas de verificación de los hosts que desea exportar.
3. Haga clic en *Export* debajo de la lista para exportar en el formato YAML predeterminado, o haga clic en la flecha para seleccionar un formato de exportación diferente.



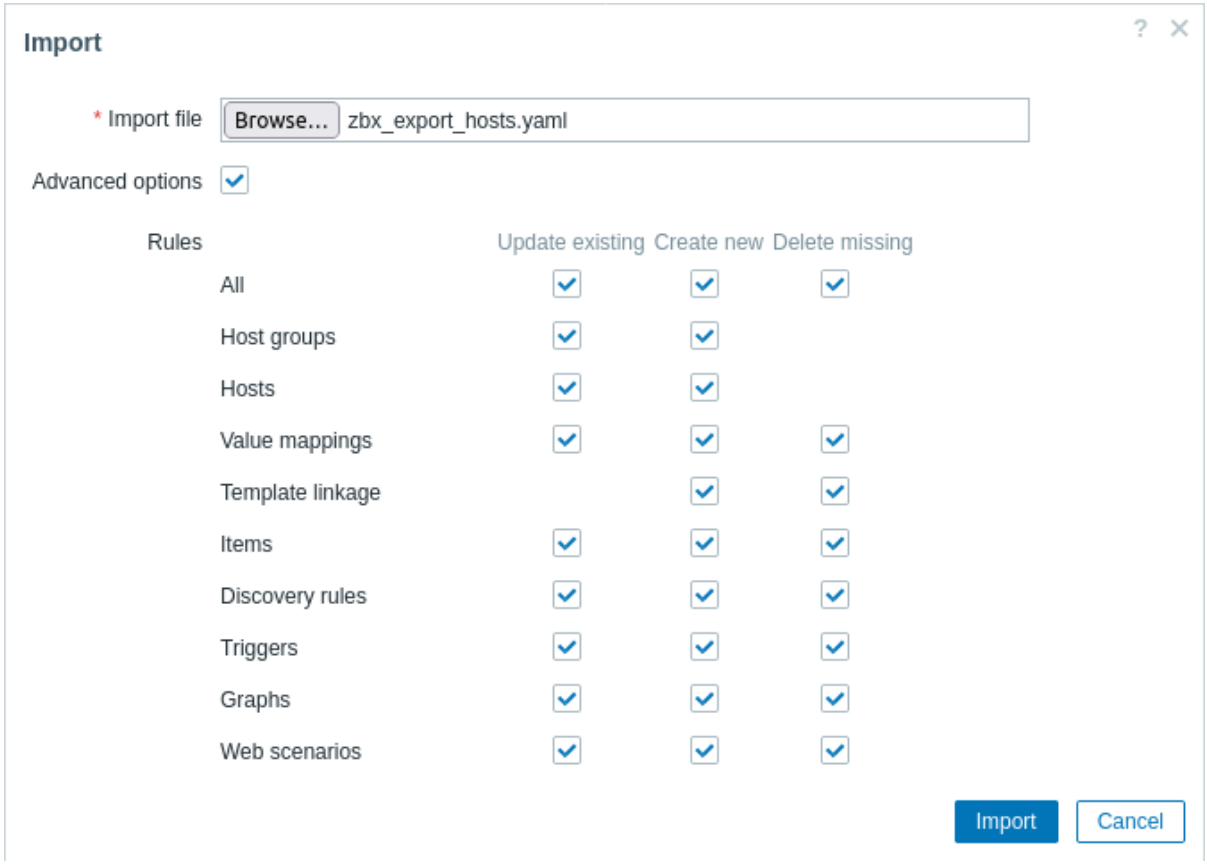
Según el formato seleccionado, los hosts se exportan a un archivo local con un nombre predeterminado:

- `zabbix_export_hosts.yaml` - en la exportación YAML;
- `zabbix_export_hosts.xml` - en la exportación XML;
- `zabbix_export_hosts.json` - en la exportación JSON.

Importación

Para importar hosts, siga estos pasos:

1. Vaya a *Recopilación de datos* > *Hosts*.
2. Haga clic en *Importar* en la esquina superior derecha.
3. Seleccione el archivo de importación.
4. Marque las casillas de las reglas de importación que desea aplicar (consulte más abajo).
5. Haga clic en *Importar* en la esquina inferior derecha del formulario de configuración.



Si marca la casilla *Opciones avanzadas*, se mostrará una lista detallada de todos los elementos que se pueden importar; marque o desmarque cada regla de importación según sea necesario.

Si hace clic en la casilla de la fila *Todos*, se marcarán o desmarcarán todos los elementos que aparecen debajo.

Reglas de importación:

Regla	Descripción
Actualizar existentes	Los elementos existentes se actualizarán utilizando los datos del archivo de importación. De lo contrario, no se actualizarán.
Crear nuevos	Los elementos nuevos se crearán utilizando los datos del archivo de importación. De lo contrario, no se crearán.
Eliminar faltantes	Los elementos existentes que no estén presentes en el archivo de importación se eliminarán. De lo contrario, no se eliminarán. Si se marca *Eliminar faltantes* para *Vinculación de templates*, se desvinculará la vinculación actual de templates que no esté presente en el archivo de importación. Las entidades (items, triggers, gráficos, etc.) heredadas de los templates desvinculados no se eliminarán (a menos que también se seleccione la opción *Eliminar faltantes* para cada entidad).

No puede revisar los cambios realizados en los hosts durante la importación. En el frontend se mostrará un mensaje de éxito o de error de la importación.

Formato de exportación

Cuando se exporta un host, Zabbix genera un formato estructurado YAML, JSON o XML. La exportación incluye elementos del host, como metadatos del host, items, macros, triggers y más.

Cada elemento cumple una función específica y puede contener elementos anidados.

Las siguientes secciones describen cada elemento del formato de exportación. Los ejemplos usan un host con la template MySQL by Zabbix agent 2. Con fines de demostración de los elementos, la template se desvincula después de crear el host. Desvincular conserva todos los elementos en la configuración (a diferencia de desvincular y limpiar, que los elimina).

Una elipsis (`...`) indica elementos omitidos por brevedad. La nota (`see table below`) se usa cuando el elemento se explica con más detalle en secciones posteriores.

```yaml
zabbix_export:
  version: '8.0'
  host_groups:
    - uuid: 748ad4d098d447d492bb935c907f652f
      name: Databases
  hosts: (see table below)
  graphs: (see table below)
```

Metadatos del host

Element		Type	Description
version		string	(required) Versión de Zabbix.
host_groups			Elemento raíz para los grupos de host.
	uuid	string	(required) Identificador único para este grupo de host.
	name	string	(required) Nombre del grupo de host.
hosts			Elemento raíz para hosts.
graphs			Elemento raíz para gráficos de host.

Hosts

```yaml
zabbix_export:
  (...)
  hosts:
    - host: 'MySQL server'
      name: 'MySQL server'
      groups:
        - name: Databases
      interfaces: (see table below)
      items: (see table below)
      discovery_rules: (see table below)
      tags:
        - tag: location
          value: Riga
      macros:
        (...)
        - macro: '{$MYSQL.DSN}'
          value: 192.0.2.1
        (...)
        - macro: '{$MYSQL.PASSWORD}'
          type: SECRET_TEXT
          description: 'MySQL user password.'
        (...)
      valuemaps: (see table below)
  (...)
```

Element	Type	Description
host	string	(required) Unique host name.
name	string	Visible host name.
description	text	Descripción del host.
monitored_by	string	Cómo se monitoriza el host. Valores posibles:[1] SERVER (0, predeterminado), PROXY (1) o PROXY_GROUP (2).
proxy		Elemento raíz para proxy.
name	string	(required) Nombre del proxy (si existe) que monitoriza el host.
proxy_group		Elemento raíz para el grupo de proxy.
name	string	(required) Nombre del grupo de proxy (si existe) que se utiliza para monitorizar el host.
status	string	Estado del host. Valores posibles:[1] ENABLED (0, predeterminado), DISABLED (1).
ipmi_authtype	string	Tipo de autenticación de la sesión IPMI. Valores posibles:[1] DEFAULT (-1, predeterminado), NONE (0), MD2 (1), MD5 (2), STRAIGHT (4), OEM (5), RMCP_PLUS (6).

Element	Type	Description
ipmi_privilege	string	Nivel de privilegio de la sesión IPMI. Valores posibles:[1] CALLBACK (1), USER (2, predeterminado), OPERATOR (3), ADMIN (4), OEM (5).
ipmi_username	string	Nombre de usuario para las comprobaciones IPMI.
ipmi_password	string	Contraseña para las comprobaciones IPMI.
templates		Elemento raíz para las templates vinculadas.
name	string	(required) Nombre de la template.
groups		Elemento raíz para los grupos de host a los que pertenece el host.
name	string	(required) Nombre del grupo de host.
interfaces		Elemento raíz para las interfaces del host.
items		Elemento raíz para los items del host.
discovery_rules		Elemento raíz para las reglas de descubrimiento de bajo nivel del host.
http_tests		Elemento raíz para los escenarios web del host.
tags		Elemento raíz para las etiquetas del host.
tag	string	(required) Nombre de la etiqueta.
value	string	Valor de la etiqueta.
macros		Elemento raíz para las macros del host.
macro	string	(required) Nombre de la macro de usuario.
type	string	Tipo de macro de usuario. Valores posibles:[1] TEXT (0, predeterminado), SECRET_TEXT (1), VAULT (2).
value	string	Valor de la macro de usuario.
description	string	Descripción de la macro de usuario.
inventory		Elemento raíz para el inventario del host.
<inventory_property>	string	Propiedad del inventario. Todas las propiedades tienen su respectivo elemento (`type`, `name`, `os`, etc.; por ejemplo, consulte Export format).
inventory_mode	string	Modo de inventario. Valores posibles:[1] DISABLED (-1), MANUAL (0, predeterminado), AUTOMATIC (1).
valuemaps		Elemento raíz para los value maps del host.

Note:
Véase también: Host object (consulte la propiedad correspondiente con el mismo nombre).

Interfaces del host

```
zabbix_export:
  (...)
  hosts:
    - host: 'MySQL server'
      (...)
      interfaces:
        - ip: 192.0.2.1
          interface_ref: if1
      (...)
```

Element	Type	Description
default	string	Indica si esta es la interfaz principal del host. Tenga en cuenta que solo puede haber una interfaz principal de un tipo en un host. Valores posibles:[1] NO (0), YES (1, default).
type	string	Tipo de interfaz. Valores posibles:[1] ZABBIX (1, default), SNMP (2), IPMI (3), JMX (4).
useip	string	Indica si se debe usar IP como interfaz para conectarse al host (de lo contrario, se usará DNS). Valores posibles:[1] NO (0), YES (1, default).
ip	string	(required for IP connections) Dirección IP (IPv4 o IPv6).
dns	string	(required for DNS connections) Nombre DNS.
port	string	Número de puerto.
details		Elemento raíz para los detalles de la interfaz.
version	string	Use esta versión de SNMP. Valores posibles:[1] SNMPV1 (1), SNMP_V2C (2, default), SNMP_V3 (3).
community	string	(required for SNMPv1 and SNMPv2 items) Comunidad SNMP.

Element	Type	Description
max_repetitions	string	Valor máximo de repeticiones para solicitudes masivas nativas de SNMP (GetBulkRequest-PDUs). Compatible con items SNMPv2 y SNMPv3 (`discovery[]` y `walk[]`). Predeterminado: 10.
contextname	string	Nombre de contexto de SNMPv3. Compatible con items SNMPv3.
securityname	string	Nombre de seguridad de SNMPv3. Compatible con items SNMPv3.
securitylevel	string	Nivel de seguridad de SNMPv3. Compatible con items SNMPv3. Valores posibles:[1] NOAUTHNOPRIV (0, default), AUTHNOPRIV (1), AUTHPRIV (2).
authprotocol	string	Protocolo de autenticación de SNMPv3. Compatible con items SNMPv3. Valores posibles:[1] MD5 (0, default), SHA1 (1), SHA224 (2), SHA256 (3), SHA384 (4), SHA512 (5).
authpassphrase	string	Frase de contraseña de autenticación de SNMPv3. Compatible con items SNMPv3.
privprotocol	string	Protocolo de privacidad de SNMPv3. Compatible con items SNMPv3. Valores posibles:[1] DES (0, default), AES128 (1), AES192 (2), AES256 (3), AES192C (4), AES256C (5).
privpassphrase	string	Frase de contraseña de privacidad de SNMPv3. Compatible con items SNMPv3.
bulk	string	Usar solicitudes masivas para SNMP. Valores posibles:[1] NO (0), YES (1, default).
interface_ref	string	Nombre de referencia de la interfaz que se usará en los items (formato: `if<N>`).

Note:
Véase también: Objeto de interfaz del host (consulte la propiedad correspondiente con el mismo nombre).

Items del host

```
zabbix_export:
  (...)
  hosts:
    - host: 'MySQL server'
      (...)
      items:
        (...)
        - name: 'Uso de disco de la caché de binlog'
          type: DEPENDENT
          key: mysql.binlog_cache_disk_use
          value_type: FLOAT
          description: 'Número de transacciones que utilizaron una caché de disco temporal porque no cabía
          preprocessing: (consulte la tabla siguiente)
          master_item:
            key: 'mysql.get_status_variables["{$MYSQL.DSN}","{$MYSQL.USER}","{$MYSQL.PASSWORD}"]'
          tags:
            - tag: component
              value: cache
        (...)
        - name: 'Utilización del pool de búferes'
          type: CALCULATED
          key: mysql.buffer_pool_utilization
          value_type: FLOAT
          units: '%'
          params: |
            ( last(//mysql.innodb_buffer_pool_pages_total) -
            last(//mysql.innodb_buffer_pool_pages_free) ) /
            ( last(//mysql.innodb_buffer_pool_pages_total) +
            ( last(//mysql.innodb_buffer_pool_pages_total) = 0 ) ) * 100 *
```

```
        ( last(//mysql.innodb_buffer_pool_pages_total) > 0 )
      description: 'Proporción de páginas utilizadas respecto al total de páginas del pool de búferes.
      tags:
        - tag: component
          value: memory
      triggers: (consulte la tabla siguiente)
    (...)
    - name: Tiempo de actividad
      type: DEPENDENT
      key: mysql.uptime
      units: uptime
      description: 'Número de segundos que el server ha estado activo.'
      preprocessing: (consulte la tabla siguiente)
      master_item:
        key: 'mysql.get_status_variables["{$MYSQL.DSN}","{$MYSQL.USER}","{$MYSQL.PASSWORD}"]'
      tags:
        - tag: component
          value: application
      triggers: (consulte la tabla siguiente)
    (...)
```

Elemento	Tipo	Descripción
name	string	(obligatorio) Nombre del item.
type	string	Tipo de item. Valores posibles:[1] ZABBIX_PASSIVE (0, predeterminado), TRAP (2), SIMPLE (3), INTERNAL (5), ZABBIX_ACTIVE (7), EXTERNAL (10), ODBC (11), IPMI (12), SSH (13), TELNET (14), CALCULATED (15), JMX (16), SNMP_TRAP (17), DEPENDENT (18), HTTP_AGENT (19), SNMP_AGENT (20), ITEM_TYPE_SCRIPT (21), ITEM_TYPE_BROWSER (22), ITEM_TYPE_NESTED (23), TELEMETRY_QUERY (24).
snmp_oid	string	(obligatorio para items SNMP_AGENT) ID de objeto SNMP.
key	string	(obligatorio) Clave del item.
delay	string	Intervalo de actualización del item. Predeterminado: 1m. El valor siempre será 0 para los items TRAP.
history	string	Período de tiempo (mediante sufijo de tiempo, macro de usuario o macro de LLD) durante el cual se deben conservar los datos históricos. Predeterminado: 31d.
trends	string	Período de tiempo (mediante sufijo de tiempo, macro de usuario o macro de LLD) durante el cual se deben conservar los datos de tendencias. Predeterminado: 365d.
status	string	Estado del item. Valores posibles:[1] ENABLED (0, predeterminado), DISABLED (1).
value_type	string	Tipo de valor recibido. Valores posibles:[1] FLOAT (0), CHAR (1), LOG (2), UNSIGNED (3, predeterminado), TEXT (4), BINARY (5), JSON (6).
allowed_hosts	string	Lista de direcciones IP, separadas por comas, de los hosts autorizados a enviar datos para el item. Compatible con items TRAP y HTTP_AGENT.
units	string	Unidades del valor recibido (bps, B, etc.).
params	text	Parámetros adicionales según el tipo de item (script ejecutado para items SSH y TELNET; consulta SQL para items ODBC; fórmula para items CALCULATED; script para items ITEM_TYPE_SCRIPT y ITEM_TYPE_BROWSER).
ipmi_sensor	string	Sensor IPMI. Compatible con items IPMI.
authtype	string	Tipo de autenticación. Compatible con items SSH y HTTP_AGENT. Valores posibles para items SSH:[1] PASSWORD (0, predeterminado), PUBLIC_KEY (1). Valores posibles para items HTTP_AGENT:[1] NONE (0, predeterminado), BASIC (1), NTLM (2), Kerberos (3) o Digest (4).
username	string	(obligatorio para items SSH y TELNET) Nombre de usuario para la autenticación. Compatible con items SIMPLE, ODBC, JMX y HTTP_AGENT. Cuando se utiliza para items JMX, también se debe especificar `password` (consulte más abajo), o ambos elementos deben dejarse en blanco.

Elemento	Tipo	Descripción
password	string	(obligatorio para items SSH y TELNET) Contraseña para la autenticación. Compatible con items SIMPLE, ODBC, JMX y HTTP_AGENT. Cuando se utiliza para items JMX, también se debe especificar `username` (consulte más arriba), o ambos elementos deben dejarse en blanco.
publickey	string	(obligatorio para items SSH) Nombre del archivo de clave pública.
privatekey	string	(obligatorio para items SSH) Nombre del archivo de clave privada.
description	text	Descripción del item.
inventory_link	string	Campo del inventario del host que se completa mediante el item. Valores posibles:[1] NONE (0), ALIAS (4), etc. (consulte Inventario del host para ver los campos compatibles).
valuemap		Elemento raíz para los mapas de valores del item.
name	string	(obligatorio) Nombre del mapa de valores que se utilizará para el item.
logtimefmt	string	Formato de la hora en las entradas del registro. Compatible con items de tipo de valor LOG.
preprocessing		Elemento raíz para el preprocesamiento de valores del item.
step		Elemento raíz para los pasos de preprocesamiento de valores de items del host.
interface_ref	string	Referencia a la interfaz del host (formato: `if<N>`).
jmx_endpoint	string	Endpoint JMX. Compatible con items JMX.
master_item		(obligatorio para items DEPENDENT) Elemento raíz para el item maestro del item dependiente.
key	string	(obligatorio) Clave del item maestro del item dependiente.
timeout	string	Tiempo de espera de la solicitud de sondeo de datos del item. Compatible con la lista de tipos de items de Tiempos de espera.
url	string	(obligatorio para items HTTP_AGENT) Cadena de URL.
query_fields		Elemento raíz para los parámetros de consulta. Compatible con items HTTP_AGENT.
name	string	(obligatorio para items HTTP_AGENT) Nombre del parámetro de consulta.
value	string	Valor del parámetro de consulta. Compatible con items HTTP_AGENT.
parameters		Elemento raíz para los parámetros definidos por el usuario. Compatible con items ITEM_TYPE_SCRIPT y ITEM_TYPE_BROWSER.
name	string	(obligatorio para items ITEM_TYPE_SCRIPT y ITEM_TYPE_BROWSER) Nombre del parámetro definido por el usuario.
value	string	Valor del parámetro definido por el usuario. Compatible con items ITEM_TYPE_SCRIPT y ITEM_TYPE_BROWSER.
posts	string	Datos del cuerpo de la solicitud HTTP(S). Compatible con items HTTP_AGENT.
status_codes	string	Rangos de códigos de estado HTTP requeridos, separados por comas. Compatible con items HTTP_AGENT.
follow_redirects	string	Seguir las redirecciones de respuesta durante el sondeo de datos. Compatible con items HTTP_AGENT. Valores posibles:[1] NO (0), YES (1, predeterminado).
post_type	string	Tipo de cuerpo de los datos enviados. Compatible con items HTTP_AGENT. Valores posibles:[1] RAW (0, predeterminado), JSON (2), XML (3).
http_proxy	string	Cadena de conexión del proxy HTTP(S). Compatible con items HTTP_AGENT.
headers		Elemento raíz para los encabezados de la solicitud HTTP(S). Compatible con items HTTP_AGENT.
name	string	(obligatorio para items HTTP_AGENT) Nombre del encabezado.
value	string	(obligatorio para items HTTP_AGENT) Valor del encabezado.
retrieve_mode	string	Parte de la respuesta que se debe almacenar. Compatible con items HTTP_AGENT. Valores posibles:[1] BODY (0, predeterminado), HEADERS (1), BOTH (2).
request_method	string	Tipo de método de solicitud. Compatible con items HTTP_AGENT. Valores posibles:[1] GET (0, predeterminado), POST (1), PUT (2), HEAD (3).
output_format	string	Cómo procesar la respuesta. Compatible con items HTTP_AGENT. Valores posibles:[1] RAW (0, predeterminado), JSON (1).

Elemento	Tipo	Descripción
allow_traps	string	Permitir que el valor se complete de la misma forma que para un item trapper. Compatible con items HTTP_AGENT. Valores posibles:[1] NO (0, predeterminado), YES (1).
ssl_cert_file	string	Ruta del archivo de clave pública SSL. Compatible con items HTTP_AGENT.
ssl_key_file	string	Ruta del archivo de clave privada SSL. Compatible con items HTTP_AGENT.
ssl_key_password	string	Contraseña del archivo de clave SSL. Compatible con items HTTP_AGENT.
verify_peer	string	Indica si se debe validar que el certificado del host sea auténtico. Compatible con items HTTP_AGENT. Valores posibles:[1] NO (0, predeterminado), YES (1).
verify_host	string	Indica si se debe validar que el nombre del host de la conexión coincida con el del certificado del host. Compatible con items HTTP_AGENT. Valores posibles:[1] NO (0, predeterminado), YES (1).
time_shift	string	Hasta qué momento anterior a la hora actual termina el intervalo de tiempo que consulta el item. Compatible con items TELEMETRY_QUERY.
lookback_limit	string	Hasta qué momento anterior a la hora actual consulta el item los datos cuando no existe una consulta anterior que continuar (por ejemplo, durante la primera ejecución del item), o cuando la consulta anterior es más antigua que `delay` (por ejemplo, porque el server de Zabbix estaba inactivo). Compatible con items TELEMETRY_QUERY.
granularity	string	Duración de cada segmento en el que se divide el intervalo de tiempo completo de la consulta. Compatible con items TELEMETRY_QUERY.
query		(obligatorio para items TELEMETRY_QUERY) Elemento raíz para la consulta de telemetría. Define qué consulta el item en el origen de datos, cómo filtra los datos y cómo agrega el resultado.
signal_type	string	(obligatorio) Tipo de señal que consulta el item. Compatible con items TELEMETRY_QUERY. Valores posibles:[1] TRACES (0), METRICS (1), LOGS (2).
metric_point_type	string	(obligatorio para el tipo de señal METRICS) Tipo de punto de métrica que consulta el item. Compatible con items TELEMETRY_QUERY. Valores posibles:[1] SUM (0), GAUGE (1), HISTOGRAM (2), EXPONENTIAL_HISTOGRAM (3).
columns		(obligatorio) Elemento raíz para las columnas de datos de telemetría que consulta el item. Este elemento contiene los siguientes elementos: `column (string)` - (obligatorio) Nombre de la columna de datos de telemetría que consulta el item. `attribute_key (string)` - (obligatorio para columnas que contienen pares clave-valor) Nombre de la clave que consulta el item dentro de la columna que contiene pares clave-valor.
aggregated_columns		(obligatorio) Elemento raíz para las columnas agregadas de datos de telemetría que consulta el item. Este elemento contiene los siguientes elementos: `column (string)` - (obligatorio para las funciones MIN, MAX, AVG, SUM y PERCENTILE) Nombre de la columna de datos de telemetría que consulta el item. `function (string)` - (obligatorio) Función que el item aplica a los registros de la columna. Valores posibles:[1] MIN (1), MAX (2), AVG (3), COUNT (4), SUM (5), PERCENTILE (8). `parameters (array)` - (obligatorio para las funciones PERCENTILE) Percentil que el item calcula a partir de los registros de la columna. `alias (string)` - (obligatorio) Nombre único del valor calculado, utilizado para identificarlo en el valor del item.

Elemento	Tipo	Descripción
filter		(obligatorio) Elemento raíz para el filtro de consulta de telemetría aplicado a los registros de columnas de la consulta. Este elemento contiene los siguientes elementos: `evaltype` (`string`) - (obligatorio) Método de evaluación de las condiciones del filtro. Valores posibles:[1] AND_OR (0), AND (1), OR (2), FORMULA (3). `formula` (`string`) - (obligatorio para los métodos de evaluación FORMULA) Expresión definida por el usuario que se utilizará para evaluar las condiciones de los filtros con una expresión personalizada. `conditions` (`string`) - (obligatorio) Elemento raíz para las condiciones de filtro de datos de telemetría que debe cumplir un registro de una columna de datos de telemetría. El elemento `conditions` contiene los siguientes elementos: `column` (`string`) - (obligatorio) Nombre de la columna de datos de telemetría a la que se aplican las condiciones del filtro. `attribute_key` (`string`) - (obligatorio para columnas que contienen pares clave-valor) Nombre de la clave a la que se aplica la condición dentro de la columna que contiene pares clave-valor. `operator` (`string`) - (obligatorio) Operador de la condición del filtro. Valores posibles para columnas de un solo valor:[1] EQUAL (0), NOT_EQUAL (1), LIKE (2), NOT_LIKE (3). Valores posibles para columnas que contienen pares clave-valor:[1] EQUAL (0), NOT_EQUAL (1), EXISTS (12). `formulaid` (`string`) - (obligatorio para los métodos de evaluación FORMULA) ID único arbitrario utilizado para hacer referencia a la condición desde una expresión personalizada. `value` (`string`) - (obligatorio para las operaciones EQUAL, NOT_EQUAL, LIKE y NOT LIKE) Valor con el que se debe comparar.
tags		Elemento raíz para las etiquetas del item.
tag	string	(obligatorio) Nombre de la etiqueta.
value	string	Valor de la etiqueta.
triggers		Elemento raíz para los triggers de items del host.

Note:
Véase también: Objeto item (consulte la propiedad correspondiente con un nombre coincidente).

Pasos de preprocesamiento del valor del item del host

```
zabbix_export:
  (...)
  hosts:
    - host: 'MySQL server'
      (...)
      items:
        (...)
        - name: 'Binlog cache disk use'
          (...)
          preprocessing:
            - type: JSONPATH
              parameters:
                - $.Binlog_cache_disk_use
            - type: DISCARD_UNCHANGED_HEARTBEAT
              parameters:
                - 6h
          (...)
```

Element	Type	Description
type	string	(required) El tipo de paso de preprocesamiento del valor del item. Valores posibles:[1] MULTIPLIER (1), RTRIM (2), LTRIM (3), TRIM (4), REGEX (5), BOOL_TO_DECIMAL (6), OCTAL_TO_DECIMAL (7), HEX_TO_DECIMAL (8), SIMPLE_CHANGE (9, calculated: received value - previous value), CHANGE_PER_SECOND (10, calculated: (received value - previous value)/(time now - time of last check)), XMLPATH (11), JSONPATH (12), IN_RANGE (13), MATCHES_REGEX (14), NOT_MATCHES_REGEX (15), CHECK_JSON_ERROR (16), CHECK_XML_ERROR (17), CHECK_REGEX_ERROR (18), DISCARD_UNCHANGED (19), DISCARD_UNCHANGED_HEARTBEAT (20), JAVASCRIPT (21), PROMETHEUS_PATTERN (22), PROMETHEUS_TO_JSON (23), CSV_TO_JSON (24), STR_REPLACE (25), CHECK_NOT_SUPPORTED (26), XML_TO_JSON (27), SNMP_WALK_VALUE (28), SNMP_WALK_TO_JSON (29), SNMP_GET_VALUE (30).
parameters		(required) Elemento raíz para los parámetros del paso de preprocesamiento del valor del item.
parameter	string	Parámetro individual del paso de preprocesamiento del valor del item.
error_handler	string	Tipo de acción usado en caso de fallo del paso de preprocesamiento. Valores posibles:[1] ORIGINAL_ERROR (0, default), DISCARD_VALUE (1), CUSTOM_VALUE (2), CUSTOM_ERROR (3).
error_handler_params	string	Parámetros del controlador de errores.

Note:
Véase también: Objeto de preprocesamiento de item (consulte la propiedad correspondiente con el mismo nombre).

Triggers de item del host

```
zabbix_export:
  (...)
  hosts:
    - host: 'MySQL server'
      (...)
      items:
        (...)
        - name: Uptime
          (...)
          triggers:
            - expression: 'nodata(/MySQL server/mysql.uptime,30m)=1'
              name: 'MySQL: Failed to fetch info data'
              event_name: 'MySQL: Failed to fetch info data (or no data for 30m)'
              priority: INFO
              description: 'Zabbix no ha recibido ningún dato para los items durante los últimos 30 minuto
              dependencies:
                - name: 'MySQL: Service is down'
                  expression: 'last(/MySQL server/mysql.ping["{$MYSQL.DSN}","{$MYSQL.USER}","{$MYSQL.PASSW
              tags:
                - tag: scope
                  value: availability
            - expression: 'last(/MySQL server/mysql.uptime)<10m'
              name: 'MySQL: Service has been restarted'
              event_name: 'MySQL: Service has been restarted (uptime < 10m)'
              priority: INFO
              description: 'El tiempo de actividad de MySQL es inferior a 10 minutos.'
              tags:
                - tag: scope
                  value: notice
        (...)
```

Element	Type	Description
uuid	string	(required) Identificador único de este trigger.
expression	string	(required) Expresión del trigger.

Element	Type	Description
recovery_mode	string	Base para generar eventos OK. Valores posibles:[1] EXPRESSION (0, predeterminado), RECOVERY_EXPRESSION (1), NONE (2).
recovery_expression	string	Expresión de recuperación del trigger.
correlation_mode	string	Modo de correlación (sin correlación de eventos o correlación de eventos por etiqueta). Valores posibles:[1] DISABLED (0, predeterminado), TAG_VALUE (1).
correlation_tag	string	Nombre de la etiqueta que se usará para la correlación de eventos.
name	string	(required) Nombre del trigger.
event_name	string	Nombre del evento.
opdata	string	Datos operativos.
url_name	string	Etiqueta para la URL asociada al trigger.
url	string	URL asociada al trigger.
status	string	Estado del trigger. Valores posibles:[1] ENABLED (0, predeterminado), DISABLED (1).
priority	string	Severidad del trigger. Valores posibles:[1] NOT_CLASSIFIED (0, predeterminado), INFO (1), WARNING (2), AVERAGE (3), HIGH (4), DISASTER (5).
description	text	Descripción del trigger.
type	string	Tipo de generación de eventos (un único evento de problema o varios eventos de problema). Valores posibles:[1] SINGLE (0, predeterminado), MULTIPLE (1).
manual_close	string	Cierre manual de eventos de problema. Valores posibles:[1] NO (0, predeterminado), YES (1).
dependencies		Elemento raíz para las dependencias.
name	string	(required) Nombre del trigger dependiente.
expression	string	(required) Expresión del trigger dependiente.
recovery_expression	string	Expresión de recuperación del trigger dependiente.
tags		Elemento raíz para las etiquetas del trigger.
tag	string	(required) Nombre de la etiqueta.
value	string	Valor de la etiqueta.

Note:
Véase también: Objeto Trigger (consulte la propiedad correspondiente con el mismo nombre).

Reglas de descubrimiento de bajo nivel de host

```
zabbix_export:
  (...)
  hosts:
    - host: 'MySQL server'
      (...)
      discovery_rules:
        - name: 'Database discovery'
          key: 'mysql.db.discovery["{$MYSQL.DSN}","{$MYSQL.USER}","{$MYSQL.PASSWORD}"]'
          delay: 1h
          filter: (see table below)
          description: 'Scanning databases in DBMS.'
          interface_ref: if1
          item_prototypes:
            - name: 'Size of database {#DATABASE}'
              key: 'mysql.db.size["{$MYSQL.DSN}","{$MYSQL.USER}","{$MYSQL.PASSWORD}","{#DATABASE}"]'
              delay: 5m
              units: B
              description: 'Database size.'
              preprocessing:
                - type: DISCARD_UNCHANGED_HEARTBEAT
                  parameters:
                    - 1h
              interface_ref: if1
              tags:
                - tag: component
```

```
                value: storage
              - tag: database
                value: '{#DATABASE}'
        lld_macro_paths:
          - lld_macro: '{#DATABASE}'
            path: $.Database
        preprocessing:
          - type: DISCARD_UNCHANGED_HEARTBEAT
            parameters:
              - 1d
    (...)
```

Attention:
La mayoría de los elementos de las reglas de descubrimiento de bajo nivel de host son los mismos que para los items de host. La tabla siguiente describe aquellos elementos que difieren de los items de host.

Element	Type	Description
type	string	Tipo de item. Valores posibles:[1] ZABBIX_PASSIVE (0, predeterminado), TRAP (2), SIMPLE (3), INTERNAL (5), ZABBIX_ACTIVE (7), EXTERNAL (10), ODBC (11), IPMI (12), SSH (13), TELNET (14), JMX (16), DEPENDENT (18), HTTP_AGENT (19), SNMP_AGENT (20), ITEM_TYPE_SCRIPT (21), ITEM_TYPE_BROWSER (22).
key	string	(obligatorio) La clave de la regla de descubrimiento de bajo nivel.
filter		Elemento raíz para los filtros de reglas de descubrimiento de bajo nivel de host.
lifetime	string	Período de tiempo (usando segundos, sufijo de tiempo o macro de usuario), tras el cual se eliminarán los recursos descubiertos que ya no existan. Predeterminado: 7d.
lifetime_type	string	Escenario para eliminar recursos LLD perdidos. Valores posibles: DELETE_NEVER, DELETE_IMMEDIATELY, DELETE_AFTER.
enabled_lifetime	string	Período de tiempo (usando segundos, sufijo de tiempo o macro de usuario), tras el cual se deshabilitarán los recursos descubiertos que ya no existan.
enabled_lifetime_type	string	Escenario para deshabilitar recursos LLD perdidos. Valores posibles: DISABLE_NEVER, DISABLE_IMMEDIATELY, DISABLE_AFTER.
item_prototypes		Elemento raíz para los elementos prototipo de item de host, que son los mismos que para los items de host.
trigger_prototypes		Elemento raíz para los elementos prototipo de trigger de host, que son los mismos que para los triggers de item de host.
graph_prototypes		Elemento raíz para los prototipos de gráfico de host, que son los mismos que para los gráficos de host.
host_prototypes		Elemento raíz para los prototipos de host, que son los mismos que para los hosts.
parent_discovery_rule		Elemento raíz para la regla de descubrimiento de bajo nivel principal (o prototipo de regla) del prototipo de regla de descubrimiento de bajo nivel. Esta propiedad indica que se trata de un prototipo de regla LLD, hijo directo de la regla/prototipo de regla referenciado.
key	string	(obligatorio) La clave de la regla de descubrimiento de bajo nivel principal (o prototipo de regla).
master_item	string	(obligatorio para reglas DEPENDENT) Elemento raíz para el item maestro de la regla dependiente.
lld_macro_paths		Elemento raíz para las rutas de macros de la regla de descubrimiento de bajo nivel.
lld_macro	string	(obligatorio) Nombre de la macro de la regla de descubrimiento de bajo nivel.
path	string	(obligatorio) Selector del valor que se asignará a la macro correspondiente.
preprocessing		Elemento raíz para el preprocesamiento del valor de la regla de descubrimiento de bajo nivel.
step		Elemento raíz para los elementos de paso de preprocesamiento del valor de la regla de descubrimiento de bajo nivel, que son los mismos que para los pasos de preprocesamiento del valor de item de host, excepto que tienen menos valores posibles. Véase también: objeto de preprocesamiento de la regla LLD.
overrides		Elemento raíz para las reglas de anulación de la regla de descubrimiento de bajo nivel.
name	string	(obligatorio) Nombre único de la anulación.
step	string	(obligatorio) Número de orden único de la anulación.
stop	string	Detener el procesamiento de las siguientes anulaciones si coincide.

Element	Type	Description
filter		Elemento raíz para los elementos de filtro de la regla de anulación de la regla de descubrimiento de bajo nivel, que son los mismos que para los filtros de reglas de descubrimiento de bajo nivel de host.
operations		Elemento raíz para las operaciones de anulación de reglas de descubrimiento de bajo nivel de host.

Note:
Véase también: objeto de regla LLD (consulte la propiedad correspondiente con un nombre coincidente).

Filtros de la regla de descubrimiento de bajo nivel de host

```
zabbix_export:
  (...)
  hosts:
    - host: 'MySQL server'
      (...)
      discovery_rules:
        - name: 'Database discovery'
          (...)
          filter:
            evaltype: AND
            conditions:
              - macro: '{#DATABASE}'
                value: '{$MYSQL.DBNAME.MATCHES}'
              - macro: '{#DATABASE}'
                value: '{$MYSQL.DBNAME.NOT_MATCHES}'
                operator: NOT_MATCHES_REGEX
          (...)
```

Element	Type	Description
evaltype	string	Anula el método de evaluación de las condiciones del filtro. Valores posibles:[1] AND_OR (0, predeterminado), AND (1), OR (2), FORMULA (3).
formula	string	Fórmula de cálculo personalizada para las condiciones del filtro.
conditions		Elemento raíz para las condiciones del filtro.
macro	string	(obligatorio) El nombre de la macro de descubrimiento de bajo nivel sobre la que se realizará la comprobación.
value	string	Valor con el que comparar.
operator	string	Operador de la condición. Valores posibles:[1] MATCHES_REGEX (8, predeterminado), NOT_MATCHES_REGEX (9).
formulaid	string	(obligatorio) ID único arbitrario que se usa para referenciar una condición desde la expresión personalizada. Solo puede contener letras mayúsculas. El ID debe ser definido por el usuario al modificar las condiciones del filtro, pero se generará de nuevo al solicitarlas posteriormente.

Note:
Véase también: Objeto de filtro de la regla LLD (consulte la propiedad correspondiente con el nombre coincidente).

Operaciones de anulación de reglas de descubrimiento de bajo nivel de host

Element	Type	Description
operationobject	string	Objeto al que se aplica la operación. Valores posibles:[1] ITEM_PROTOTYPE (0), TRIGGER_PROTOTYPE (1), GRAPH_PROTOTYPE (2), HOST_PROTOTYPE (3), DISCOVERY_PROTOTYPE (4).
operator	string	Operador de la condición de anulación. Valores posibles:[1] EQUAL (1), NOT_EQUAL (2), LIKE (3), NOT_LIKE (4), REGEXP (5), NOT_REGEXP (6).
value	string	Una expresión regular o una cadena para el operador de condición de anulación.
status	string	Estado del objeto tras la operación de anulación.

Element	Type	Description
discover	string	Indica si el objeto se agrega como resultado del descubrimiento.
delay	string	Intervalo de actualización establecido para el prototipo de item tras la operación de anulación.
history	string	Período de almacenamiento del historial establecido para el prototipo de item tras la operación de anulación.
trends	string	Período de almacenamiento de tendencias establecido para el prototipo de item tras la operación de anulación.
severity	string	Gravedad del prototipo de trigger establecida tras la operación de anulación.
tags		Elemento raíz para las etiquetas establecidas para el objeto tras la operación de anulación.
tag	string	(obligatorio) Nombre de la etiqueta.
value	string	Valor de la etiqueta.
templates		Elemento raíz para las templates vinculadas al prototipo de host tras la operación de anulación.
name	string	(obligatorio) Nombre de la template.
inventory_mode	string	Modo de inventario del prototipo de host establecido tras la operación de anulación.

Note:
Véase también: Objeto de operación de anulación de la regla LLD (consulte la propiedad correspondiente con el mismo nombre).

Escenarios web del host

Element	Type	Description
uuid	string	(required) Identificador único de este escenario web.
name	string	(required) Nombre del escenario web.
delay	string	Frecuencia (usando segundos, time suffix, o user macro) de ejecución del escenario web. Default: 1m.
attempts	integer	Número de intentos para ejecutar los pasos del escenario web. Valores posibles: 1-10 (default: 1).
agent	string	Agent del cliente. Zabbix se hará pasar por el navegador seleccionado. Esto es útil cuando un sitio web devuelve contenido diferente para distintos navegadores. Default: Zabbix.
http_proxy	string	Proxy que se usará para el escenario web, indicado como: `http://[username[:password]@]proxy.example.com[:port]`
variables		Elemento raíz para las variables del escenario web que pueden usarse en los pasos del escenario.
name	string	(required) Nombre de la variable.
value	text	(required) Valor de la variable.
headers		Elemento raíz para las cabeceras HTTP que se enviarán al realizar una solicitud. Las cabeceras deben enumerarse usando la misma sintaxis que tendrían en el protocolo HTTP.
name	string	(required) Nombre de la cabecera.
value	text	(required) Valor de la cabecera.
status	string	Estado del escenario web. Valores posibles:[1] ENABLED (0, default), DISABLED (1).
authentication	string	Método de autenticación. Valores posibles:[1] NONE (0, default), BASIC (1), NTLM (2), Kerberos (3), o Digest (4).
http_user	string	Nombre de usuario usado para autenticación BASIC (HTTP), NTLM, Kerberos, o Digest.
http_password	string	Contraseña usada para autenticación BASIC (HTTP), NTLM, Kerberos, o Digest.
verify_peer	string	Verificar el certificado SSL del servidor web. Valores posibles:[1] NO (0, default), YES (1).
verify_host	string	Verificar que el campo *Common Name* o el campo *Subject Alternate Name* del certificado del servidor web coincida. Valores posibles:[1] NO (0, default), YES (1).
ssl_cert_file	string	Nombre del archivo de certificado SSL usado para la autenticación del cliente (debe estar en formato PEM).

Element	Type	Description
ssl_key_file	string	Nombre del archivo de clave privada SSL usado para la autenticación del cliente (debe estar en formato PEM).
ssl_key_password	string	Contraseña del archivo de clave privada SSL.
steps		(required) Elemento raíz para los pasos del escenario web del host.
tags		Elemento raíz para las etiquetas del escenario web.
tag	string	(required) Nombre de la etiqueta.
value	string	Valor de la etiqueta.

Note:
Véase también: Objeto de escenario web (consulte la propiedad correspondiente con el mismo nombre).

Pasos del escenario web del host

Element	Type	Description
name	string	(required) Nombre del paso del escenario web.
url	string	(required) URL para la monitorización.
query_fields		Elemento raíz para los parámetros de consulta (un array de campos HTTP que se añadirán a la URL al realizar una solicitud).
name	string	(required) Nombre del parámetro de consulta.
value	string	Valor del parámetro de consulta.
posts		Elemento raíz para las variables HTTP POST (una cadena (datos POST sin procesar) o un array de campos HTTP (datos de campos de formulario)).
name	string	(required) Nombre del campo POST.
value	string	(required) Valor del campo POST.
variables		Elemento raíz de las variables a nivel de paso (macros) que deben aplicarse después de este paso. Si el valor de la variable tiene el prefijo 'regex:', su valor se extrae de los datos devueltos por este paso según el patrón de expresión regular que sigue al prefijo 'regex:'
name	string	(required) Nombre de la variable.
value	text	(required) Valor de la variable.
headers		Elemento raíz para las cabeceras HTTP que se enviarán al realizar una solicitud.
name	string	(required) Nombre de la cabecera.
value	text	(required) Valor de la cabecera.
follow_redirects	string	Seguir redirecciones HTTP. Valores posibles:[1] NO (0), YES (1, predeterminado).
retrieve_mode	string	Modo de recuperación de la respuesta HTTP. Valores posibles:[1] BODY (0, predeterminado), HEADERS (1), BOTH (2).
timeout	string	Tiempo de espera (en segundos, sufijo de tiempo o macro de usuario) de la ejecución del paso. Predeterminado: 15s.
required	string	Texto que debe estar presente en la respuesta (se ignora si está vacío).
status_codes	string	Lista separada por comas de códigos de estado HTTP aceptados (por ejemplo, 200-201,210-299; se ignora si está vacío).

Note:
Véase también: Objeto de paso del escenario web (consulte la propiedad correspondiente con el mismo nombre).

Gráficos de host

```
zabbix_export:
  (...)
  graphs:
    - name: 'MySQL: Bandwidth'
      graph_items: (see table below)
    (...)
```

Element	Type	Description
uuid	string	Identificador único del gráfico.
name	string	(required) Nombre del gráfico.
width	integer	Ancho del gráfico, en píxeles. Se usa para la vista previa y para gráficos de tipo pie/exploded. Valores posibles:[1] 20-65535 (default: 900).
height	integer	Altura del gráfico, en píxeles. Se usa para la vista previa y para gráficos de tipo pie/exploded. Valores posibles:[1] 20-65535 (default: 900).
yaxismin	double	Valor mínimo del eje Y. Compatible con el valor mínimo FIXED del eje Y. Default: 0.
yaxismax	double	Valor máximo del eje Y. Compatible con el valor mínimo FIXED del eje X. Default: 0.
show_work_period	string	Resaltar las horas no laborables. Compatible con gráficos NORMAL y STACKED. Valores posibles:[1] NO (0), YES (1, default).
show_triggers	string	Mostrar los valores de trigger simples como una línea. Compatible con gráficos NORMAL y STACKED. Valores posibles:[1] NO (0), YES (1, default).
type	string	Tipo de gráfico. Valores posibles:[1] NORMAL (0, default), STACKED (1), PIE (2), EXPLODED (3).
show_legend	string	Mostrar la leyenda del gráfico. Valores posibles:[1] NO (0), YES (1, default).
show_3d	string	Habilitar estilo 3D. Compatible con gráficos NORMAL y STACKED. Valores posibles:[1] NO (0, default), YES (1).
percent_left	double	Mostrar la línea de percentil para el eje izquierdo. Compatible con gráficos NORMAL. Default: 0.
percent_right	double	Mostrar la línea de percentil para el eje derecho. Compatible con gráficos NORMAL. Default: 0.
ymin_type_1	string	Valor mínimo del eje Y. Compatible con gráficos NORMAL y STACKED. Valores posibles:[1] CALCULATED (0, default), FIXED (1), ITEM (2).
ymin_item_1		(required if `ymin_type_1` is set to `ITEM`) Elemento raíz para los detalles de cada item.
host	string	(required) Host del item.
key	string	(required) Clave del item.
ymax_type_1	string	Valor máximo del eje Y. Compatible con gráficos NORMAL y STACKED. Valores posibles:[1] CALCULATED (0, default), FIXED (1), ITEM (2).
ymax_item_1		(required if `ymax_type_1` is set to `ITEM`) Elemento raíz para los detalles de cada item.
host	string	(required) Host del item.
key	string	(required) Clave del item.
graph_items		(required) Elemento raíz para elementos del gráfico del host.

Note:
See also: Graph object (refer to the relevant property with a matching name).

Elementos del gráfico del host

```
zabbix_export:
  (...)
  graphs:
    - name: 'MySQL: Bandwidth'
      graph_items:
        - drawtype: GRADIENT_LINE
          color: 199C0D
          item:
            host: 'MySQL server'
```

```
            key: mysql.bytes_received.rate
        - sortorder: '1'
          drawtype: GRADIENT_LINE
          color: F63100
          item:
            host: 'MySQL server'
            key: mysql.bytes_sent.rate
    (...)
```

Elemento	Tipo	Descripción
sortorder	integer	Orden de dibujo. El valor más pequeño se dibuja primero. Puede usarse para dibujar líneas o regiones detrás de otra, o delante de ella.
drawtype	string	Estilo de dibujo del elemento del gráfico. Compatible con gráficos NORMAL. Valores posibles:[1] SINGLE_LINE (0, predeterminado), FILLED_REGION (1), BOLD_LINE (2), DOTTED_LINE (3), DASHED_LINE (4), GRADIENT_LINE (5).
color	string	Color del elemento (6 símbolos, hex).
yaxisside	string	Lado del gráfico donde se dibujará la escala Y del elemento del gráfico. Compatible con gráficos NORMAL y STACKED. Valores posibles:[1] LEFT (0, predeterminado), RIGHT (1).
calc_fnc	string	Datos que se dibujarán si existe más de un valor para un item. Valores posibles:[1] MIN (1), AVG (2, predeterminado), MAX (4), ALL (7; mínimo, promedio y máximo; compatible con gráficos simples), LAST (9, compatible con gráficos de pastel/explotados).
type	string	Tipo de elemento del gráfico. Valores posibles:[1] SIMPLE (0, predeterminado), GRAPH_SUM (2; el valor del item representa el pastel completo; compatible con gráficos de pastel/explotados).
item		(obligatorio) Item individual.
host	string	(obligatorio) Host del item.
key	string	(obligatorio) Clave del item.

Note:
Véase también: Objeto graph item (consulte la propiedad correspondiente con el mismo nombre).

Mapas de valores del host

```
zabbix_export:
  (...)
  hosts:
    - host: 'MySQL server'
      (...)
      valuemaps:
        - name: Example value map
          mappings:
            - value: '1'
              newvalue: Example value
  (...)
```

Element	Type	Description
uuid	string	(required) Identificador único para este mapa de valores.
name	string	(required) Nombre del mapa de valores.
mapping		Elemento raíz para las asignaciones.
type	string	Tipo de coincidencia de la asignación. Valores posibles:[1] EQUAL (0, predeterminado), GREATER_OR_EQUAL (2), LESS_OR_EQUAL (3), IN_RANGE (4), REGEXP (5), DEFAULT (6).
value	string	Valor original.
newvalue	string	(required) Valor al que se asigna el valor original.

Note:
Véase también: Objeto de mapa de valores (consulte la propiedad correspondiente con un nombre coincidente).

Notas al pie [1] Los valores enteros de la API entre paréntesis, por ejemplo, ENABLED (0), se mencionan solo como referencia. Para obtener más información, consulte la página del objeto de API enlazada en la entrada de la tabla o al final de cada sección.

Mapas de red

Resumen

El mapa de red exportación contiene:

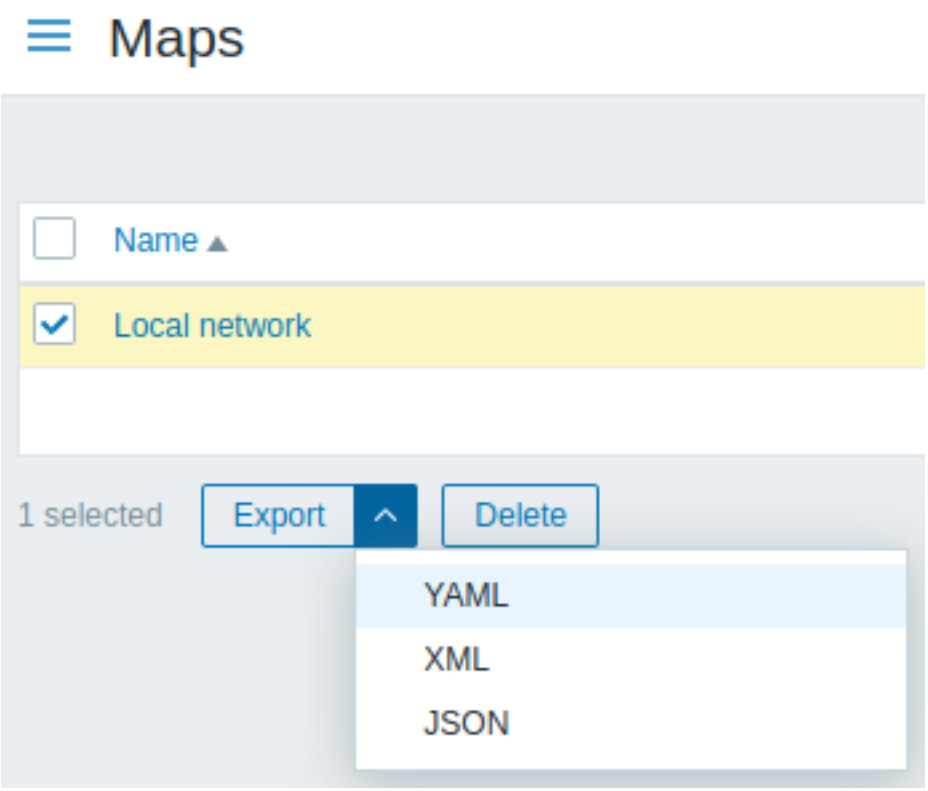
- Todas las imágenes relacionadas
- La estructura del mapa (todos los ajustes del mapa, todos los elementos que contiene con sus ajustes, los enlaces del mapa y los indicadores de estado de los enlaces del mapa)

Warning:
No se exportan los grupos de host, hosts, triggers, otros mapas ni otros elementos que puedan estar relacionados con el mapa exportado. Por lo tanto, si falta al menos uno de los elementos a los que hace referencia el mapa, la importación fallará.

Exportación

Para exportar mapas de red, haga lo siguiente:

1. Vaya a *Monitoring* > *Maps*.
2. Marque las casillas de verificación de los mapas de red que desea exportar.
3. Haga clic en *Export* debajo de la lista para exportar en el formato YAML predeterminado, o haga clic en la flecha para seleccionar un formato de exportación diferente.



Según el formato seleccionado, los mapas se exportan a un archivo local con un nombre predeterminado:

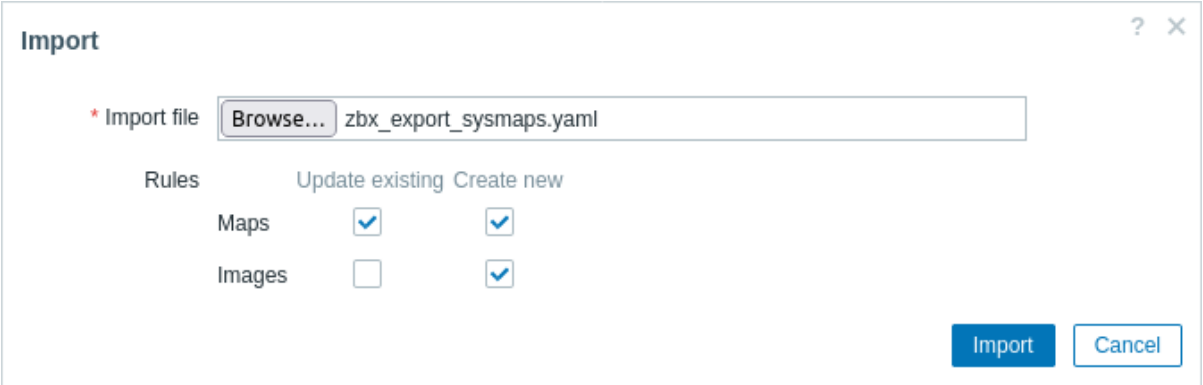
- `zabbix_export_maps.yaml` - en exportación YAML;
- `zabbix_export_maps.xml` - en exportación XML;
- `zabbix_export_maps.json` - en exportación JSON.

Importación

Para importar mapas de red, haga lo siguiente:

1. Vaya a *Monitoring* > *Maps*.
2. Haga clic en *Import* en la esquina superior derecha.
3. Seleccione el archivo de importación.
4. Marque las casillas de las reglas de importación que desea aplicar (consulte más abajo).

5. Haga clic en *Import* en la esquina inferior derecha del formulario de configuración.



Reglas de importación:

Regla	Descripción
Update existing	Los mapas existentes se actualizarán utilizando los datos del archivo de importación. De lo contrario, no se actualizarán.
Create new	Se crearán mapas nuevos utilizando los datos del archivo de importación. De lo contrario, no se crearán.

Si desmarca ambas opciones de mapas y marca las opciones correspondientes para las imágenes, solo se importarán las imágenes. La importación de imágenes solo está disponible para usuarios *Super admin*.

No puede revisar los cambios en los mapas de red durante la importación. En el frontend se mostrará un mensaje de éxito o de error de la importación.

Warning:
Si reemplaza una imagen existente, esto afectará a todos los mapas que utilicen dicha imagen.

Formato de exportación

Exportar a YAML:

```yaml
zabbix_export:
  version: '8.0'
  images:
    - name: Zabbix_server_3D_(128)
      imagetype: '1'
      encodedImage: iVBOR...5CYII=
  maps:
    - name: 'Red local'
      width: '680'
      height: '200'
      label_type: '0'
      label_location: '0'
      highlight: '1'
      expandproblem: '1'
      markelements: '1'
      show_unack: '0'
      severity_min: '0'
      show_suppressed: '0'
      grid_size: '50'
      grid_show: '1'
      grid_align: '1'
      label_format: '0'
      label_type_host: '2'
      label_type_hostgroup: '2'
      label_type_trigger: '2'
      label_type_map: '2'
      label_type_image: '2'
      label_string_host: ''
```

```
    label_string_hostgroup: ''
    label_string_trigger: ''
    label_string_map: ''
    label_string_image: ''
    expand_macros: '1'
    background: {  }
    iconmap: {  }
    urls: {  }
    selements:
      - elementtype: '0'
        elements:
          - host: 'Zabbix server'
        label: |
          {HOST.NAME}
          {HOST.CONN}
        label_location: '0'
        x: '111'
        'y': '61'
        elementsubtype: '0'
        areatype: '0'
        width: '200'
        height: '200'
        viewtype: '0'
        use_iconmap: '0'
        selementid: '1'
        icon_off:
          name: Zabbix_server_3D_(128)
        icon_on: {  }
        icon_disabled: {  }
        icon_maintenance: {  }
        urls: {  }
        evaltype: '0'
    shapes:
      - type: '0'
        x: '0'
        'y': '0'
        width: '680'
        height: '15'
        text: '{MAP.NAME}'
        font: '9'
        font_size: '11'
        font_color: '000000'
        text_halign: '0'
        text_valign: '0'
        border_type: '0'
        border_width: '0'
        border_color: '000000'
        background_color: ''
        zindex: '0'
    lines: {  }
    links: {  }
```

Elementos exportados

Los elementos exportados se explican en la tabla siguiente.

Element	Type	Description
images		Elemento raíz para las imágenes.
name	string	Nombre único de la imagen.
imagetype	integer	Tipo de imagen. Valores posibles: 1 - Imagen; 2 - Fondo.

Element	Type	Description
encodedImage	string	Imagen codificada en Base64.
maps		Elemento raíz para maps.

Maps

Element	Type	Description
name	string	Nombre único del mapa.
width	integer	Ancho del mapa, en píxeles.
height	integer	Altura del mapa, en píxeles.
label_type	integer	Tipo de etiqueta del elemento del mapa. Valores posibles: 0 - Etiqueta; 1 - Dirección IP del host; 2 - Nombre del elemento; 3 - Solo estado; 4 - Nada.
label_location	integer	Ubicación predeterminada de la etiqueta del elemento del mapa. Valores posibles: 0 - Abajo; 1 - Izquierda; 2 - Derecha; 3 - Arriba.
highlight	integer	Habilitar el resaltado de iconos para triggers activos y estados de host. Valores posibles: 0 - No; 1 - Sí.
expandproblem	integer	Mostrar el trigger de problema para elementos con un único problema. Valores posibles: 0 - No; 1 - Sí.
markelements	integer	Resaltar los elementos del mapa que han cambiado recientemente su estado. Valores posibles: 0 - No; 1 - Sí.
show_unack	integer	Visualización de problemas. Valores posibles: 0 - Recuento de todos los problemas; 1 - Recuento de problemas no reconocidos; 2 - Recuento de problemas reconocidos y no reconocidos por separado.
severity_min	integer	Severidad mínima del trigger que se mostrará en el mapa de forma predeterminada. Valores posibles: 0 - No clasificado; 1 - Información; 2 - Advertencia; 3 - Promedio; 4 - Alto; 5 - Desastre.
show_suppressed	integer	Mostrar problemas que, de otro modo, estarían suprimidos (no mostrados) debido al mantenimiento del host. Valores posibles: 0 - No; 1 - Sí.
grid_size	integer	Tamaño de celda de una cuadrícula de mapa en píxeles. Compatible si `grid_show` está establecido en 0. Valores posibles: 20, 40, 50, 75 o 100.
grid_show	integer	Mostrar una cuadrícula en la configuración del mapa. Valores posibles: 0 - Sí; 1 - No.

Element	Type	Description
grid_align	integer	Alinear automáticamente los iconos en la configuración del mapa. Valores posibles: 0 - Sí; 1 - No.
label_format	integer	Usar configuración avanzada de etiquetas. Valores posibles: 0 - No; 1 - Sí.
label_type_host	integer	Mostrar la etiqueta como etiqueta de host. Compatible si `label_format` está establecido en 1. Valores posibles: 0 - Etiqueta; 1 - Dirección IP del host; 2 - Nombre del elemento; 3 - Solo estado; 4 - Nada; 5 - Etiqueta personalizada.
label_type_hostgroup	integer	Mostrar la etiqueta como etiqueta de grupo de hosts. Compatible si `label_format` está establecido en 1. Valores posibles: 0 - Etiqueta; 2 - Nombre del elemento; 3 - Solo estado; 4 - Nada; 5 - Etiqueta personalizada.
label_type_trigger	integer	Mostrar la etiqueta como etiqueta de trigger. Compatible si `label_format` está establecido en 1. Valores posibles: 0 - Etiqueta; 2 - Nombre del elemento; 3 - Solo estado; 4 - Nada; 5 - Etiqueta personalizada.
label_type_map	integer	Mostrar la etiqueta como etiqueta de mapa. Compatible si `label_format` está establecido en 1. Valores posibles: 0 - Etiqueta; 2 - Nombre del elemento; 3 - Solo estado; 4 - Nada; 5 - Etiqueta personalizada.
label_type_image	integer	Mostrar la etiqueta como etiqueta de imagen. Compatible si `label_format` está establecido en 1. Valores posibles: 0 - Etiqueta; 2 - Nombre del elemento; 4 - Nada; 5 - Etiqueta personalizada.
label_string_host	string	Etiqueta personalizada para elementos de host. Compatible si `label_type_host` está establecido en 5.
label_string_hostgroup	string	Etiqueta personalizada para elementos de grupo de hosts. Compatible si `label_type_hostgroup` está establecido en 5.
label_string_trigger	string	Etiqueta personalizada para elementos de trigger. Compatible si `label_type_trigger` está establecido en 5.
label_string_map	string	Etiqueta personalizada para elementos de mapa. Compatible si `label_type_map` está establecido en 5.
label_string_image	string	Etiqueta personalizada para elementos de imagen. Compatible si `label_type_image` está establecido en 5.
expand_macros	integer	Expandir macros en las etiquetas en la configuración del mapa. Valores posibles: 0 - No; 1 - Sí.

Element	Type	Description
background		Elemento raíz para la imagen de fondo (si existe). Compatible si `imagetype` está establecido en 2.
name	string	Nombre de la imagen de fondo.
iconmap		Elemento raíz para el mapeo de iconos (si existe).
name	string	Nombre del mapeo de iconos.
urls		Elemento raíz para las URLs utilizadas por los mapas o por cada elemento del mapa.
name	string	Nombre del enlace.
url	string	URL del enlace.
elementtype	integer	Tipo de elemento del mapa al que pertenece el enlace. Valores posibles: 0 - Host; 1 - Mapa; 2 - Trigger; 3 - Grupo de hosts; 4 - Imagen.
selements		Elemento raíz para map selements.
shapes		Elemento raíz para las formas del mapa.
type	integer	Tipo de forma. Valores posibles: 0 - Rectángulo; 1 - Elipse.
x	integer	Coordenadas X de la forma en píxeles.
y	integer	Coordenadas Y de la forma en píxeles.
width	integer	Ancho de la forma.
height	integer	Altura de la forma.
text	string	Texto dentro de la forma.
font	integer	Estilo de fuente del texto. Valores posibles: 0 - Georgia, serif; 1 - ”Palatino Linotype”, ”Book Antiqua”, Palatino, serif; 2 - ”Times New Roman”, Times, serif; 3 - Arial, Helvetica, sans-serif; 4 - ”Arial Black”, Gadget, sans-serif; 5 - ”Comic Sans MS”, cursive, sans-serif; 6 - Impact, Charcoal, sans-serif; 7 - ”Lucida Sans Unicode”, ”Lucida Grande”, sans-serif; 8 - Tahoma, Geneva, sans-serif; 9 - ”Trebuchet MS”, Helvetica, sans-serif; 10 - Verdana, Geneva, sans-serif; 11 - ”Courier New”, Courier, monospace; 12 - ”Lucida Console”, Monaco, monospace.
font_size	integer	Tamaño de fuente en píxeles.
font_color	string	Color de la fuente representado en código hexadecimal.
text_halign	integer	Alineación horizontal del texto. Valores posibles: 0 - Centro; 1 - Izquierda; 2 - Derecha.
text_valign	integer	Alineación vertical del texto. Valores posibles: 0 - Medio; 1 - Arriba; 2 - Abajo.
border_type	integer	Tipo de borde para la forma. Valores posibles: 0 - Ninguno; 1 - Línea gruesa; 2 - Línea punteada; 3 - Línea discontinua.
border_width	integer	Ancho del borde en píxeles.
border_color	string	Color del borde representado en código hexadecimal.

Element		Type	Description
	background_color	string	Color de fondo (relleno) representado en código hexadecimal.
	zindex	integer	Valor para el orden de todas las formas y líneas (z-index).
lines			Elemento raíz para las líneas del mapa.
	x1	integer	Coordenadas X del punto 1 de la línea en píxeles.
	y1	integer	Coordenadas Y del punto 1 de la línea en píxeles.
	x2	integer	Coordenadas X del punto 2 de la línea en píxeles.
	y2	integer	Coordenadas Y del punto 2 de la línea en píxeles.
	line_type	integer	Tipo de línea. Valores posibles: 0 - Ninguno; 1 - Línea gruesa; 2 - Línea punteada; 3 - Línea discontinua.
	line_width	integer	Ancho de la línea en píxeles.
	line_color	string	Color de la línea representado en código hexadecimal.
	zindex	integer	Valor para el orden de todas las formas y líneas (z-index).
links			Elemento raíz para los enlaces entre elementos del mapa.
	drawtype	integer	Estilo del enlace. Valores posibles: 0 - Línea; 2 - Línea gruesa; 3 - Línea punteada; 4 - Línea discontinua.
	color	string	Color del enlace (6 símbolos, hex).
	label	string	Etiqueta del enlace.
	selementid1	id	ID de uno de los elementos a conectar.
	selementid2	id	ID del otro elemento a conectar.
linktriggers			Elemento raíz para indicadores de estado de enlace.

Note:
Véase también: Objeto Map (consulte la propiedad correspondiente con el mismo nombre).

Elementos de mapas

Element	Type	Description
elementtype	integer	Tipo de elemento del mapa. Valores posibles: 0 - Host; 1 - Map; 2 - Trigger; 3 - Grupo de hosts; 4 - Imagen.
elements		Elemento raíz para entidades de Zabbix (host, grupo de hosts, map, etc.) que se representan en el mapa. Todas las entidades tienen su respectivo elemento (`host`, etc.; consulte, por ejemplo, Export format).
label	string	Etiqueta del icono.
label_location	integer	Ubicación de la etiqueta. Valores posibles: -1 - Usar el valor predeterminado del mapa; 0 - Inferior; 1 - Izquierda; 2 - Derecha; 3 - Superior.
x	integer	Ubicación en el eje X.
y	integer	Ubicación en el eje Y.

Element	Type	Description
elementsubtype	integer	Subtipo de elemento. Compatible si `elementtype` está establecido en 3. Valores posibles: 0 - Un solo grupo de hosts; 1 - Todos los grupos de hosts.
areatype	integer	Tamaño del área. Compatible si `elementtype` está establecido en 1. Valores posibles: 0 - Igual que todo el mapa; 1 - Tamaño personalizado.
width	integer	Ancho del área. Compatible si `areatype` está establecido en 1.
height	integer	Altura del área. Compatible si `areatype` está establecido en 1.
viewtype	integer	Algoritmo de colocación del área. Compatible si `elementsubtype` está establecido en 1. Valores posibles: 0 - Colocar uniformemente en el área.
use_iconmap	integer	Usar el mapeo de iconos para este elemento. Relevante solo si el mapeo de iconos está activado a nivel de mapa. Valores posibles: 0 - No; 1 - Sí.
selementid	id	ID único del registro del elemento.
icon_off		Elemento raíz para la imagen que se usará cuando el elemento esté en estado `OK`.
name	string	Nombre único de la imagen.
icon_on		Elemento raíz para la imagen que se usará cuando el elemento esté en estado `Problem`.
name	string	Nombre único de la imagen.
icon_disabled		Elemento raíz para la imagen que se usará cuando el elemento esté deshabilitado.
name	string	Nombre único de la imagen.
icon_maintenance		Elemento raíz para la imagen que se usará cuando el elemento esté en mantenimiento.
name	string	Nombre único de la imagen.
urls		Elemento raíz para las URLs utilizadas por los mapas o por cada elemento del mapa.
name	string	Nombre del enlace.
url	string	URL del enlace.
evaltype	integer	Tipo de evaluación para las etiquetas.
tags		Elemento raíz para las etiquetas de problema (para elementos de host y grupo de hosts). Si se proporcionan etiquetas, solo se mostrarán en el mapa los problemas con estas etiquetas.
tag	string	Nombre de la etiqueta.
value	string	Valor de la etiqueta.
operator	integer	Operador.

Note:
Véase también: Objeto de elemento de mapa (consulte la propiedad correspondiente con el mismo nombre).

Indicadores de estado del enlace del mapa

Elemento	Tipo	Descripción
drawtype	integer	Estilo del enlace cuando el trigger está en el estado 'Problem'. Valores posibles: 0 - Línea; 2 - Línea gruesa; 3 - Línea punteada; 4 - Línea discontinua.
color	string	Color del enlace (6 símbolos, hexadecimal) cuando el trigger está en el estado 'Problem'.

Elemento	Tipo	Descripción
trigger		Elemento raíz del trigger utilizado para indicar el estado del enlace.
description	string	Nombre del trigger.
expression	string	Expresión del trigger.
recovery_expression	string	Expresión de recuperación del trigger.

Note:
Véase también: Objeto trigger del enlace del mapa (consulte la propiedad correspondiente con el mismo nombre).

Tipos de medios

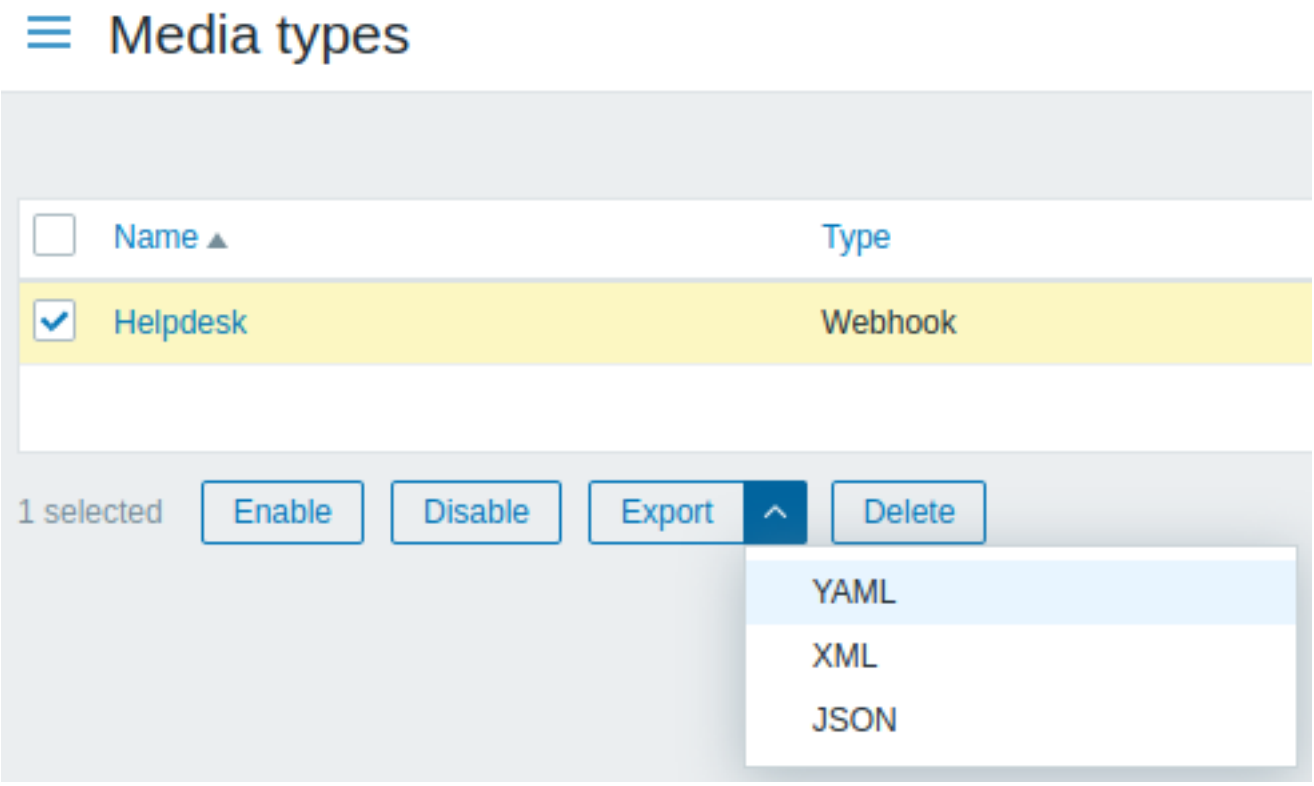
Resumen

Los tipos de medio se exportan con todos los objetos relacionados y las relaciones entre objetos.

Exportación

Para exportar tipos de medios, haga lo siguiente:

1. Vaya a *Alerts* > *Media types*.
2. Marque las casillas de verificación de los tipos de medios que desea exportar.
3. Haga clic en *Export* debajo de la lista para exportar en el formato YAML predeterminado, o haga clic en la flecha para seleccionar un formato de exportación diferente.



Según el formato seleccionado, los tipos de medios se exportan a un archivo local con un nombre predeterminado:

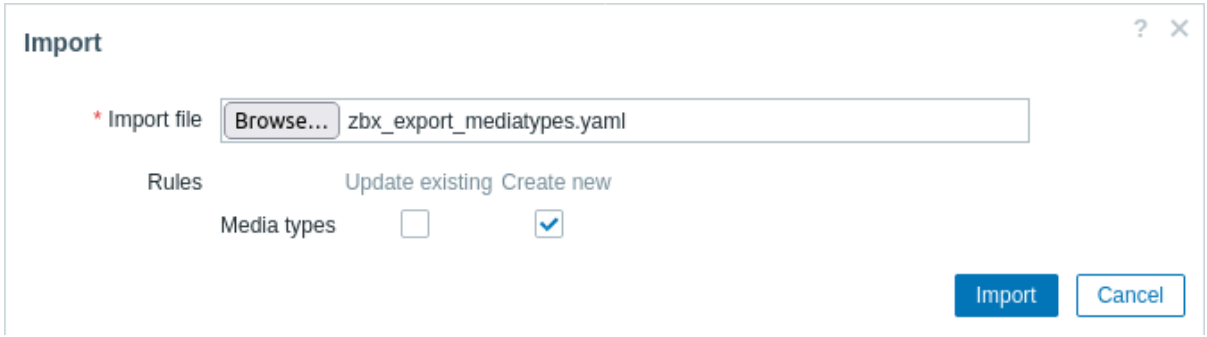
- `zabbix_export_mediatypes.yaml` - en exportación YAML;
- `zabbix_export_mediatypes.xml` - en exportación XML;
- `zabbix_export_mediatypes.json` - en exportación JSON.

Importación

Para importar tipos de medios, haga lo siguiente:

1. Vaya a *Alertas* > *Tipos de medios*.
2. Haga clic en *Importar* en la esquina superior derecha.
3. Seleccione el archivo de importación.
4. Marque las casillas de verificación de las reglas de importación que desea aplicar (consulte a continuación).

5. Haga clic en *Importar* en la esquina inferior derecha del formulario de configuración.



Reglas de importación:

Regla	Descripción
Actualizar existentes	Marque esta casilla de verificación para actualizar los tipos de medios existentes utilizando los datos del archivo de importación.
Crear nuevos	Marque esta casilla de verificación para crear nuevos tipos de medios utilizando los datos del archivo de importación. Si el archivo de importación contiene un tipo de medio que ya existe (por ejemplo, *Correo electrónico*), no se creará uno nuevo y tampoco se actualizará el tipo de medio existente.

No puede revisar los cambios en los tipos de medios durante la importación. En el frontend se mostrará un mensaje de éxito o de error de la importación.

Formato de exportación

Exportar a YAML:

```yaml
zabbix_export:
  version: '8.0'
  media_types:
    - name: Pushover
      type: WEBHOOK
      parameters:
        - name: endpoint
          value: 'https://api.pushover.net/1/messages.json'
        - name: eventid
          value: '{EVENT.ID}'
        - name: event_nseverity
          value: '{EVENT.NSEVERITY}'
        - name: event_source
          value: '{EVENT.SOURCE}'
        - name: event_value
          value: '{EVENT.VALUE}'
        - name: expire
          value: '1200'
        - name: message
          value: '{ALERT.MESSAGE}'
        - name: priority_average
          value: '0'
        - name: priority_default
          value: '0'
        - name: priority_disaster
          value: '0'
        - name: priority_high
          value: '0'
        - name: priority_information
          value: '0'
        - name: priority_not_classified
          value: '0'
        - name: priority_warning
          value: '0'
```

```
    - name: retry
      value: '60'
    - name: title
      value: '{ALERT.SUBJECT}'
    - name: token
      value: '<PUSHOVER TOKEN HERE>'
    - name: triggerid
      value: '{TRIGGER.ID}'
    - name: url
      value: '{$ZABBIX.URL}'
    - name: url_title
      value: Zabbix
    - name: user
      value: '{ALERT.SENDTO}'
  status: DISABLED
  max_sessions: '0'
  script: |
    try {
        var params = JSON.parse(value),
            request = new HttpRequest(),
            data,
            response,
            severities = [
                {name: 'not_classified', color: '#97AAB3'},
                {name: 'information', color: '#7499FF'},
                {name: 'warning', color: '#FFC859'},
                {name: 'average', color: '#FFA059'},
                {name: 'high', color: '#E97659'},
                {name: 'disaster', color: '#E45959'},
                {name: 'resolved', color: '#009900'},
                {name: 'default', color: '#000000'}
            ],
            priority;

        if (typeof params.HTTPProxy === 'string' && params.HTTPProxy.trim() !== '') {
            request.setProxy(params.HTTPProxy);
        }

        if ([0, 1, 2, 3].indexOf(parseInt(params.event_source)) === -1) {
            throw 'Incorrect "event_source" parameter given: "' + params.event_source + '".\nMust be 0

        if (params.event_value !== '0' && params.event_value !== '1'
            && (params.event_source === '0' || params.event_source === '3')) {
            throw 'Incorrect "event_value" parameter given: ' + params.event_value + '\nMust be 0 or 1
        }

        if ([0, 1, 2, 3, 4, 5].indexOf(parseInt(params.event_nseverity)) === -1) {
            params.event_nseverity = '7';
        }

        if (params.event_value === '0') {
            params.event_nseverity = '6';
        }

        priority = params['priority_' + severities[params.event_nseverity].name] || params.priority_de

        if (isNaN(priority) || priority < -2 || priority > 2) {
            throw '"priority" should be -2..2';
        }

        if (params.event_source === '0' && isNaN(params.triggerid)) {
            throw 'field "triggerid" is not a number';
```

```
}

if (isNaN(params.eventid)) {
    throw 'field "eventid" is not a number';
}

if (typeof params.message !== 'string' || params.message.trim() === '') {
    throw 'field "message" cannot be empty';
}

data = {
    token: params.token,
    user: params.user,
    title: params.title,
    message: params.message,
    url: (params.event_source === '0')
        ? params.url + '/tr_events.php?triggerid=' + params.triggerid + '&eventid=' + params.e
        : params.url,
    url_title: params.url_title,
    priority: priority
};

if (priority == 2) {
    if (isNaN(params.retry) || params.retry < 30) {
        throw 'field "retry" should be a number with value of at least 30 if "priority" is set
    }

    if (isNaN(params.expire) || params.expire > 10800) {
        throw 'field "expire" should be a number with value of at most 10800 if "priority" is
    }

    data.retry = params.retry;
    data.expire = params.expire;
}

data = JSON.stringify(data);
Zabbix.log(4, '[ Pushover Webhook ] Sending request: ' + params.endpoint + '\n' + data);

request.addHeader('Content-Type: application/json');
response = request.post(params.endpoint, data);

Zabbix.log(4, '[ Pushover Webhook ] Received response with status code ' + request.getStatus()

if (response !== null) {
    try {
        response = JSON.parse(response);
    }
    catch (error) {
        Zabbix.log(4, '[ Pushover Webhook ] Failed to parse response received from Pushover');
        response = null;
    }
}

if (request.getStatus() != 200 || response === null || typeof response !== 'object' || respons
    if (response !== null && typeof response === 'object' && typeof response.errors === 'objec
            && typeof response.errors[0] === 'string') {
        throw response.errors[0];
    }
    else {
        throw 'Unknown error. Check debug log for more information.';
    }
}
```

```
        return 'OK';
    }
    catch (error) {
        Zabbix.log(4, '[ Pushover Webhook ] Pushover notification failed: ' + error);
        throw 'Pushover notification failed: ' + error;
    }
description: |
  Consulte la guía de configuración aquí: https://git.zabbix.com/projects/ZBX/repos/zabbix/browse/te

  Establezca el parámetro token con la clave de su aplicación Pushover.
  Al asignar el medio Pushover al usuario de Zabbix, agregue la clave de usuario en el campo "send t
message_templates:
  - event_source: TRIGGERS
    operation_mode: PROBLEM
    subject: 'Problema: {EVENT.NAME}'
    message: |
      El problema comenzó a las {EVENT.TIME} el {EVENT.DATE}
      Nombre del problema: {EVENT.NAME}
      host: {HOST.NAME}
      Severidad: {EVENT.SEVERITY}
      Datos operativos: {EVENT.OPDATA}
      ID del problema original: {EVENT.ID}
      {TRIGGER.URL}
  - event_source: TRIGGERS
    operation_mode: RECOVERY
    subject: 'Resuelto en {EVENT.DURATION}: {EVENT.NAME}'
    message: |
      El problema se resolvió a las {EVENT.RECOVERY.TIME} el {EVENT.RECOVERY.DATE}
      Nombre del problema: {EVENT.NAME}
      Duración del problema: {EVENT.DURATION}
      host: {HOST.NAME}
      Severidad: {EVENT.SEVERITY}
      ID del problema original: {EVENT.ID}
      {TRIGGER.URL}
  - event_source: TRIGGERS
    operation_mode: UPDATE
    subject: 'Problema actualizado en {EVENT.AGE}: {EVENT.NAME}'
    message: |
      {USER.FULLNAME} {EVENT.UPDATE.ACTION} el problema el {EVENT.UPDATE.DATE} a las {EVENT.UPDATE.T
      {EVENT.UPDATE.MESSAGE}

      El estado actual del problema es {EVENT.STATUS}, la antigüedad es {EVENT.AGE}, reconocido: {EV
  - event_source: DISCOVERY
    operation_mode: PROBLEM
    subject: 'Descubrimiento: {DISCOVERY.DEVICE.STATUS} {DISCOVERY.DEVICE.IPADDRESS}'
    message: |
      Regla de descubrimiento: {DISCOVERY.RULE.NAME}

      IP del dispositivo: {DISCOVERY.DEVICE.IPADDRESS}
      DNS del dispositivo: {DISCOVERY.DEVICE.DNS}
      Estado del dispositivo: {DISCOVERY.DEVICE.STATUS}
      Tiempo de actividad del dispositivo: {DISCOVERY.DEVICE.UPTIME}

      Nombre del servicio del dispositivo: {DISCOVERY.SERVICE.NAME}
      Puerto del servicio del dispositivo: {DISCOVERY.SERVICE.PORT}
      Estado del servicio del dispositivo: {DISCOVERY.SERVICE.STATUS}
      Tiempo de actividad del servicio del dispositivo: {DISCOVERY.SERVICE.UPTIME}
  - event_source: AUTOREGISTRATION
    operation_mode: PROBLEM
    subject: 'Autoregistro: {HOST.HOST}'
    message: |
      Nombre del host: {HOST.HOST}
```

```
            IP del host: {HOST.IP}
            Puerto del agent: {HOST.PORT}
```

Elementos exportados

Los elementos exportados se explican en la tabla siguiente.

Element	Type	Description
name	string	(obligatorio) Nombre del tipo de medio.
type	string	(obligatorio) Transporte utilizado por el tipo de medio. Valores posibles:[1] EMAIL (0), SMS (1), SCRIPT (2), WEBHOOK (4).
status	string	Indica si el tipo de medio está habilitado. Valores posibles:[1] ENABLED (0, predeterminado), DISABLED (1)
max_sessions	integer	El número máximo de alertas que se pueden procesar en paralelo. Valores posibles para SMS:[1] 1 (predeterminado). Valores posibles para otros tipos de medio:[1] 0-100 (donde 0 = ilimitado).
attempts	integer	El número máximo de intentos para enviar una alerta. Valores posibles:[1] 1-10 (predeterminado: 3).
attempt_interval	string	El intervalo (usando segundos o time suffix) entre los intentos de reintento. Valores posibles:[1] 0-60s (predeterminado: 10s).
description	string	Descripción del tipo de medio.
message_templates		Elemento raíz para las plantillas de mensajes del tipo de medio.
event_source	string	(obligatorio) Origen del evento. Valores posibles:[1] TRIGGERS (0), DISCOVERY (1), AUTOREGISTRATION (2), INTERNAL (3), SERVICE (4).
operation_mode	string	Modo de operación. Valores posibles:[1] PROBLEM (0), RECOVERY (1), UPDATE (2).
subject	string	Asunto del mensaje.
message	string	Cuerpo del mensaje.

Note:
Véase también: Media type object (consulte la propiedad correspondiente con el mismo nombre).

Email

Los siguientes elementos adicionales se exportan solo para el tipo de medio *Email*.

Element	Type	Description
provider	string	Proveedor de correo electrónico.
smtp_server	string	Servidor SMTP.
smtp_port	integer	Puerto del servidor SMTP al que conectarse. Predeterminado: 25.
smtp_helo	string	SMTP helo.
smtp_email	string	Dirección de correo electrónico desde la que se enviarán las notificaciones.
smtp_security	string	Nivel de seguridad de la conexión SMTP que se usará. Valores posibles:[1] NONE (0, predeterminado), STARTTLS (1), SSL_OR_TLS (2).
smtp_verify_host	string	Verificación SSL del host para SMTP. Valores posibles:[1] NO (0, predeterminado), YES (1).
smtp_verify_peer	string	Verificación SSL del peer para SMTP. Valores posibles:[1] NO (0, predeterminado), YES (1).
smtp_authentication	string	Método de autenticación SMTP que se usará. Valores posibles:[1] NONE (0, predeterminado), PASSWORD (1), OAUTH (2).
username	string	Nombre de usuario.
password	string	Contraseña de autenticación.
redirection_url	string	URL del frontend de Zabbix a la que redirigir de vuelta la autorización OAuth.
client_id	string	Identificador del cliente registrado en el servidor de autorización OAuth.
authorization_url	string	URL de OAuth, con parámetros, para obtener tokens de acceso y actualización.
token_url	string	URL de OAuth para intercambiar el token de autorización por tokens de acceso y actualización.

Element	Type	Description
message_format	string	Formato del mensaje. Valores posibles:[1] TEXT (0), HTML (1, predeterminado).

Note:
Véase también: Media type object (consulte la propiedad correspondiente con el nombre coincidente).

SMS

Los siguientes elementos adicionales se exportan solo para el tipo de medio *SMS*.

Element	Type	Description
gsm_modem	string	(required) Nombre del dispositivo serie del módem GSM.

Note:
Consulte también: Media type object (consulte la propiedad correspondiente con el mismo nombre).

Script

Los siguientes elementos adicionales se exportan solo para el tipo de medio *Script*.

Element	Type	Description
script name	string	(required) Nombre del script.
parameters		Elemento raíz para los parámetros del script.
sortorder	string	(required) Orden de los parámetros del script que se pasan al script como argumentos de línea de comandos.
value	string	Valor del parámetro del script.

Note:
Véase también: Media type object (consulte la propiedad correspondiente con el mismo nombre).

Webhook

Los siguientes elementos adicionales se exportan solo para el tipo de medio *Webhook*.

Element	Type	Description
script	string	Script.
timeout	string	Intervalo de tiempo de espera de la solicitud HTTP del script de Javascript. Valores posibles:[1] 1-60s (predeterminado: 30s).
process_tags	string	Indica si se deben procesar las etiquetas devueltas. Valores posibles:[1] NO (0, predeterminado), YES (1).
show_event_menu	string	Indica la presencia de una entrada en el menú de eventos si la macro {EVENT.TAGS.*} se resolvió correctamente en los campos `event_menu_url` y `event_menu_name`. Valores posibles:[1] NO (0, predeterminado), YES (1).
event_menu_url	string	URL de la entrada del menú de eventos. Admite la macro {EVENT.TAGS.*}.
event_menu_name	string	Nombre de la entrada del menú de eventos. Admite la macro {EVENT.TAGS.*}.
parameters		Elemento raíz para los parámetros del tipo de medio webhook.
name	string	(obligatorio) Nombre del parámetro de webhook.
value	string	Valor del parámetro de webhook.

Note:
Véase también: Media type object (consulte la propiedad correspondiente con el mismo nombre).

Notas al pie

[1] Los valores enteros de la API entre paréntesis, por ejemplo, ENABLED (0), se mencionan solo como referencia. Para obtener más información, consulte la página del objeto API enlazada en la entrada de la tabla o al final de cada sección.

Paneles de control

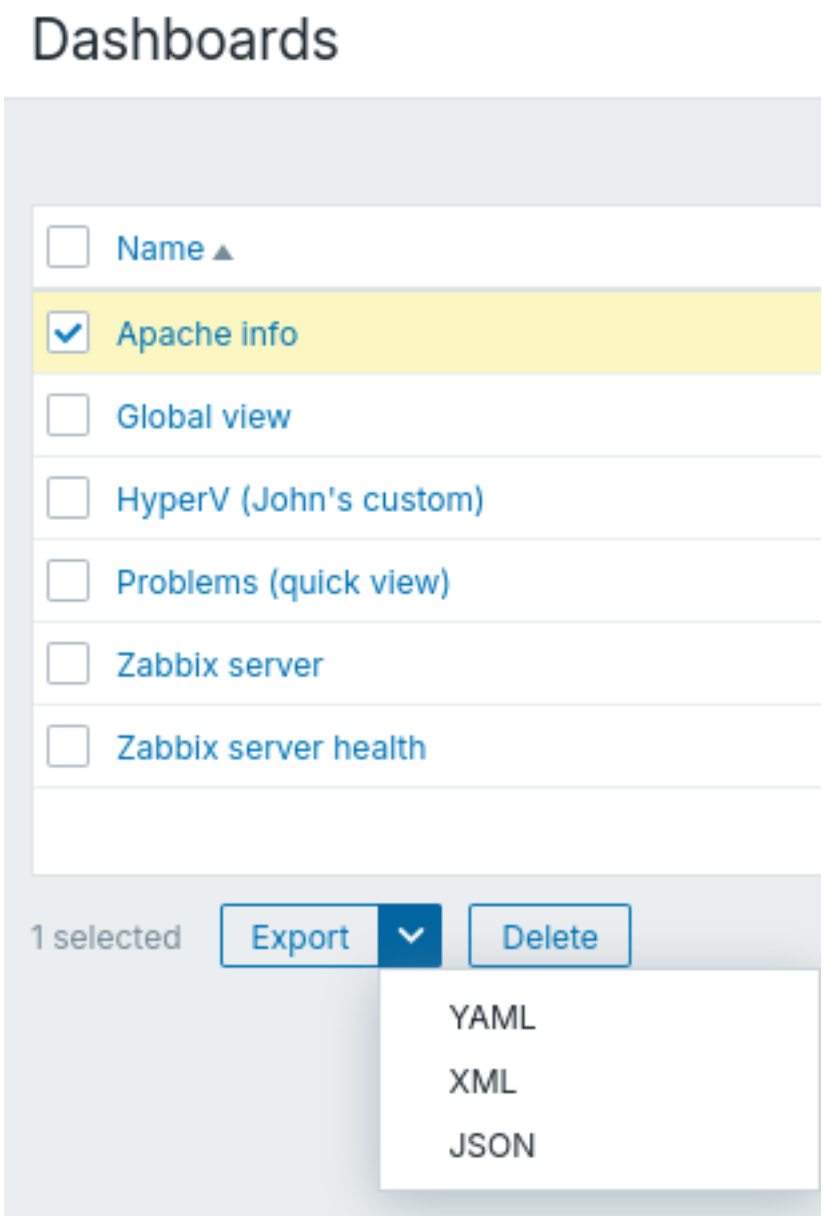
Resumen

Los dashboards se exportan con todos los objetos relacionados y las relaciones entre objetos.

Exportación

Para exportar dashboards, haga lo siguiente:

1. Vaya a *Dashboards* > *All dashboards*.
2. Marque las casillas de verificación de los dashboards que desea exportar.
3. Haga clic en *Export* debajo de la lista para exportar en el formato YAML predeterminado, o haga clic en la flecha para seleccionar un formato de exportación diferente.



Según el formato seleccionado, los dashboards se exportan a un archivo local con un nombre predeterminado:

- `zbx_export_dashboards.yaml` - en la exportación YAML;
- `zbx_export_dashboards.xml` - en la exportación XML;
- `zbx_export_dashboards.json` - en la exportación JSON.

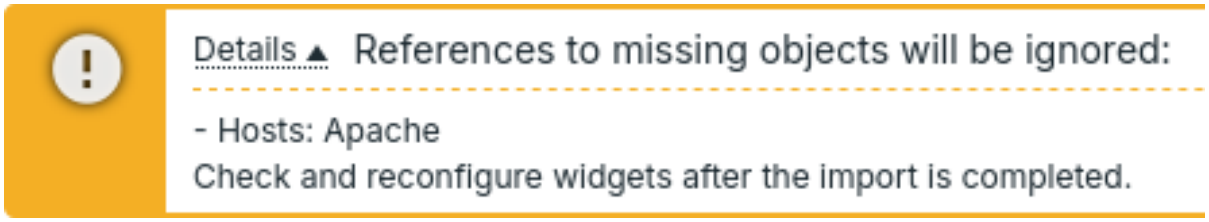
Importación

Para importar dashboards, siga estos pasos:

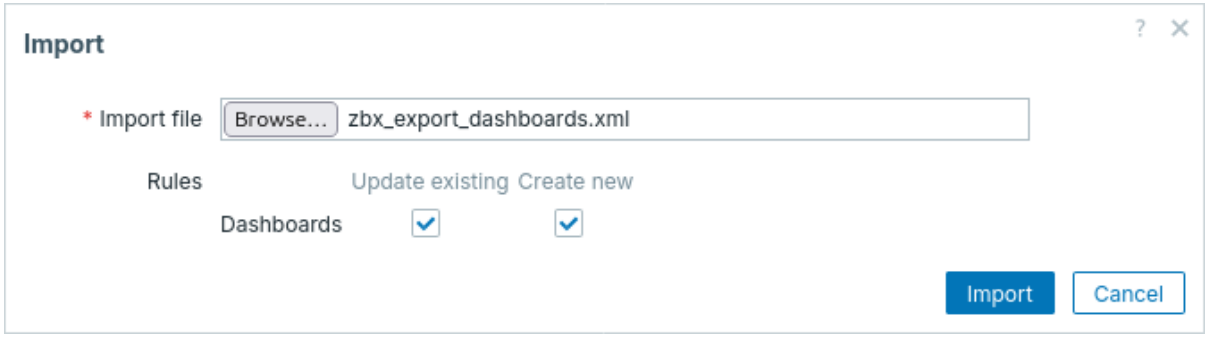
1. Vaya a *Dashboards* > *All dashboards*.
2. Haga clic en *Import* en la esquina superior derecha. Esta opción está deshabilitada si el usuario no tiene permiso para *Create and edit dashboards*.

3. Seleccione el archivo de importación.
4. Marque las casillas de las reglas de importación que desea aplicar (consulte más abajo).
5. Haga clic en *Import* en la esquina inferior derecha del formulario de configuración.

Si el archivo de importación hace referencia a objetos que no existen en el sistema de destino, aparece un mensaje de alerta. Esto es normal en el caso de widgets que hacen referencia a hosts, items u otros objetos específicos. En estos casos, Zabbix permite importar el dashboard, elimina las referencias que faltan y marca el widget como no configurado completamente cuando los datos son obligatorios, o elimina los criterios de filtrado cuando los campos se utilizan únicamente para filtrar.



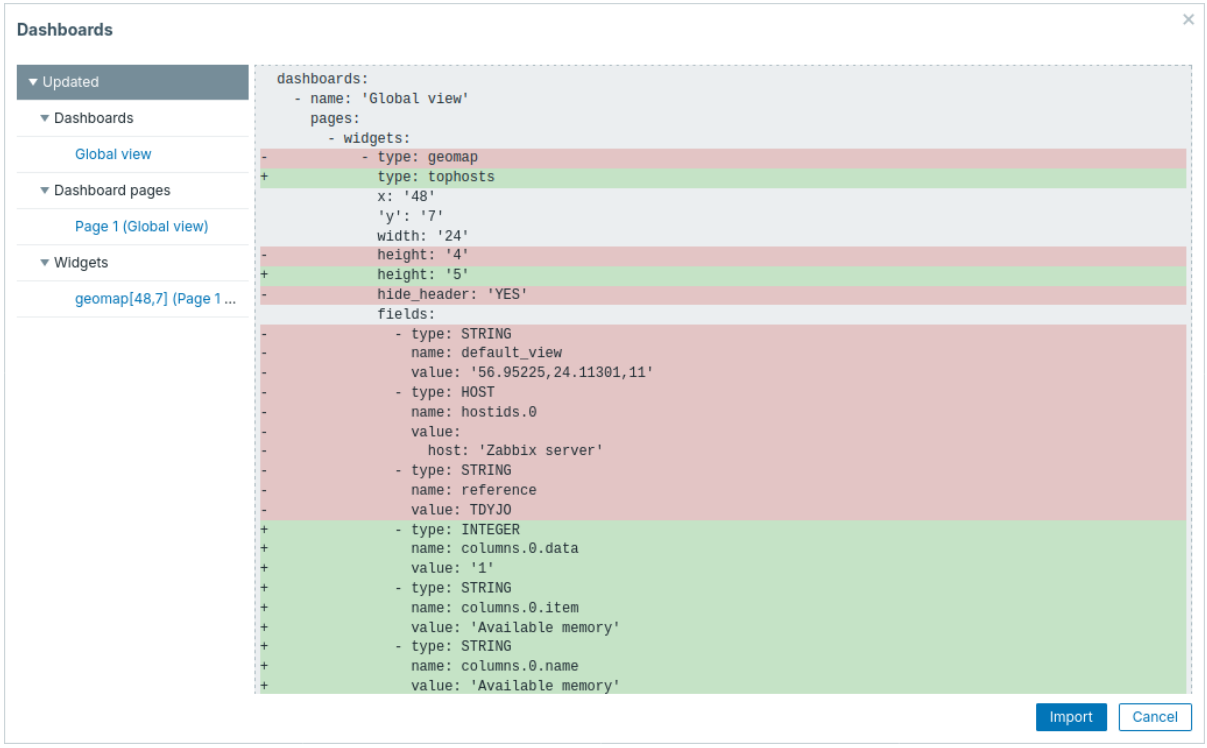
Al hacer clic de nuevo en el botón *Import*, se añadirá y/o actualizará el dashboard.



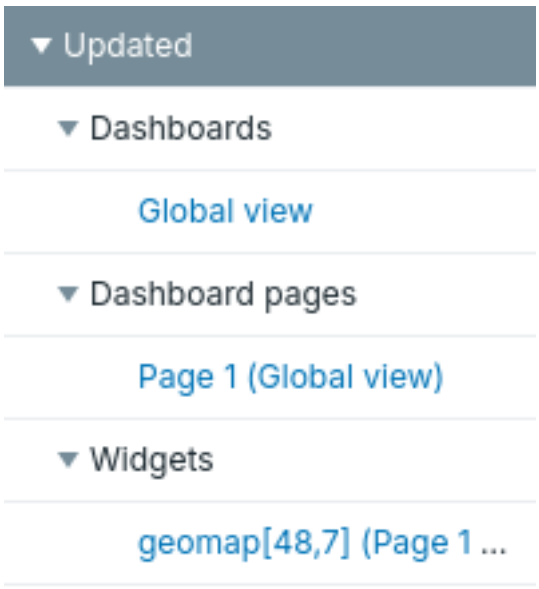
Reglas de importación:

Regla	Descripción
Update existing	Los elementos existentes se actualizarán utilizando los datos del archivo de importación. De lo contrario, no se actualizarán.
Create new	Los elementos nuevos se crearán utilizando los datos del archivo de importación. De lo contrario, no se crearán.

En la siguiente pantalla, podrá ver el contenido de los dashboards que se están importando. Si se trata de un dashboard nuevo, todos los elementos aparecerán en verde. Si se actualiza un dashboard existente, los elementos nuevos del dashboard se resaltarán en verde; los elementos eliminados del dashboard se resaltarán en rojo; los elementos que no hayan cambiado aparecerán sobre un fondo gris.



El menú de la izquierda se puede utilizar para navegar por la lista de cambios. La sección *Updated* resalta todos los cambios realizados en los elementos existentes del dashboard. La sección *Added* muestra los elementos nuevos del dashboard. Los elementos de cada sección se agrupan por tipo de elemento; haga clic en la flecha gris para expandir o contraer el grupo de elementos.



Revise los cambios del dashboard y, a continuación, haga clic en *Import* para realizar la importación del dashboard. En el frontend se mostrará un mensaje de éxito o de error de la importación.

Formato de exportación

Exportar a YAML:

```
zabbix_export:
  version: '8.0'
  dashboards:
    - name: 'Global view'
      pages:
        - widgets:
            - type: tophosts
```

```
            name: 'Top hosts by CPU utilization'
            width: '25'
            height: '4'
            fields:
              - type: INTEGER
                name: column
                value: '1'
              - type: INTEGER
                name: columns.0.data
                value: '2'
              - type: STRING
                name: columns.0.item
                value: 'System name'
              - type: STRING
                name: columns.0.name
                value: 'Host name'
              - type: STRING
                name: columns.1.base_color
                value: 4CAF50
              - type: INTEGER
                name: columns.1.data
                value: '1'
              - type: INTEGER
                name: columns.1.display
                value: '3'
              - type: STRING
                name: columns.1.item
                value: 'CPU utilization'
              - type: STRING
                name: columns.1.max
                value: '100'
              - type: STRING
                name: columns.1.min
                value: '0'
              - type: STRING
                name: columns.1.name
                value: Utilization
              - type: INTEGER
                name: columns.2.data
                value: '1'
              - type: STRING
                name: columns.2.item
                value: 'Load average (1m avg)'
              - type: STRING
                name: columns.2.name
                value: '1m avg'
              - type: INTEGER
                name: columns.3.data
                value: '1'
              - type: STRING
                name: columns.3.item
                value: 'Load average (5m avg)'
              - type: STRING
                name: columns.3.name
                value: '5m avg'
              - type: INTEGER
                name: columns.4.data
                value: '1'
              - type: STRING
                name: columns.4.item
                value: 'Load average (15m avg)'
              - type: STRING
```

```
            name: columns.4.name
            value: '15m avg'
          - type: INTEGER
            name: columns.5.data
            value: '1'
          - type: INTEGER
            name: columns.5.decimal_places
            value: '0'
          - type: STRING
            name: columns.5.item
            value: 'Number of processes'
          - type: STRING
            name: columns.5.name
            value: Processes
          - type: STRING
            name: columnsthresholds.1.color.0
            value: FFFF00
          - type: STRING
            name: columnsthresholds.1.color.1
            value: FF8000
          - type: STRING
            name: columnsthresholds.1.color.2
            value: FF4000
          - type: STRING
            name: columnsthresholds.1.threshold.0
            value: '50'
          - type: STRING
            name: columnsthresholds.1.threshold.1
            value: '80'
          - type: STRING
            name: columnsthresholds.1.threshold.2
            value: '90'
          - type: STRING
            name: reference
            value: NLVMO
      - type: hostavail
        'y': '4'
        width: '25'
        fields:
          - type: INTEGER
            name: interface_type.0
            value: '1'
      - type: problems
        name: 'Problemas actuales'
        'y': '6'
        width: '48'
        height: '5'
        fields:
          - type: STRING
            name: reference
            value: BRUQE
          - type: INTEGER
            name: show
            value: '3'
          - type: INTEGER
            name: show_tags
            value: '3'
      - type: systeminfo
        x: '25'
        width: '23'
        height: '4'
      - type: problemsbysv
```

```
      name: 'Problemas por severidad'
      x: '25'
      'y': '4'
      width: '23'
      fields:
        - type: STRING
          name: reference
          value: LBSVB
        - type: INTEGER
          name: show_type
          value: '1'
    - type: gauge
      name: 'Zabbix server'
      x: '48'
      width: '12'
      height: '4'
      hide_header: 'YES'
      fields:
        - type: INTEGER
          name: angle
          value: '270'
        - type: INTEGER
          name: desc_size
          value: '8'
        - type: ITEM
          name: itemid.0
          value:
            host: 'Zabbix server'
            key: vm.memory.utilization
        - type: STRING
          name: max
          value: '100'
        - type: STRING
          name: min
          value: '0'
        - type: INTEGER
          name: scale_size
          value: '11'
        - type: INTEGER
          name: show.0
          value: '1'
        - type: INTEGER
          name: show.1
          value: '2'
        - type: INTEGER
          name: show.2
          value: '3'
        - type: INTEGER
          name: show.3
          value: '4'
        - type: INTEGER
          name: show.4
          value: '5'
        - type: STRING
          name: thresholds.0.color
          value: 42A5F5
        - type: STRING
          name: thresholds.0.threshold
          value: '0'
        - type: STRING
          name: thresholds.1.color
          value: 9CCC65
```

```
        - type: STRING
          name: thresholds.1.threshold
          value: '20'
        - type: STRING
          name: thresholds.2.color
          value: FFEE58
        - type: STRING
          name: thresholds.2.threshold
          value: '40'
        - type: STRING
          name: thresholds.3.color
          value: FFA726
        - type: STRING
          name: thresholds.3.threshold
          value: '60'
        - type: STRING
          name: thresholds.4.color
          value: FF7043
        - type: STRING
          name: thresholds.4.threshold
          value: '80'
        - type: INTEGER
          name: th_arc_size
          value: '9'
        - type: INTEGER
          name: th_show_arc
          value: '1'
        - type: INTEGER
          name: th_show_labels
          value: '1'
        - type: INTEGER
          name: units_bold
          value: '1'
        - type: INTEGER
          name: units_size
          value: '24'
        - type: INTEGER
          name: value_arc_size
          value: '30'
        - type: INTEGER
          name: value_bold
          value: '1'
        - type: INTEGER
          name: value_size
          value: '24'
    - type: svggraph
      x: '48'
      'y': '4'
      width: '24'
      height: '3'
      hide_header: 'YES'
      fields:
        - type: INTEGER
          name: axisx
          value: '0'
        - type: STRING
          name: ds.0.color
          value: AB47BC
        - type: INTEGER
          name: ds.0.fill
          value: '0'
        - type: STRING
```

```
            name: ds.0.hosts.0
            value: 'Zabbix server'
          - type: STRING
            name: ds.0.items.0
            value: 'Available memory'
          - type: INTEGER
            name: ds.0.transparency
            value: '2'
          - type: INTEGER
            name: ds.0.type
            value: '2'
          - type: INTEGER
            name: ds.0.width
            value: '2'
          - type: INTEGER
            name: ds.1.axisy
            value: '1'
          - type: STRING
            name: ds.1.color
            value: 42A5F5
          - type: INTEGER
            name: ds.1.fill
            value: '0'
          - type: STRING
            name: ds.1.hosts.0
            value: 'Zabbix server'
          - type: STRING
            name: ds.1.items.0
            value: 'CPU utilization'
          - type: INTEGER
            name: ds.1.transparency
            value: '0'
          - type: INTEGER
            name: ds.1.type
            value: '2'
          - type: INTEGER
            name: ds.1.width
            value: '2'
          - type: STRING
            name: reference
            value: TOYGS
      - type: geomap
        x: '48'
        'y': '7'
        width: '24'
        height: '4'
        hide_header: 'YES'
        fields:
          - type: STRING
            name: default_view
            value: '56.95225,24.11301,11'
          - type: HOST
            name: hostids.0
            value:
              host: 'Zabbix server'
          - type: STRING
            name: reference
            value: TDYJO
      - type: clock
        name: 'Hora local'
        x: '60'
        width: '12'
```

```
            hide_header: 'YES'
            fields:
              - type: INTEGER
                name: clock_type
                value: '1'
              - type: INTEGER
                name: show.0
                value: '1'
              - type: INTEGER
                name: show.1
                value: '2'
              - type: INTEGER
                name: show.2
                value: '3'
              - type: INTEGER
                name: time_sec
                value: '0'
              - type: INTEGER
                name: tzone_format
                value: '1'
          - type: item
            name: 'Rendimiento del server'
            x: '60'
            'y': '2'
            width: '12'
            hide_header: 'YES'
            fields:
              - type: INTEGER
                name: aggregate_function
                value: '2'
              - type: INTEGER
                name: decimal_size
                value: '32'
              - type: STRING
                name: description
                value: 'Values per second'
              - type: INTEGER
                name: desc_h_pos
                value: '0'
              - type: INTEGER
                name: desc_v_pos
                value: '0'
              - type: ITEM
                name: itemid.0
                value:
                  host: 'Zabbix server'
                  key: 'zabbix[wcache,values]'
              - type: INTEGER
                name: show.0
                value: '1'
              - type: INTEGER
                name: show.1
                value: '2'
              - type: INTEGER
                name: show.2
                value: '5'
              - type: INTEGER
                name: sparkline.fill
                value: '2'
              - type: INTEGER
                name: sparkline.width
                value: '2'
```

```
            - type: STRING
              name: thresholds.0.color
              value: DCEDC8
            - type: STRING
              name: thresholds.0.threshold
              value: '20'
            - type: STRING
              name: thresholds.1.color
              value: FFF9C4
            - type: STRING
              name: thresholds.1.threshold
              value: '40'
            - type: STRING
              name: thresholds.2.color
              value: FFCCBC
            - type: STRING
              name: thresholds.2.threshold
              value: '60'
            - type: INTEGER
              name: units_bold
              value: '0'
            - type: INTEGER
              name: units_show
              value: '0'
            - type: INTEGER
              name: value_h_pos
              value: '0'
            - type: INTEGER
              name: value_size
              value: '32'
```

Elementos exportados

Los elementos exportados se explican en la tabla siguiente.

Element	Type	Description
name	string	(required) Nombre del dashboard.
pages		Elemento raíz para las páginas del dashboard
widgets		Elemento raíz para las páginas del dashboard

Note:
Véase también: Dashboard object (consulte la propiedad correspondiente con el mismo nombre).

Widget del dashboard

Element	Type	Description
type	string	(obligatorio) Tipo del widget del dashboard. Valores posibles: actionlog - Registro de acciones; clock - Reloj; discovery - Estado de descubrimiento; favgraphs - Gráficos favoritos; favmaps - Mapas favoritos; gauge - Medidor; geomap - Geomap; graph - Gráfico (clásico); graphprototype - Prototipo de gráfico; honeycomb - Honeycomb; hostavail - Disponibilidad del host; hostcard - Tarjeta del host; hostnavigator - Navegador de hosts; itemcard - Tarjeta del item; itemhistory - Historial del item; itemnavigator - Navegador de items; item - Valor del item; map - Mapa; navtree - Árbol de navegación del mapa; piechart - Gráfico circular; problemhosts - Hosts con problemas; problems - Problemas; problemsbysv - Problemas por severidad; scatterplot - Diagrama de dispersión; slareport - Informe de SLA; svggraph - Gráfico; systeminfo - Información del sistema; tophosts - Hosts principales; topitems - Items principales; toptriggers - Triggers principales; trigover - Resumen de triggers; url - URL; web - Supervisión web.
x	integer	Una posición horizontal desde el lado izquierdo del dashboard. Valores posibles:[1] 0-71.
y	integer	Una posición vertical desde la parte superior del dashboard. Valores posibles:[1] 0-63.
name	string	Nombre personalizado del widget.
width	integer	El ancho del widget. Valores posibles:[1] 1-72.
height	integer	La altura del widget. Valores posibles:[1] 1-64.
hide_header	string	Ocultar el encabezado del widget. Valores posibles:[1] NO (0, predeterminado), YES (1).
fields		Elemento raíz para los campos del widget del dashboard.

Note:
Véase también: Widget del dashboard (consulte la propiedad correspondiente con el mismo nombre).

Campo de widget del dashboard

Element	Type	Description
type	string	(required) Tipo de campo del widget. Valores posibles:[1] INTEGER (0), STRING (1), HOST_GROUP (2), HOST (3), ITEM (4), ITEM_PROTOTYPE (5), GRAPH (6), GRAPH_PROTOTYPE (7), MAP (8), SERVICE (9), SLA (10), USER (11), ACTION (12), MEDIA_TYPE (13).

Element	Type	Description
name	string	(required) Nombre del campo del widget.
value	mixed	(required) Valor del campo del widget, según el tipo de campo.
host	string	Host del item.
key	string	Clave del item.

Note:
Véase también: Campo de widget del dashboard (consulte la propiedad correspondiente con el nombre coincidente).

Notas al pie

[1] Los valores enteros de la API entre paréntesis, por ejemplo, ENABLED (0), se mencionan solo como referencia. Para obtener más información, consulte la página del objeto API enlazada en la entrada de la tabla o al final de cada sección.

Expresiones regulares

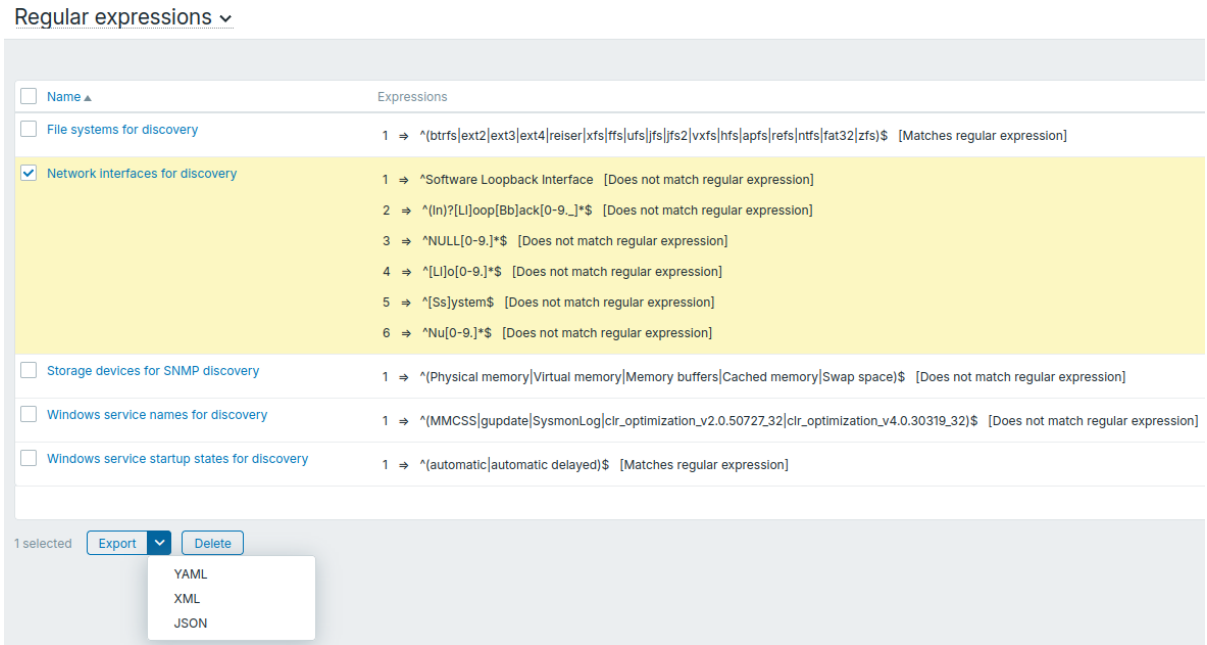
Resumen

Las expresiones regulares se exportan con todos los objetos relacionados y las relaciones entre objetos.

Exportación

Para exportar expresiones regulares, haga lo siguiente:

1. Vaya a *Administration* > *General* > *Regular expressions*.
2. Marque las casillas de verificación de las expresiones regulares que desea exportar.
3. Haga clic en *Export* debajo de la lista para exportar en el formato YAML predeterminado, o haga clic en la flecha para seleccionar un formato de exportación diferente.



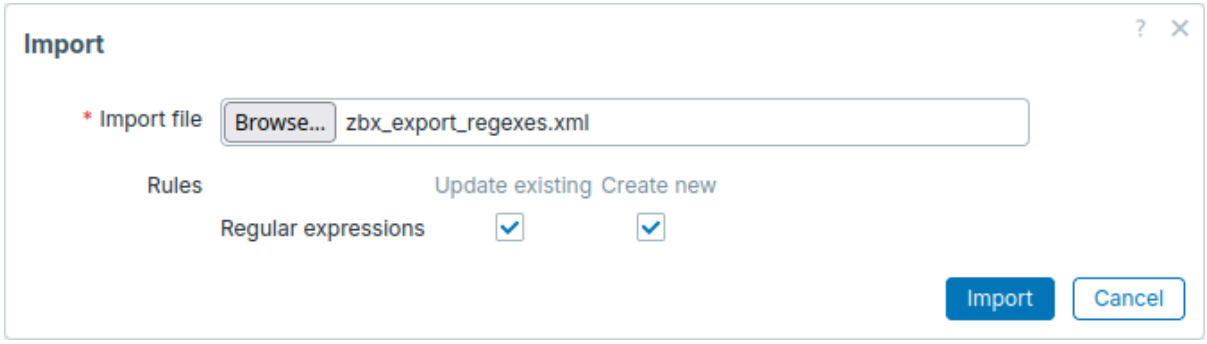
Según el formato seleccionado, las expresiones regulares se exportan a un archivo local con un nombre predeterminado:

- `zbx_export_regexes.yaml` - en exportación YAML;
- `zbx_export_regexes.xml` - en exportación XML;
- `zbx_export_regexes.json` - en exportación JSON.

Importación

Para importar expresiones regulares, haga lo siguiente:

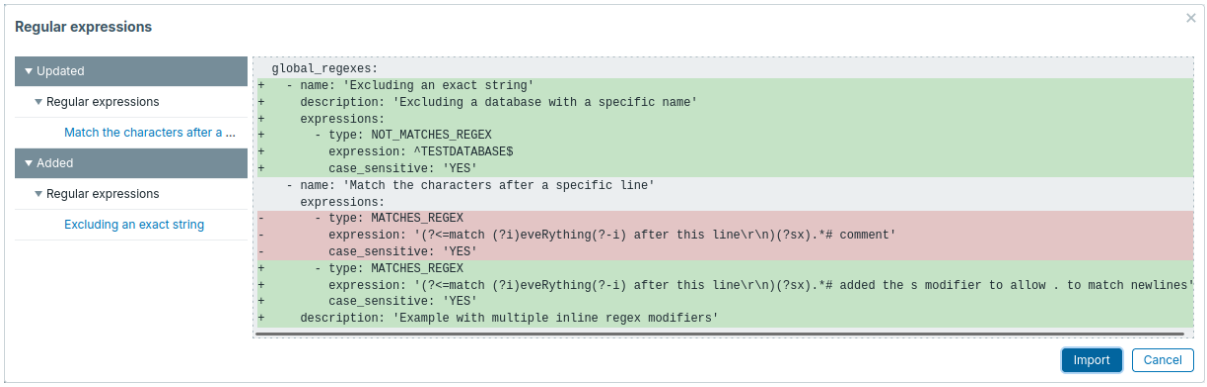
1. Vaya a *Administración* > *General* > *Expresiones regulares*.
2. Haga clic en *Importar* en la esquina superior derecha.
3. Seleccione el archivo de importación.
4. Marque las casillas de las reglas de importación que desea aplicar (consulte más abajo).
5. Haga clic en *Importar* en la esquina inferior derecha del formulario de configuración.



Reglas de importación:

Regla	Descripción
Actualizar existentes	Las expresiones regulares existentes se actualizarán utilizando los datos del archivo de importación, sobrescribiendo por completo las expresiones existentes con el mismo nombre. De lo contrario, no se actualizarán.
Crear nuevas	Las nuevas expresiones regulares se crearán utilizando los datos del archivo de importación. De lo contrario, no se crearán.

En la siguiente pantalla, podrá ver el contenido de las expresiones regulares que se están importando. Si se trata de una expresión regular nueva, todos los datos aparecerán en verde. Si se actualiza una expresión regular existente, los datos nuevos de la expresión regular se resaltarán en verde; los datos eliminados se resaltarán en rojo; los datos sin cambios se mostrarán sobre un fondo gris.



El menú de la izquierda se puede utilizar para navegar por la lista de cambios. La sección *Actualizadas* resalta todos los cambios realizados en las expresiones regulares existentes. La sección *Añadidas* muestra las nuevas expresiones regulares. Haga clic en la flecha para expandir o contraer un grupo de expresiones regulares.

Revise los cambios de las expresiones regulares y, a continuación, haga clic en *Importar* para realizar la importación de las expresiones regulares. En el frontend se mostrará un mensaje de éxito o de error de la importación.

Formato de exportación

Exportar a YAML:

```yaml
zabbix_export:
  version: '8.0'
  global_regexes:
    - name: 'File systems for discovery'
      description: 'Expresión regular global utilizada para descubrir e identificar sistemas de archivos e
      expressions:
        - type: MATCHES_REGEX
          expression: ^(btrfs|ext2|ext3|ext4|reiser|xfs|ffs|ufs|jfs|jfs2|vxfs|hfs|apfs|refs|ntfs|fat32|zfs
    - name: 'Storage devices for SNMP discovery'
      expressions:
        - type: NOT_MATCHES_REGEX
          expression: '^(Physical memory|Virtual memory|Memory buffers|Cached memory|Swap space)$'
          case_sensitive: 'YES'
```

Elementos exportados

Elementos exportados de expresiones regulares:

Element	Type	Description
global_regexes	object	Elemento raíz para expresiones regulares.
name	string	(required) Nombre de la expresión regular.
description	text	Descripción de la expresión regular.
expressions	object	Elemento raíz para expresiones regulares.
type	string	Tipo de la expresión regular.
expression	string	Expresión regular.
exp_delimiter	string	Delimitador de la expresión regular; presente cuando el tipo es *Contains any substring from list*.
case_sensitive	string	Opción de distinción entre mayúsculas y minúsculas para la expresión regular.

Note:
Véase también: Regular expression object (consulte la propiedad correspondiente con el mismo nombre).

Expresiones regulares

Resumen Perl Compatible Regular Expressions (PCRE, PCRE2) son compatibles en Zabbix.

Hay dos formas de usar expresiones regulares en Zabbix:

- Introducir manualmente una expresión regular
- Usar una expresión regular global creada en Zabbix

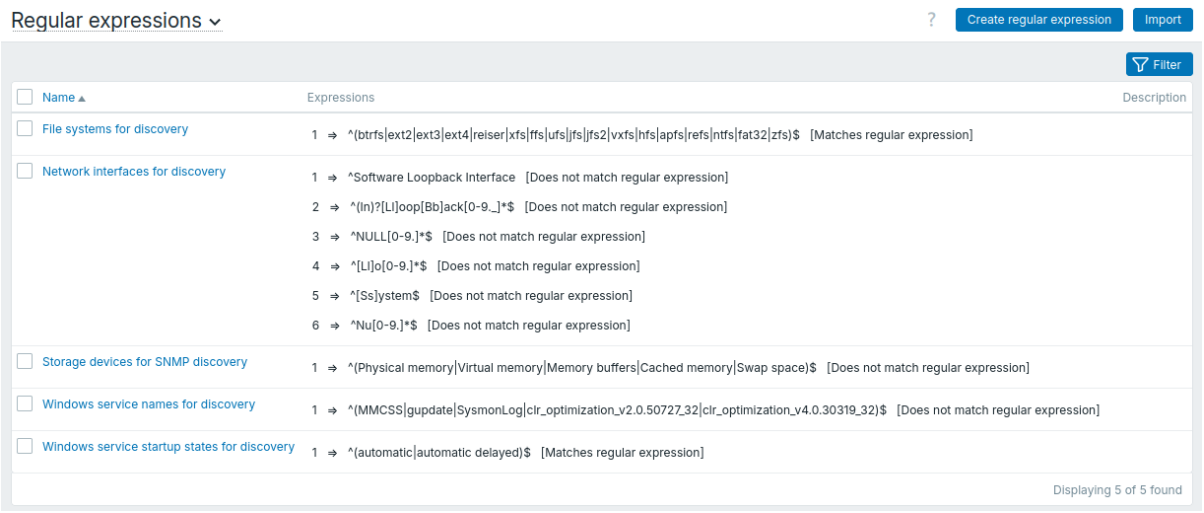
Expresiones regulares Puede introducir manualmente una expresión regular en los lugares compatibles.
 Tenga en cuenta que la expresión no debe comenzar con @, ya que este símbolo se usa en Zabbix para hacer referencia a expresiones regulares globales.

Attention:
Es posible quedarse sin memoria de pila al usar expresiones regulares. Consulte PCRE PERFORMANCE para obtener más información.

Tenga en cuenta que, en la coincidencia multilínea, los anclajes ^ y $ coinciden al principio y al final de cada línea, respectivamente, en lugar de al principio y al final de toda la cadena.

Consulte también ejemplos de escape correcto en varios contextos.

Expresiones regulares globales Zabbix proporciona un editor avanzado en el frontend para crear y probar expresiones regulares complejas.
 Para acceder a él, vaya a *Administration* > *General* > *Regular expressions*.
 Zabbix incluye un número de expresiones regulares globales predefinidas en su conjunto de datos predeterminado.

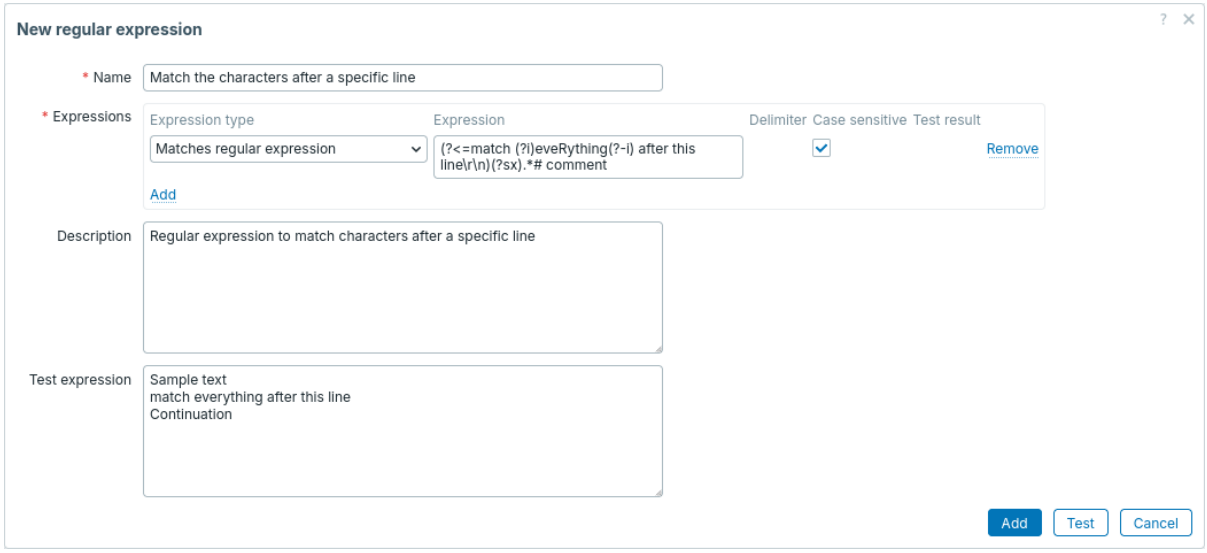


Puede filtrar la lista de expresiones regulares por nombre y descripción para encontrar las expresiones que necesita.

Configuración

Para crear una expresión regular global:

1. Haga clic en *Crear expresión regular*.
2. Introduzca los parámetros de la expresión regular en el formulario.



Todos los campos de entrada obligatorios están marcados con un asterisco rojo.

Parámetro	Descripción
Nombre	Establezca el nombre de la expresión regular. Se permite cualquier carácter Unicode.
Expresiones	Haga clic en *Añadir* en la sección *Expresiones* para añadir una nueva subexpresión.
Tipo de expresión	Seleccione el tipo de expresión: **Contiene una cadena**: coincide con la subcadena. **Contiene cualquier subcadena de la lista**: coincide con cualquier subcadena de una lista delimitada. La lista delimitada incluye una coma `,`, un punto `.` o una barra diagonal `/`. **No contiene una cadena**: coincide con cualquier cadena excepto la subcadena. **Coincide con una expresión regular**: coincide con la expresión regular. **No coincide con una expresión regular**: no coincide con la expresión regular.
Expresión	Introduzca la subcadena o expresión regular. Longitud máxima: 2048 caracteres.
Delimitador	Una coma `,`, un punto `.` o una barra diagonal `/` para separar cadenas de texto en una expresión regular. Este parámetro solo está activo cuando se selecciona el tipo de expresión *Contiene cualquier subcadena de la lista*.
Distingue mayúsculas y minúsculas	Una casilla de verificación para especificar si la expresión regular distingue entre mayúsculas y minúsculas.
Descripción	Introduzca una descripción para la expresión regular.
Probar expresión	Un valor que se utilizará para probar la expresión regular.

Una barra diagonal `/` en la expresión se trata literalmente y no como un delimitador. Esto permite guardar expresiones que contienen una barra sin errores.

Attention:
El nombre de una expresión regular personalizada en Zabbix puede contener comas, espacios, etc. En los casos en los que esto pueda provocar una interpretación incorrecta al hacer referencia a ella (por ejemplo, una coma en el parámetro de una clave de item), la referencia completa puede encerrarse entre comillas, como en este ejemplo: *"@Mi expresión regular personalizada para propósito1, propósito2"*.
 Los nombres de expresiones regulares no deben escribirse entre comillas en otras ubicaciones (por ejemplo, en las propiedades de una regla de LLD).

Una vez creada una expresión regular, puede utilizarse en varios lugares del frontend haciendo referencia a su nombre, precedido por `@`, por ejemplo, `@mycustomregexp`.

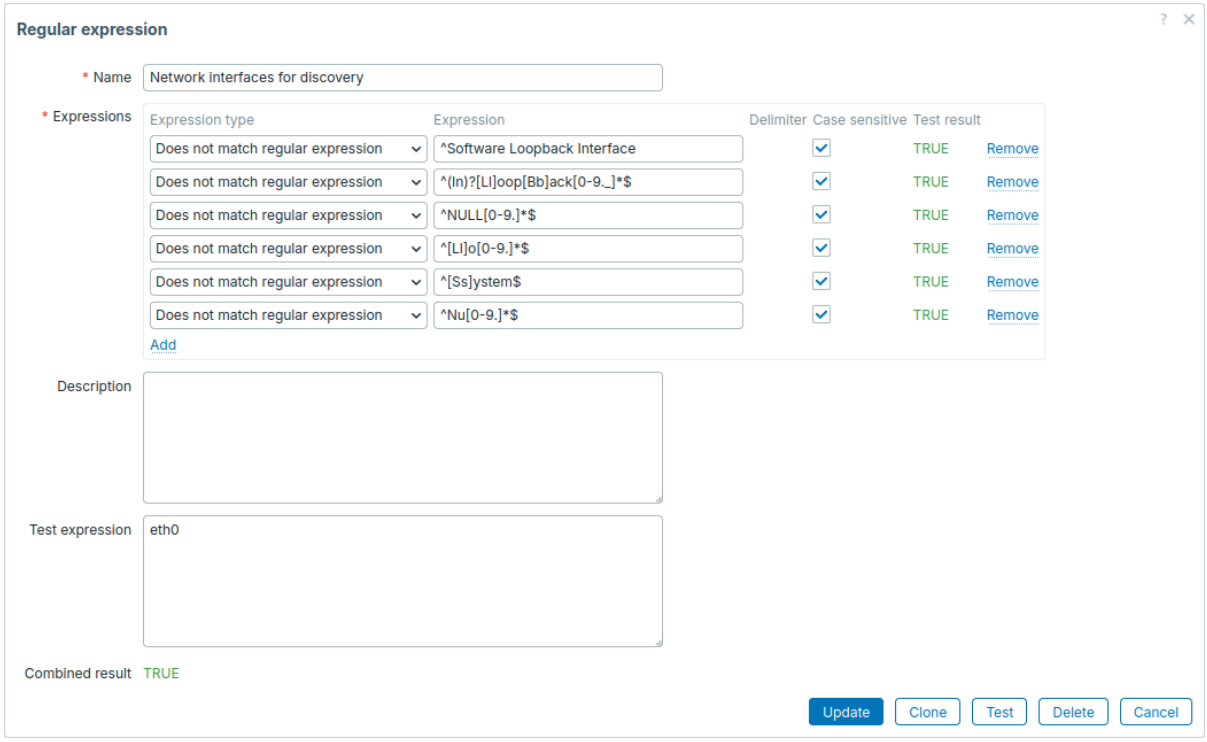
Botones del formulario

Los botones en la parte inferior del formulario le permiten realizar varias operaciones:

Button	Description
Add	Agregar una expresión regular. Este botón solo está disponible para nuevas expresiones regulares.
Update	Actualizar las propiedades de una expresión regular.
Clone	Crear otra expresión regular basada en las propiedades de la expresión regular actual.
Test	Probar una expresión regular y sus subexpresiones con un valor proporcionado.
Delete	Eliminar la expresión regular.
Cancel	Cancelar la edición de las propiedades de la expresión regular.

Prueba de una expresión regular

Puede probar la expresión regular y sus subexpresiones introduciendo un valor en el campo *Test expression* y haciendo clic en el botón *Test*.



En la columna *Test result*, se muestra el estado de cada subexpresión. El campo *Combined result* muestra el estado general de la expresión regular. Si se definen varias subexpresiones, Zabbix usa el operador lógico AND para calcular *Combined result*; es decir, si al menos un resultado es `False`, *Combined result* también es `False`.

Expresiones regulares globales predeterminadas

El conjunto de datos predeterminado incluye las siguientes expresiones regulares globales predefinidas:

Name	Expression	Matches
File systems for discovery	`^(btrfs|ext2|ext3|ext4|reiser|xfs|ffs|ufs|jfs|jfs2|vxfs|hfs|apfs|refs|ntfs|fat32|zfs)$`	”btrfs” o ”ext2” o ”ext3” o ”ext4” o ”reiser” o ”xfs” o ”ffs” o ”ufs” o ”jfs” o ”jfs2” o ”vxfs” o ”hfs” o ”apfs” o ”refs” o ”ntfs” o ”fat32” o ”zfs”.
Network interfaces for discovery	`^Software Loopback Interface`	Cadenas que comienzan con ”Software Loopback Interface”.
	`^lo$`	”lo”.
	`^(In)?[Ll]oop[Bb]ack[0-9._]*$`	Cadenas que opcionalmente comienzan con ”In”, seguidas de ”L” o ”l”, seguidas de ”oop”, seguidas de ”B” o ”b”, seguidas de ”ack”, y opcionalmente seguidas por cualquier número de dígitos, puntos o guiones bajos.

Name	Expression	Matches
	`^NULL[0-9.]*$`	Cadenas que comienzan con ”NULL” y opcionalmente van seguidas de cualquier número de dígitos o puntos.
	`^[Ll]o[0-9.]*$`	Cadenas que comienzan con ”Lo” o ”lo” y opcionalmente van seguidas de cualquier número de dígitos o puntos.
	`^[Ss]ystem$`	”System” o ”system”.
	`^Nu[0-9.]*$`	Cadenas que comienzan con ”Nu” y opcionalmente van seguidas de cualquier número de dígitos o puntos.
Storage devices for SNMP discovery	`^(Physical memory|Virtual memory|Memory buffers|Cached memory|Swap space)$`	”Physical memory” o ”Virtual memory” o ”Memory buffers” o ”Cached memory” o ”Swap space”.
Windows service names for discovery	`^(MMCSS|gupdate|SysmonLog|clr_optimization_v2.0.50727_32|clr_optimization_v4.0.30319_32)$`	”MMCSS” o ”gupdate” o ”SysmonLog” o cadenas que coinciden con ”clr_optimization_v2.0.50727_32” o ”clr_optimization_v4.0.30319_32”, donde cada punto puede ser reemplazado por cualquier carácter excepto un salto de línea.
Windows service startup states for discovery	`^(automatic|automatic delayed)$`	”automatic” o ”automatic delayed”.

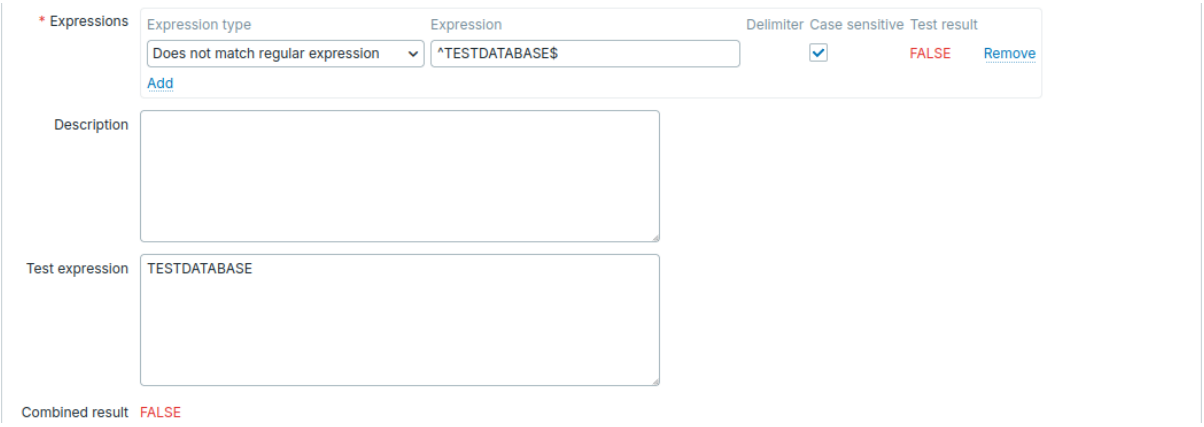
Importación/exportación de una expresión regular

Puede importar o exportar expresiones regulares globales como se describe en la página de configuración de exportación/importación de Regular expressions.

Ejemplos Ejemplo de exclusión de una cadena exacta

Use la siguiente expresión en el descubrimiento de bajo nivel para descubrir bases de datos, excepto una base de datos con un nombre específico:

```
^TESTDATABASE$
```

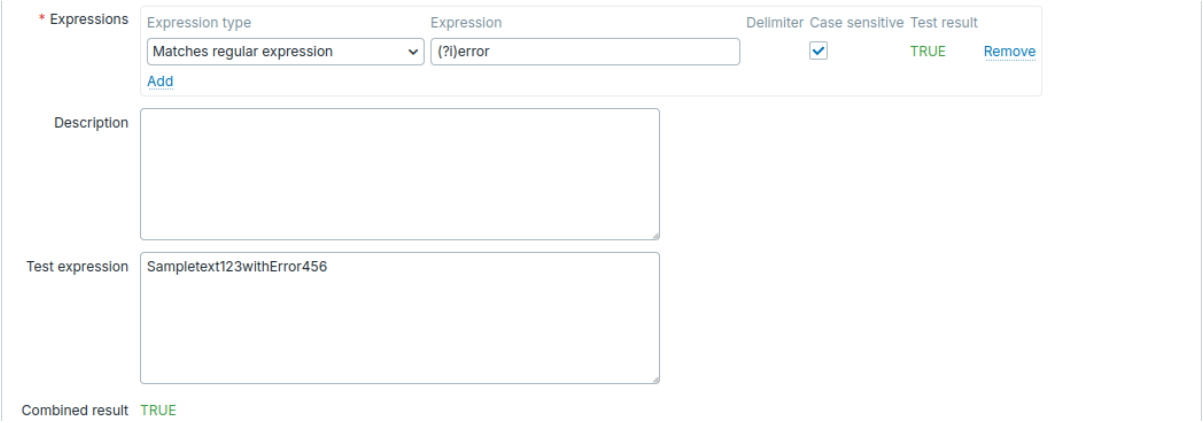


Tanto *Test result* como *Combined result* muestran FALSE: el nombre coincide con ”TESTDATABASE”, por lo que la condición *Does not match regular expression* no se cumple.

Ejemplo con un modificador de expresión regular en línea para una coincidencia que no distingue mayúsculas y minúsculas

Use la siguiente expresión regular, que incluye el modificador en línea `(?i)` para realizar una coincidencia de los caracteres ”error” sin distinguir mayúsculas y minúsculas:

```
(?i)error
```



Tanto *Test result* como *Combined result* muestran TRUE: los caracteres "error" coinciden incluso con *Case sensitive* seleccionado, porque el modificador en línea `(?i)` hace que la coincidencia no distinga mayúsculas y minúsculas.

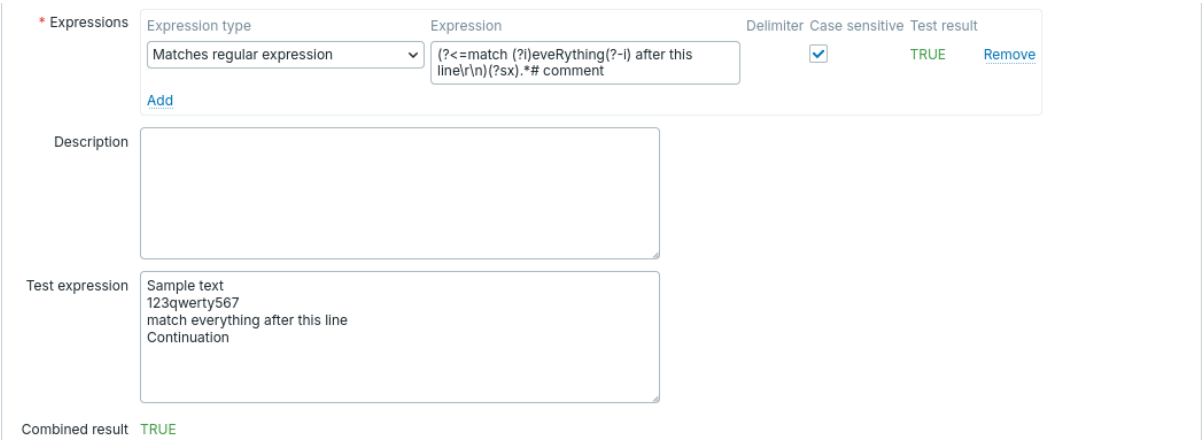
Ejemplo con múltiples modificadores inline de expresiones regulares

Use la siguiente expresión regular, que incluye varios modificadores inline, para coincidir con los caracteres después de una línea específica:

```
(?<=match (?i)eveRything(?-i) after this line\r\n)(?sx).*# comment
```

donde:

- `(?i)` - activa la coincidencia sin distinción entre mayúsculas y minúsculas;
- `(?-i)` - desactiva la coincidencia sin distinción entre mayúsculas y minúsculas;
- `(?s)` - habilita el modo dotall, permitiendo que `.` coincida con caracteres de nueva línea;
- `(?x)` - habilita el modo extendido, permitiendo que los espacios en blanco y los comentarios en la expresión regular se ignoren.



Tanto *Test result* como *Combined result* muestran TRUE: los caracteres que siguen a la línea específica coinciden.

Attention:
El modificador g no se puede especificar inline. La lista de modificadores disponibles se puede encontrar en la página del manual pcresyntax. Para obtener más información sobre la sintaxis de PCRE, consulte la documentación HTML de PCRE.

Compatibilidad de expresiones regulares por ubicación

Ubicación		Expresión regular	Expresión regular global	Coincidencia multilínea	Comentarios
Items de agent					
	eventlog[]	Sí	Sí	Sí	Parámetros `regexp`, `severity`, `source`, `eventid`
	eventlog.count[]				Parámetros `regexp`, `severity`, `source`, `eventid`
	log[]				Parámetro `regexp`

Ubicación		Expresión regular	Expresión regular global	Coincidencia multilínea	Comentarios
	log.count[]				
	logrt[]		Sí/No		El parámetro `regexp` admite ambos; el parámetro `file_regexp` solo admite expresiones no globales
	logrt.count[]				
	proc.cpu.util[]		No	No	Parámetro `cmdline`
	proc.get[]				
	proc.mem[]				
	proc.num[]				
	sensor[]				Parámetros `device` y `sensor` en Linux 2.4
	system.hw.macaddr[]				Parámetro `interface`
	system.sw.packages[]				Parámetro `regexp`
	system.sw.packages.get[]				Parámetro `regexp`
	vfs.dir.count[]				Parámetros `regex_incl`, `regex_excl`, `regex_excl_dir`
	vfs.dir.get[]				Parámetros `regex_incl`, `regex_excl`, `regex_excl_dir`
	vfs.dir.size[]				Parámetros `regex_incl`, `regex_excl`, `regex_excl_dir`
	vfs.file.regexp[]			Sí	Parámetro `regexp`
	vfs.file.regmatch[]				
	web.page.regexp[]				
Traps SNMP					
	snmptrap[]	Sí	Sí	No	Parámetro `regexp`
Preprocesamiento del valor del item		Sí	No	No	Parámetro `pattern`
Funciones para trig-gers/items calcula-dos					
	count()	Sí	Sí	Sí	Parámetro `pattern` si el parámetro `operator` es *regexp* o *iregexp*
	countunique()	Sí	Sí		
	find()	Sí	Sí		
	logeventid()	Sí	Sí	No	Parámetro `pattern`
	logsource()				
Descubrimiento de bajo nivel					
	Filters	Sí	Sí	No	Campo *Regular expression*
	Overrides	Sí	No		En las opciones *matches*, *does not match* de las condiciones de *Operation*
Condiciones de acción		Sí	No	No	En las opciones *matches*, *does not match* de las condiciones de autorregistro de *Host name* y *Host metadata*
Scripts		Sí	Sí	No	Campo *Input validation rule*
Monitorización web		Sí	No	Sí	*Variables* con el prefijo **regex:** Campo *Required string*

Ubicación		Expresión regular	Expresión regular global	Coincidencia multilínea	Comentarios
Contexto de macro de usuario Funciones de macro		Sí	No	No	En el contexto de macro con el prefijo **regex:**
	regsub() iregsub()	Sí	No	No	Parámetro `pattern`
Indicadores de enlace en mapas		Sí	No	No	Campo *Pattern* (para items de texto)
Asignación de iconos		Sí	Sí	No	Campo *Expression*
Asignación de valores		Sí	No	No	Campo *Value* si el tipo de asignación es `regexp`
Parámetros de configu-ración de agent					
	AllowKeyRegexp DenyKeyRegexp	Sí	No	No	Patrón

Eventos y resolución

Resumen Hay varios tipos de eventos generados en Zabbix:

- eventos de trigger - cada vez que un trigger cambia su estado (*OK→PROBLEM→OK*)
- eventos de servicio - cada vez que un servicio cambia su estado (*OK→PROBLEM→OK*)
- eventos de descubrimiento - cuando se detectan hosts o servicios
- eventos de autorregistro - cuando los agents activos se registran automáticamente por el server
- eventos internos - cuando un item/regla de descubrimiento de bajo nivel deja de ser compatible o un trigger entra en un estado desconocido

Los eventos tienen marca de tiempo y pueden ser la base de acciones como enviar correos de notificación, crear tickets o ejecutar comandos remotos.

Para ver los detalles del evento en el frontend, vaya a *Monitoring* > *Problems*. Allí puede hacer clic en la fecha y hora del evento para abrir la página de detalles del evento, donde puede ver información adicional sobre el evento, incluido si fue suprimido por mantenimiento o suprimido manualmente por un usuario.

Resolución de problemas Los eventos de problema creados por triggers se resuelven automáticamente si la condición que causó el problema ya no existe.

Para añadir flexibilidad y comodidad a la gestión de eventos de problema, existen las siguientes opciones:

- Correlación de eventos
- Reconocimiento y supresión de eventos
- Cierre manual de problemas
- Períodos de mantenimiento - definen períodos en los que los problemas se suprimen
- Etiquetado - etiquetar problemas por una característica común

Eventos de trigger El cambio de estado del trigger es la fuente de eventos más frecuente y más importante. Cada vez que el trigger cambia de estado, se genera un evento. El evento contiene detalles del cambio de estado del trigger: cuándo ocurrió y cuál es el nuevo estado.

Los triggers crean dos tipos de eventos: `Problem` y `OK`.

Eventos de problema

Se crea un evento de problema:

- cuando una expresión de trigger se evalúa como TRUE si el trigger está en estado OK;
- cada vez que una expresión de trigger se evalúa como TRUE si la generación de múltiples eventos de problema está habilitada para el trigger.

Eventos OK

Un evento OK cierra el o los eventos de problema relacionados y puede ser creado por 3 componentes:

- triggers - según la configuración de ”generación de eventos OK” y ”los eventos OK cierran”;
- correlación de eventos
- gestor de tareas - cuando un evento se cierra manualmente

Los triggers tienen una configuración de ”OK event generation” que controla cómo se generan los eventos OK:

- *Expression* - se genera un evento OK para un trigger en estado de problema cuando su expresión evalúa a FALSE. Esta es la configuración más simple y está habilitada de forma predeterminada.
- *Recovery expression* - se genera un evento OK para un trigger en estado de problema cuando su expresión evalúa a FALSE y la expresión de recuperación evalúa a TRUE. Esto puede usarse si el criterio de recuperación del trigger es diferente del criterio de problema.
- *None* - nunca se genera un evento OK. Esto puede usarse junto con la generación de múltiples eventos de problema para simplemente enviar una notificación cuando ocurre algo.

Además, los triggers tienen una configuración de ”OK event closes” que controla qué eventos de problema se cierran:

- *All problems* - un evento OK cerrará todos los problemas abiertos creados por el trigger
- *All problems if tag values match* - un evento OK cerrará los problemas abiertos creados por el trigger que tengan al menos un valor de etiqueta coincidente. La etiqueta se define mediante la configuración del trigger ”Tag for matching”. Si no hay eventos de problema que cerrar, entonces no se genera el evento OK. A esto a menudo se le llama correlación de eventos a nivel de trigger.

Correlación de eventos

La correlación de eventos (también denominada correlación global de eventos) es una forma de configurar reglas personalizadas para cerrar eventos (lo que da lugar a la generación de un evento OK).

Las reglas definen cómo se emparejan los nuevos eventos de problema con los eventos de problema existentes y permiten cerrar el nuevo evento o los eventos coincidentes mediante la generación de los eventos OK correspondientes.

Sin embargo, la correlación de eventos debe configurarse con mucho cuidado, ya que puede afectar negativamente al rendimiento del procesamiento de eventos o, si se configura incorrectamente, cerrar más eventos de los previstos (en el peor de los casos, podrían cerrarse todos los eventos de problema). Algunos consejos de configuración:

1. reduzca siempre el alcance de la correlación estableciendo una etiqueta única para el evento de control (el evento que se empareja con eventos antiguos) y utilice la condición de correlación «etiqueta del nuevo evento»
2. no olvide añadir una condición basada en el evento antiguo al utilizar la operación «cerrar evento antiguo», ya que podrían cerrarse todos los problemas existentes
3. evite utilizar nombres de etiquetas comunes empleados por distintas configuraciones de correlación

Administrador de tareas

Si la opción ’Allow manual close’ está habilitada para trigger, entonces es posible cerrar manualmente los eventos de problema generados por el trigger. Esto se hace en el frontend cuando se actualiza un problema. El evento no se cierra directamente; en su lugar, se crea una tarea de ’close event’, que el administrador de tareas procesa poco después. El administrador de tareas generará un evento OK correspondiente y el evento de problema se cerrará.

Eventos de servicio Los eventos de servicio se generan solo si las acciones de servicio para estos eventos están habilitadas. En este caso, cada cambio del estado del servicio crea un nuevo evento:

- Evento de problema - cuando el estado del servicio cambia de `OK` a `Problem`
- Evento OK - cuando el estado del servicio cambia de `Problem` a `OK`

El evento contiene detalles del cambio de estado del servicio: cuándo ocurrió y cuál es el nuevo estado.

Eventos de descubrimiento Zabbix analiza periódicamente los rangos de IP definidos en las reglas de descubrimiento de red. La frecuencia de la comprobación se puede configurar de forma individual para cada regla. Una vez que se descubre un host o un servicio, se genera un evento de descubrimiento (o varios eventos).

Zabbix genera los siguientes eventos:

Evento	Cuándo se genera
Service Up	Cada vez que Zabbix detecta un servicio activo.
Service Down	Cada vez que Zabbix no puede detectar el servicio.
Host Up	Si al menos uno de los servicios está UP para la IP.
Host Down	Si todos los servicios no responden.
Service Discovered	Si el servicio vuelve después de un tiempo de inactividad o se descubre por primera vez.
Service Lost	Si el servicio se pierde después de haber estado activo.
Host Discovered	Si el host vuelve después de un tiempo de inactividad o se descubre por primera vez.
Host Lost	Si el host se pierde después de haber estado activo.

Eventos de autorregistro de agent activo El autorregistro de agent activo crea eventos en Zabbix.

Si está configurado, se crea un evento de autorregistro de agent activo cuando un agent activo previamente desconocido solicita comprobaciones o si los metadatos del host han cambiado. El server agrega un nuevo host autorregistrado, usando la dirección IP y el puerto recibidos del agent.

Para obtener más información, consulte la página de autorregistro de agent activo.

Eventos internos Los eventos internos ocurren cuando:

- un item cambia de estado de 'normal' a 'unsupported'
- un item cambia de estado de 'unsupported' a 'normal'
- una regla de descubrimiento de bajo nivel cambia de estado de 'normal' a 'unsupported'
- una regla de descubrimiento de bajo nivel cambia de estado de 'unsupported' a 'normal'
- un trigger cambia de estado de 'normal' a 'unknown'
- un trigger cambia de estado de 'unknown' a 'normal'

El objetivo de introducir eventos internos es permitir que los usuarios sean notificados cuando se produzca cualquier evento interno, por ejemplo, cuando un item pasa a estar unsupported y deja de recopilar datos.

Los eventos internos solo se crean cuando las acciones internas para estos eventos están habilitadas. Para detener la generación de eventos internos (por ejemplo, para items que pasan a estar unsupported), deshabilite todas las acciones para eventos internos en *Alerts* > *Actions* > *Internal actions*.

> **Note:**
> Si las acciones internas están deshabilitadas, mientras un objeto esté en el estado 'unsupported', el evento de recuperación para este objeto seguirá creándose.
> Si las acciones internas están habilitadas, mientras un objeto esté en el estado 'unsupported', el evento de recuperación para este objeto se creará, aunque no se haya creado un 'problem event' para el objeto.

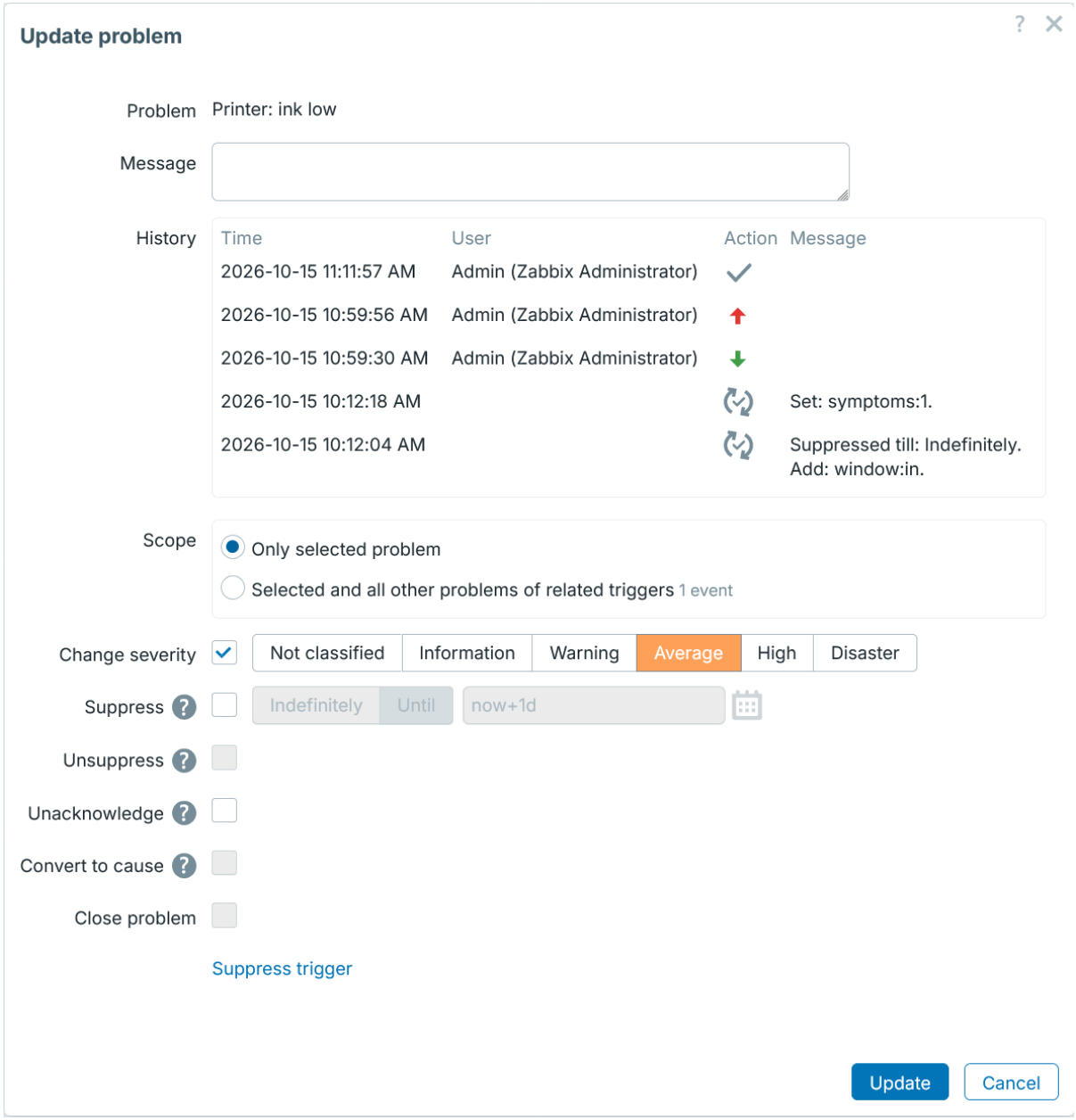
Véase también: Receiving notification on unsupported items

Reconocimiento y supresión de problemas

Los eventos de problema en Zabbix pueden ser reconocidos o suprimidos por los usuarios.

Actualización de problemas La ventana emergente de actualización del problema permite:

- Comentar el problema.
- Ver los comentarios y las acciones realizados hasta el momento.
- Cambiar la gravedad del problema.
- Suprimir o dejar de suprimir el problema.
- Reconocer o dejar de reconocer el problema.
- Cambiar el problema síntoma a problema causa.
- Cerrar manualmente el problema.
- Crear un mantenimiento ad hoc para el trigger.



Todos los campos de entrada obligatorios están marcados con un asterisco rojo.

Parámetro	Descripción
Problema	Si solo se selecciona un problema, se muestra el nombre del problema. Si se seleccionan varios problemas, se muestra *N problemas seleccionados*.
Mensaje	Introduzca el texto para comentar el problema (máximo 2048 caracteres).
Historial	Se muestran las actividades y los comentarios anteriores sobre el problema, junto con la hora y los datos del usuario. Para conocer el significado de los iconos utilizados para indicar las acciones de los usuarios, consulte la página de detalles del evento. Tenga en cuenta que el historial se muestra únicamente si se selecciona un problema para la actualización.
Ámbito	Defina el ámbito de acciones como cambiar la gravedad, reconocer o cerrar manualmente problemas: **Solo el problema seleccionado**: afectará únicamente a este evento. **El problema seleccionado y todos los demás problemas de los triggers relacionados**: en el caso de reconocer o cerrar problemas, afectará a este evento y a todos los demás problemas que aún no se hayan reconocido o cerrado. Si el ámbito contiene problemas ya reconocidos o cerrados, estos problemas no se reconocerán ni cerrarán repetidamente. Por otro lado, el número de operaciones de cambio de mensaje y gravedad no está limitado.

Parámetro	Descripción
Cambiar gravedad	Marque la casilla de verificación y haga clic en el botón de gravedad para actualizar la gravedad del problema. La casilla de verificación para cambiar la gravedad está disponible si existen permisos de lectura y escritura para al menos uno de los problemas seleccionados. Al hacer clic en *Actualizar*, solo se actualizarán los problemas con permisos de lectura y escritura. Si no existen permisos de lectura y escritura para ninguno de los triggers seleccionados, la casilla de verificación estará deshabilitada.
Suprimir	Marque la casilla de verificación para suprimir el problema: **Indefinidamente**: suprimir indefinidamente. **Hasta**: suprimir hasta una hora determinada. Se admiten formatos de hora absolutos y relativos, por ejemplo: `now+1d`: durante un día a partir de ahora (predeterminado). `now/w`: hasta el final de la semana actual. `2022-05-28 12:00:00`: hasta la fecha y hora absolutas. Tenga en cuenta que no se admite un período simple (por ejemplo, `1d`, `1w`). La disponibilidad de esta opción depende de la configuración del rol de usuario "Suprimir problemas". Consulte también: Supresión de problemas
Dejar de suprimir	Marque la casilla de verificación para dejar de suprimir el problema. Esta casilla de verificación está activa únicamente si al menos uno de los problemas seleccionados está suprimido. La disponibilidad de esta opción depende de la configuración del rol de usuario "Suprimir problemas".
Reconocer	Marque la casilla de verificación para reconocer el problema. Esta casilla de verificación está disponible si hay al menos un problema no reconocido entre los seleccionados. No es posible añadir otro reconocimiento a un problema ya reconocido, aunque sí es posible añadir otro comentario.
Dejar de reconocer	Marque la casilla de verificación para dejar de reconocer el problema. Esta casilla de verificación está disponible si hay al menos un problema reconocido entre los seleccionados.
Convertir en causa	Marque la casilla de verificación para convertir los problemas síntoma en problemas causa.
Cerrar problema	Marque la casilla de verificación para cerrar manualmente los problemas seleccionados. La casilla de verificación para cerrar un problema está disponible si la opción *Permitir cierre manual* está marcada en la configuración del trigger para al menos uno de los problemas seleccionados. Al hacer clic en *Actualizar*, solo se cerrarán los problemas que puedan cerrarse. Si ningún problema puede cerrarse manualmente, la casilla de verificación estará deshabilitada. Los problemas ya cerrados no se cerrarán repetidamente.
Suprimir trigger	Haga clic para iniciar un mantenimiento ad hoc para el trigger. Se abrirá el formulario de configuración de un período de mantenimiento. En él, se preseleccionará un período de mantenimiento único de 1 h, que comienza ahora y cuya duración se puede configurar en los ajustes y el perfil del usuario. El trigger correspondiente también se preseleccionará. El nombre del mantenimiento se rellenará previamente con el formato `Ad-hoc: <event name>`.

Confirmación de problema Si un usuario recibe una notificación sobre un evento de problema, puede ir al frontend de Zabbix, abrir la ventana emergente de actualización del problema de ese problema usando una de las formas enumeradas a continuación y confirmar el problema. Al confirmarlo, puede introducir su comentario sobre él, indicando que está trabajando en ello o cualquier otra cosa que desee decir al respecto.

De este modo, si otro usuario del sistema detecta el mismo problema, verá inmediatamente si ya ha sido confirmado y los comentarios realizados hasta el momento.

De esta forma, el flujo de trabajo para resolver problemas con más de un usuario del sistema puede llevarse a cabo de manera coordinada.

El estado de confirmación también se utiliza al definir operaciones de action. Puede definir, por ejemplo, que se envíe una notificación a un responsable de nivel superior solo si un evento no ha sido confirmado durante cierto tiempo.

Para confirmar eventos y comentar sobre ellos, un usuario debe tener al menos permisos de lectura sobre los triggers correspondientes. Para cambiar la severidad del problema o cerrar el problema, un usuario debe tener permisos de lectura y escritura sobre los triggers correspondientes.

Hay **varias** formas de acceder a la ventana emergente de actualización del problema, que permite confirmar un problema.

- Puede seleccionar problemas en *Monitoring* > *Problems* y luego hacer clic en *Mass update* debajo de la lista
- Puede hacer clic en *Update* en la columna *Update* de un problema en:
  - *Dashboards* (widgets *Problems* y *Problems by severity*)
  - *Monitoring* > *Problems*
  - *Monitoring* > *Problems* > *Event details*
- Puede hacer clic en una celda de problema no resuelto en:
  - *Dashboards* (widget *Trigger overview*)

El menú emergente contiene una opción *Update* que le llevará a la ventana de actualización del problema.

Según la información de confirmación, es posible configurar cómo se muestra el recuento de problemas en el dashboard o en los mapas. Para ello, debe realizar selecciones en la opción *Problem display*, disponible tanto en la configuración del mapa como en el widget del dashboard *Problems by severity*.

Es posible mostrar el recuento total de problemas, el recuento de problemas no confirmados separado del total o únicamente el recuento de problemas no confirmados.

Según la información de actualización de los problemas (confirmación, etc.), es posible configurar operaciones de actualización: enviar un mensaje o ejecutar comandos remotos.

Supresión de problemas La supresión de problemas ofrece una forma de ocultar temporalmente un problema que puede abordarse más adelante. Esto resulta útil para limpiar la lista de problemas y dar la máxima prioridad a los problemas más urgentes. Por ejemplo, a veces puede surgir un problema durante el fin de semana que no sea lo suficientemente urgente como para abordarlo de inmediato, por lo que se puede ”posponer” hasta el lunes por la mañana.

La supresión de problemas permite ocultar un *único* problema, a diferencia de la supresión mediante el mantenimiento del host, cuando se ocultan todos los problemas del host en mantenimiento.

Las operaciones de las acciones de trigger se pausarán para los problemas suprimidos de la misma forma que con el mantenimiento del host.

Un problema puede suprimirse a través de la ventana **actualización del problema**, donde la supresión es una de las opciones de actualización del problema junto con comentar, cambiar la severidad, reconocerlo, etc.

Un problema también puede dejar de estar suprimido a través de la misma ventana de actualización del problema.

Visualización

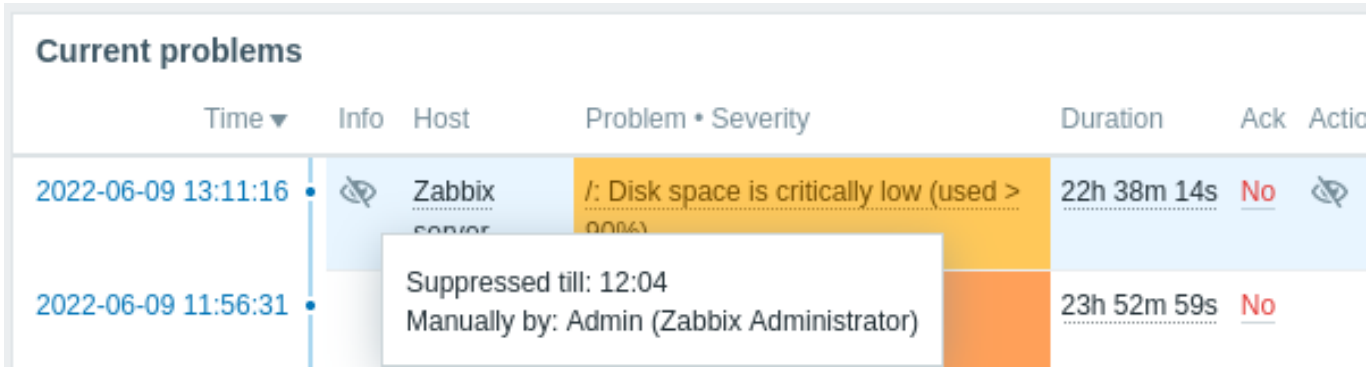
Una vez suprimido, el problema se marca con un icono de supresión parpadeante  en la columna *Info*, antes de ocultarse.

Si la supresión se debe al mantenimiento del host, el icono de supresión en la columna *Info* se muestra en naranja. Si se aplican tanto el mantenimiento como la supresión manual, la supresión manual tiene prioridad en el color del icono.

El icono de supresión parpadea mientras la tarea de supresión está en la lista de espera. Una vez que el gestor de tareas ha suprimido el problema, el icono dejará de parpadear. Si el icono de supresión sigue parpadeando durante mucho tiempo, esto puede indicar un problema del server; por ejemplo, si el server está caído y el gestor de tareas no puede completar la tarea. La misma lógica se aplica a la anulación de la supresión. En el breve período posterior a que se envía la tarea y el server no la ha completado, el icono de anulación de la supresión parpadea.

Un problema suprimido puede estar oculto o visible, según la configuración del filtro/widget de problemas.

Cuando se muestra en la lista de problemas, un problema suprimido se marca con el icono de supresión y los detalles de la supresión se muestran al pasar el mouse:



Los detalles de la supresión también se muestran en una ventana emergente al colocar el mouse sobre el icono de supresión en la columna *Actions*.

Cierre manual de problemas

Resumen Aunque, por lo general, los eventos de problema se resuelven automáticamente cuando el estado del trigger pasa de `Problem` a `OK`, puede haber casos en los que sea difícil determinar si un problema se ha resuelto mediante una expresión de trigger. En tales casos, el problema debe resolverse manualmente.

Por ejemplo, *syslog* puede informar de que algunos parámetros del kernel necesitan ajustarse para un rendimiento óptimo. En este caso, el problema se notifica a los administradores de Linux, ellos lo corrigen y luego cierran el problema manualmente.

Los problemas solo pueden cerrarse manualmente para los triggers con la opción *Allow manual close* habilitada.

Cuando un problema se "cierra manualmente", Zabbix genera una nueva tarea interna para el server de Zabbix. Luego, el proceso *task manager* ejecuta esta tarea y genera un evento OK, cerrando así el evento de problema.

El evento *OK* generado incluye el conjunto completo de etiquetas de evento resueltas para ese evento (incluidas las etiquetas heredadas de templates, hosts y triggers). Estas etiquetas están disponibles en las notificaciones y en macros como `{EVENT.RECOVERY.TAGS}` y `{EVENT.RECOVERY.TAGSJSON}`.

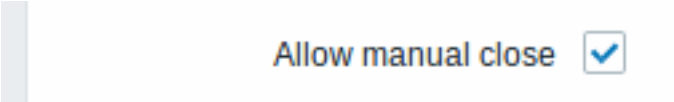
Que un problema se cierre manualmente no significa que el trigger subyacente nunca volverá a entrar en estado *Problem*. La expresión del trigger se vuelve a evaluar y puede dar lugar a un problema:

- Cuando llegan nuevos datos para cualquier item incluido en la expresión del trigger (tenga en cuenta que los valores descartados por un paso de preprocesamiento de limitación no se consideran recibidos y no provocarán que la expresión del trigger se vuelva a evaluar);
- Cuando se utilizan las funciones date and time y/o nodata() en la expresión.

Configuración Se requieren dos pasos para cerrar un problema manualmente.

Configuración de trigger

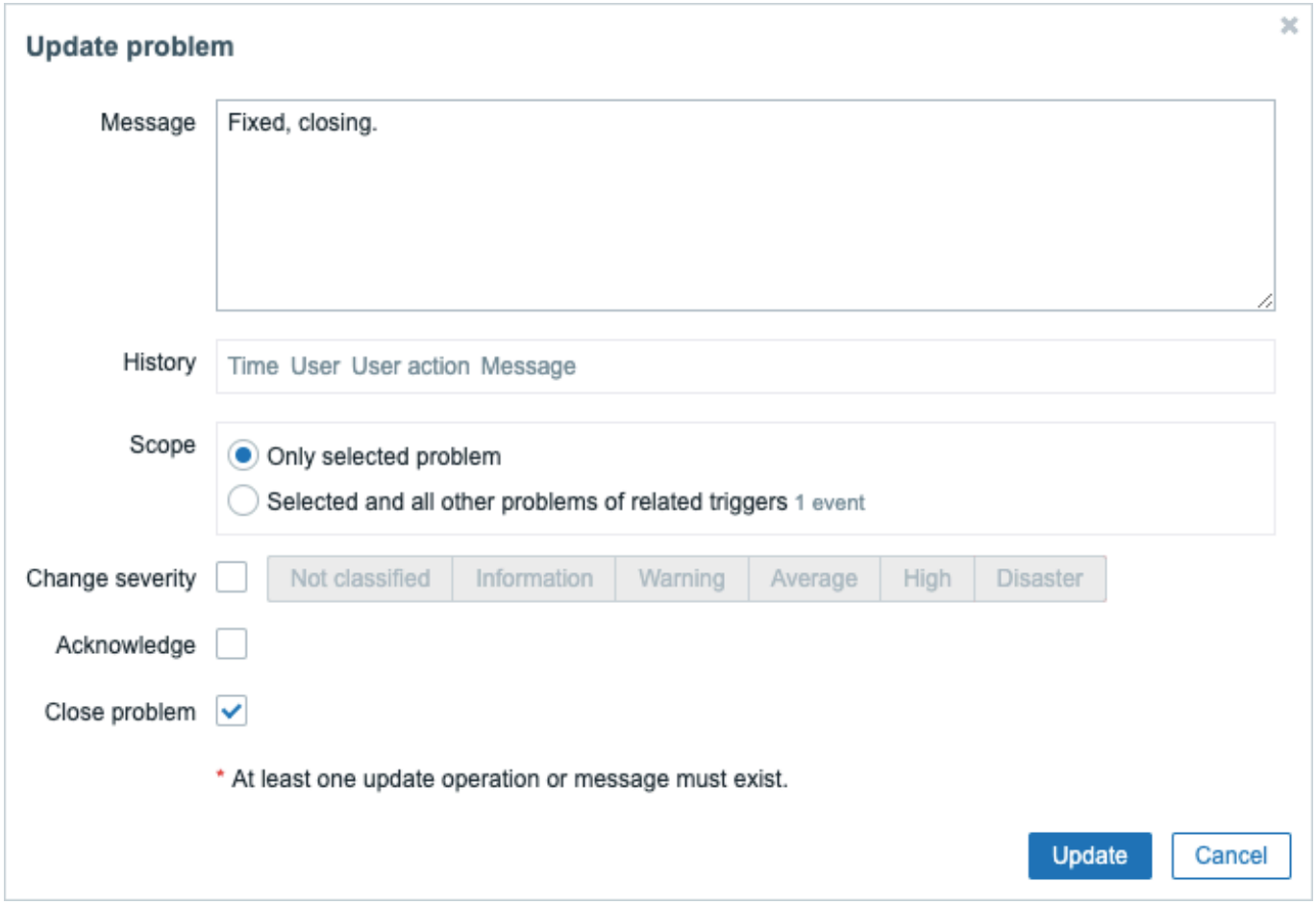
En la configuración de trigger, habilite la opción *Allow manual close*.



Ventana de actualización del problema

Si surge un problema para un trigger con la marca *Manual close*, puede abrir la ventana emergente de actualización del problema de ese problema y cerrarlo manualmente.

Para cerrar el problema, marque la opción *Close problem* en el formulario y haga clic en *Update*.



Todos los campos de entrada obligatorios están marcados con un asterisco rojo.

La solicitud es procesada por Zabbix server. Normalmente tardará unos segundos en cerrar el problema. Durante ese proceso, *CLOSING* se muestra en *Monitoring* > *Problems* como estado del problema.

Verificación Se puede verificar que un problema se ha cerrado manualmente:

- en los detalles del evento, disponibles a través de *Monitoring* > *Problems*;
- usando la macro `{EVENT.UPDATE.HISTORY}` en los mensajes de notificación, que proporcionará esta información.

Períodos de mantenimiento

Descripción general

El mantenimiento se utiliza para suprimir problemas durante períodos de tiempo predefinidos.

Puede definir períodos de mantenimiento para estas entidades:

- Grupos de hosts
- Hosts
- Triggers
- Nombres de eventos
- Etiquetas

Si el mantenimiento se define para etiquetas específicas o nombres de eventos, se activará únicamente para los triggers correspondientes; los demás triggers del host o del grupo de hosts no estarán en mantenimiento.

Hay dos tipos de mantenimiento: *Con recopilación de datos* y *Sin recopilación de datos*.

Durante el mantenimiento *Con recopilación de datos*, los triggers se procesan con normalidad y los eventos se crean cuando es necesario. Sin embargo, si la opción *Pausar las operaciones para los problemas suprimidos* está seleccionada en la configuración de la acción, las escaladas de problemas se pausan para las entidades en mantenimiento. En este caso, los pasos de escalada, que pueden incluir el envío de notificaciones o comandos remotos, se ignorarán mientras dure el período de mantenimiento. Tenga en cuenta que las operaciones de recuperación y actualización no se suprimen durante el mantenimiento, solo las escaladas. Las notificaciones de recuperación no se envían si el problema comenzó durante el mantenimiento.

Por ejemplo, si los pasos de escalada están programados a los 0, 30 y 60 minutos después del inicio de un problema, y existe un mantenimiento de media hora que comienza 10 minutos después de que aparece un problema real y termina 40 minutos después, los pasos dos y tres se ejecutarán media hora más tarde, es decir, a los 60 y 90 minutos (siempre que el problema siga existiendo). Del mismo modo, si aparece un problema durante el mantenimiento, la escalada comenzará después del mantenimiento.

Para recibir normalmente las notificaciones de problemas durante el mantenimiento (sin retrasos), debe desmarcar la opción *Pausar las operaciones para los problemas suprimidos* en la configuración de la acción.

> **Note:**
> Si al menos un host (utilizado en la expresión del trigger) no está en modo de mantenimiento, Zabbix enviará una notificación de problema.

El Zabbix server debe estar en ejecución durante el mantenimiento. Los mantenimientos se recalculan cada minuto o tan pronto como se vuelva a cargar la caché de configuración si se producen cambios en el período de mantenimiento.

Los procesos de temporizador comprueban si el estado del host debe cambiar a mantenimiento o salir de este a los 0 segundos de cada minuto. Además, cada segundo el proceso de temporizador comprueba si se debe iniciar o detener algún mantenimiento en función de si se han producido cambios en los períodos de mantenimiento después de actualizar la configuración. Por lo tanto, la velocidad de inicio o detención de los períodos de mantenimiento depende del intervalo de actualización de la configuración (10 segundos de forma predeterminada). Tenga en cuenta que los cambios en los períodos de mantenimiento no incluyen las opciones *Activo desde/Activo hasta*. Además, si se añade una entidad (grupo de hosts/host/trigger/etc.) a un período de mantenimiento activo existente, los cambios solo se activarán mediante el proceso de temporizador al inicio del minuto siguiente.

Tenga en cuenta que cuando un host entra en mantenimiento, los procesos de temporizador del Zabbix server leerán todos los problemas abiertos para comprobar si es necesario suprimirlos. Esto puede afectar al rendimiento si hay muchos problemas abiertos. El Zabbix server también leerá todos los problemas abiertos al iniciarse, incluso si en ese momento no hay mantenimientos configurados.

Tenga en cuenta que el Zabbix server (o proxy) siempre recopila datos independientemente del tipo de mantenimiento (incluido el mantenimiento *Sin recopilación de datos*). Posteriormente, el server ignora los datos si se ha establecido *Sin recopilación de datos*.

Cuando finaliza el mantenimiento *Sin recopilación de datos*, los triggers que utilizan la función `nodata()` no se activarán antes de la siguiente comprobación del período que están verificando.

Si se añade un item de registro mientras un host está en mantenimiento y este finaliza, solo se recopilarán las nuevas entradas del archivo de registro posteriores al fin del mantenimiento.

Si se envía un valor con marca de tiempo para un host que está en mantenimiento *Sin recopilación de datos* (por ejemplo, mediante Zabbix sender), este valor se descartará; sin embargo, es posible enviar un valor con marca de tiempo correspondiente a un período de mantenimiento expirado, y se aceptará.

Si el usuario cambia el período de mantenimiento o la opción de mantenimiento (grupo de hosts, host, trigger, etc.), los cambios solo surtirán efecto después de la sincronización de la caché de configuración.

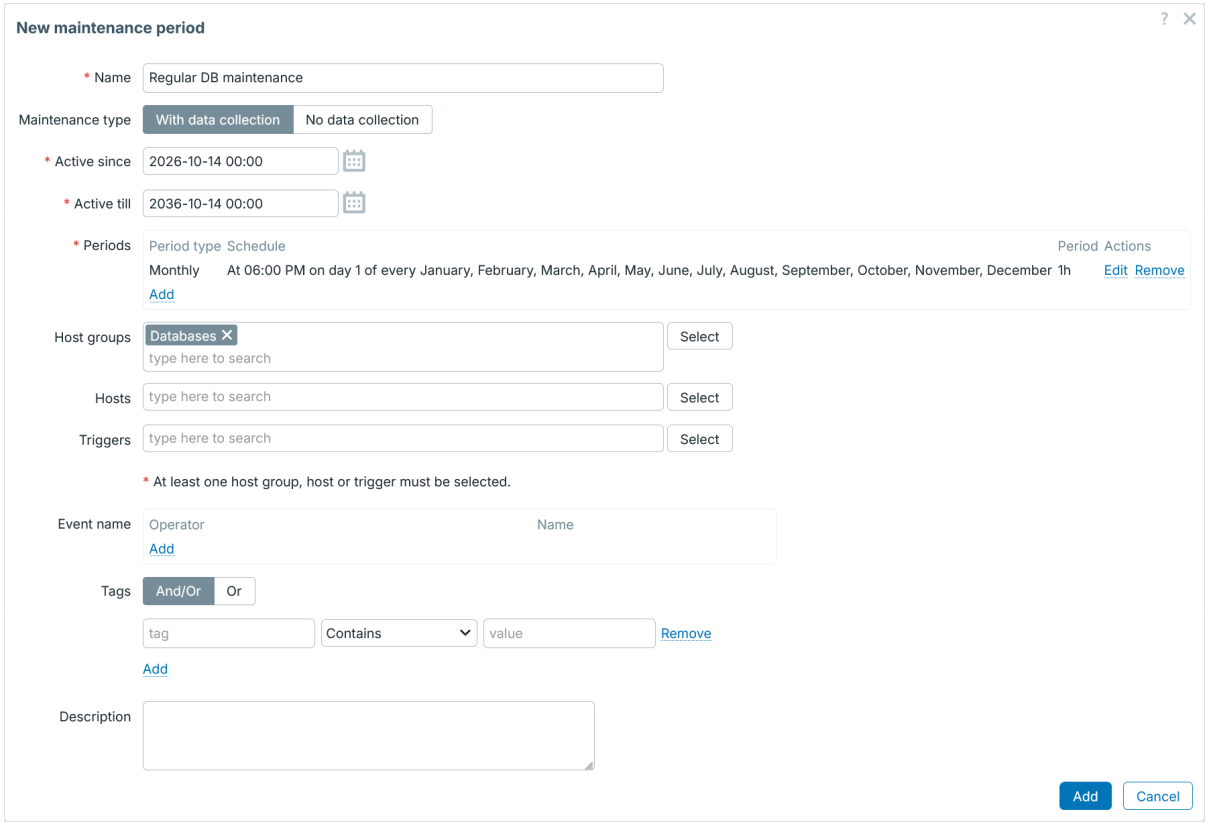
Configuración

Para iniciar un período de mantenimiento inmediatamente:

1. Vaya a *Monitoring* > *Problems*.
2. Haga clic en el nombre del problema.
3. Seleccione una opción de mantenimiento en el menú del evento (host, trigger, nombre del evento, etiquetas).
4. Modifique los parámetros de mantenimiento en el formulario.

Para configurar un período de mantenimiento planificado:

1. Vaya a *Data collection* > *Maintenance*.
2. Haga clic en *Create maintenance period* (o en el nombre de un período de mantenimiento existente).
3. Introduzca los parámetros de mantenimiento en el formulario.



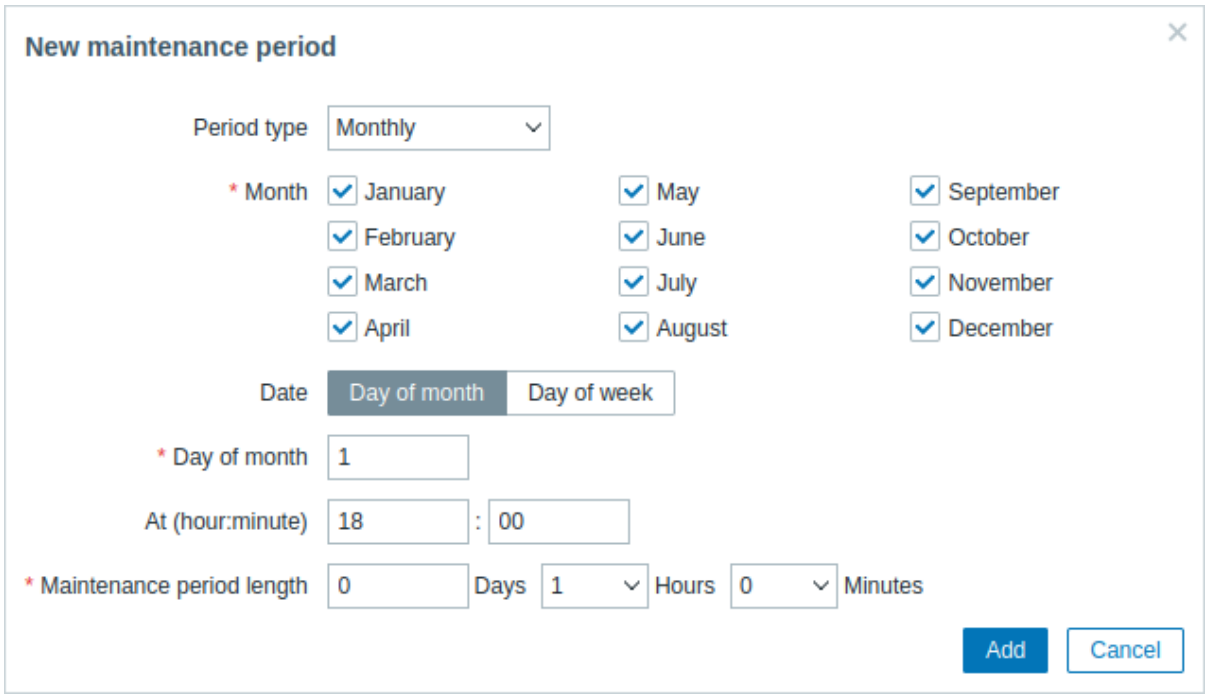
Todos los campos de entrada obligatorios están marcados con un asterisco rojo.

Parámetro	Descripción
Name	Nombre del período de mantenimiento.
Maintenance type	Se pueden establecer dos tipos de mantenimiento: **With data collection**: el server recopila datos durante el mantenimiento y se procesan los triggers; **No data collection**: es posible que los datos sigan recopilándose, pero no se guardan en la base de datos durante el mantenimiento y los triggers (incluida la función `nodata()`) no se activan. Consulte Efecto de los períodos de mantenimiento para saber cómo afecta cada tipo al informe de disponibilidad.
Active since	Fecha y hora a partir de las cuales la ejecución de los períodos de mantenimiento pasa a estar activa. *Nota:* Establecer solo esta hora no activa un período de mantenimiento; los períodos de mantenimiento deben configurarse en *Periods* (consulte más abajo).
Active till	Fecha y hora hasta las cuales la ejecución de los períodos de mantenimiento permanece activa.
Periods	Este bloque permite definir los días y las horas exactos en los que se realiza el mantenimiento. Al hacer clic en Add se abre una ventana emergente con un formulario flexible de *Maintenance period*, donde puede definir la programación del mantenimiento. Consulte Períodos de mantenimiento para obtener una descripción detallada.
Host groups	Seleccione los grupos de hosts para los que se activará el mantenimiento. El mantenimiento se activará para todos los hosts del grupo o grupos de hosts especificados. Este campo admite autocompletado; al comenzar a escribir, se mostrará una lista desplegable con todos los grupos de hosts disponibles. Especificar un grupo de hosts principal selecciona implícitamente todos los grupos de hosts anidados. Por lo tanto, el mantenimiento también se activará en los hosts de los grupos anidados.
Hosts	Seleccione los hosts para los que se activará el mantenimiento. Este campo admite autocompletado; al comenzar a escribir, se mostrará una lista desplegable con todos los hosts disponibles.
Triggers	Seleccione los triggers para los que se activará el mantenimiento. Este campo admite autocompletado; al comenzar a escribir, se mostrará una lista desplegable con todos los triggers disponibles.

Parámetro	Descripción
Event name	Especifique los nombres de eventos para los que se activará el mantenimiento. Se pueden establecer varias condiciones. Hay dos operadores disponibles para cada condición: **Contains**: incluye los nombres de etiquetas especificados cuando los valores de las etiquetas contienen la cadena introducida (coincidencia de subcadena, sin distinguir mayúsculas de minúsculas) **Does not contain**: excluye los nombres de etiquetas especificados cuando los valores de las etiquetas contienen la cadena introducida (coincidencia de subcadena, sin distinguir mayúsculas de minúsculas)
Tags	Especifique las etiquetas para las que se activará el mantenimiento. Se pueden establecer varias condiciones. La coincidencia de nombres de etiquetas siempre distingue entre mayúsculas y minúsculas. Hay varios operadores disponibles para cada condición: **Contains**: incluye los nombres de etiquetas especificados cuando los valores de las etiquetas contienen la cadena introducida (coincidencia de subcadena, distinguiendo mayúsculas de minúsculas); **Equals**: incluye los nombres y valores de etiquetas especificados (distinguiendo mayúsculas de minúsculas). **Does not equal**: excluye los nombres y valores de etiquetas especificados (distinguiendo mayúsculas de minúsculas) **Does not contain**: excluye los nombres de etiquetas especificados cuando los valores de las etiquetas contienen la cadena introducida (coincidencia de subcadena, sin distinguir mayúsculas de minúsculas) Hay dos tipos de cálculo para las condiciones: **And/Or**: deben cumplirse todas las condiciones; las condiciones que tienen el mismo nombre de etiqueta se agrupan mediante la condición Or; **Or**: basta con que se cumpla una condición. Las etiquetas solo se pueden especificar si se selecciona el tipo de mantenimiento *With data collection*.
Description	Descripción del período de mantenimiento.

Períodos de mantenimiento

La ventana del período de mantenimiento se utiliza para programar el tiempo de un mantenimiento recurrente o de una sola vez. El formulario es dinámico y los campos disponibles cambian según el *Tipo de período* seleccionado.



Tipo de período	Descripción
Solo una vez	Configure un período de mantenimiento de una sola vez: *Fecha* - fecha y hora del período de mantenimiento; *Duración del período de mantenimiento* - durante cuánto tiempo estará activo el mantenimiento.
Diario	Configure un período de mantenimiento diario: *Cada día(s)* - frecuencia del mantenimiento (1 - *(predeterminado)* cada día, 2 - cada dos días, etc.); *A las (hora:minuto)* - hora del día en que comienza el mantenimiento; *Duración del período de mantenimiento* - durante cuánto tiempo estará activo el mantenimiento. Cuando el parámetro *Cada día(s)* es mayor que ”1”, el día de inicio es el día en que cae la hora de *Activo desde*. Ejemplos: - si *Activo desde* se establece en ”2021-01-01 12:00”, *Cada día(s)* se establece en ”2” y *A las (hora:minuto)* se establece en ”23:00”, entonces el primer período de mantenimiento comenzará el 1 de enero a las 23:00, mientras que el segundo período de mantenimiento comenzará el 3 de enero a las 23:00; - si *Activo desde* se establece en ”2021-01-01 12:00”, *Cada día(s)* se establece en ”2” y *A las (hora:minuto)* se establece en ”01:00”, entonces el primer período de mantenimiento comenzará el 3 de enero a las 01:00, mientras que el segundo período de mantenimiento comenzará el 5 de enero a las 01:00.
Semanal	Configure un período de mantenimiento semanal: *Cada semana(s)* - frecuencia del mantenimiento (1 - *(predeterminado)* cada semana, 2 - cada dos semanas, etc.); *Día de la semana* - en qué día debe tener lugar el mantenimiento; *A las (hora:minuto)* - hora del día en que comienza el mantenimiento; *Duración del período de mantenimiento* - durante cuánto tiempo estará activo el mantenimiento. Cuando el parámetro *Cada semana(s)* es mayor que ”1”, la semana de inicio es la semana en que cae la hora de *Activo desde*. Para ver ejemplos, consulte la descripción del parámetro *Diario* anterior.
Mensual	Configure un período de mantenimiento mensual: *Mes* - seleccione todos los meses durante los cuales se realiza el mantenimiento regular; *Fecha:* ***Día del mes*** - seleccione esta opción si el mantenimiento debe tener lugar en la misma fecha cada mes (por ejemplo, cada día 1 del mes) y luego seleccione el día requerido en el campo *Día del mes* que aparece; *Fecha:* ***Día de la semana*** - seleccione esta opción si el mantenimiento debe tener lugar solo en ciertos días (por ejemplo, cada primer lunes del mes), luego seleccione (en la lista desplegable) la semana requerida del mes (primera, segunda, tercera, cuarta o última) y después marque las casillas de los día(s) de mantenimiento; *A las (hora:minuto)* - hora del día en que comienza el mantenimiento; *Duración del período de mantenimiento* - durante cuánto tiempo estará activo el mantenimiento.

Attention:
Al crear un período de mantenimiento, se utiliza la zona horaria del usuario que lo crea. Sin embargo, cuando se programan períodos de mantenimiento recurrentes (*Daily*, *Weekly*, *Monthly*), se utiliza la zona horaria del server de Zabbix. Para garantizar un comportamiento predecible de los períodos de mantenimiento recurrentes, es necesario usar una zona horaria común para todas las partes de Zabbix.

Cuando haya terminado, haga clic en *Add* para añadir el período de mantenimiento a la sección *Periods*.

Tenga en cuenta que los cambios de horario de verano (DST) no afectan a la duración del mantenimiento. Por ejemplo, supongamos que tenemos configurado un mantenimiento de dos horas que normalmente comienza a las 01:00 y finaliza a las 03:00:

- si después de una hora de mantenimiento (a las 02:00) se produce un cambio de DST y la hora actual cambia de 02:00 a 03:00, el mantenimiento continuará una hora más (hasta las 04:00);
- si después de dos horas de mantenimiento (a las 03:00) se produce un cambio de DST y la hora actual cambia de 03:00 a 02:00, el mantenimiento se detendrá, porque ya han transcurrido dos horas;
- si un período de mantenimiento comienza durante la hora que se omite por un cambio de DST, entonces el mantenimiento no se iniciará.

Si un período de mantenimiento se establece en ”1 day” (el período real del mantenimiento es de 24 horas, ya que Zabbix calcula los días en horas), comienza a las 00:00 y finaliza a las 00:00 del día siguiente:

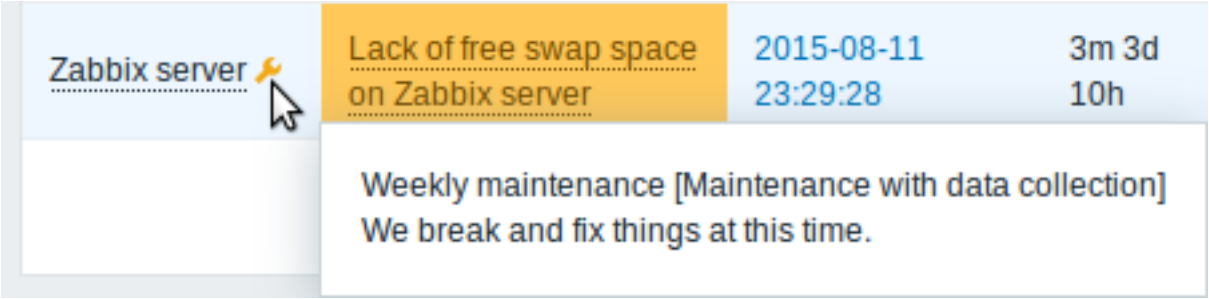
- el mantenimiento se detendrá a las 01:00 del día siguiente si la hora actual avanza una hora;
- el mantenimiento se detendrá a las 23:00 de ese día si la hora actual retrocede una hora.

Mostrar

Mostrando hosts en mantenimiento

Un icono de llave inglesa naranja junto al nombre del host indica que este host está en mantenimiento en:

- *Dashboards*
- *Monitoring* > *Problems*
- *Inventory* > *Hosts* > *Host inventory details*
- *Data collection* > *Hosts* (consulte la columna *Status*)



Los detalles del mantenimiento se muestran cuando el puntero del mouse se coloca sobre el icono.

Además, los hosts en mantenimiento tienen un fondo naranja en *Monitoring* > *Maps*.

Visualización de problemas suprimidos

Normalmente, los problemas de los hosts en mantenimiento se suprimen, es decir, no se muestran en el frontend.

Los problemas suprimidos también pueden estar ocultos porque se suprimieron manualmente.

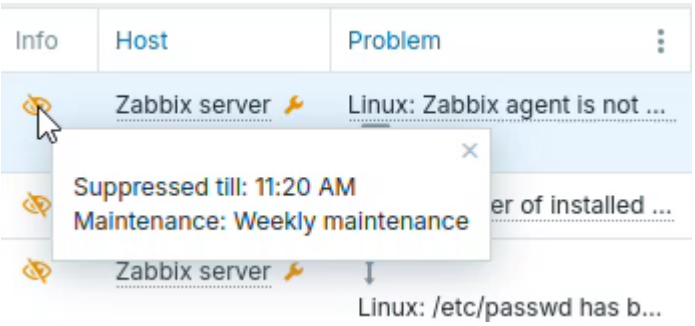
Es posible mostrar los problemas suprimidos seleccionando la opción *Mostrar problemas suprimidos* en estas ubicaciones:

- *Dashboards* (en la configuración del widget *Problem hosts*, *Problems*, *Problems by severity*, *Trigger overview*)
- *Monitoring* > *Problems* (en el filtro)
- *Monitoring* > *Maps* (en la configuración del mapa)
- notificaciones globales (en la configuración del perfil de usuario)

Cuando se muestran problemas suprimidos, se muestra el siguiente icono: . Si la supresión se debe al mantenimiento del host, se muestran los siguientes iconos:

- - el problema se ha suprimido debido al mantenimiento del host.
- - el problema se ha dejado de suprimir después del mantenimiento del host.

Al pasar el cursor sobre el icono se muestran más detalles. En los detalles del evento, puede comprobar si el evento se suprimió debido al mantenimiento o si se suprimió manualmente. Si se aplican tanto el mantenimiento como la supresión manual, la supresión manual tiene prioridad en el color del icono.



Cálculo de las colas durante el mantenimiento

Las colas mostradas en el frontend de Zabbix (Administration > Queue) son calculadas por el server de Zabbix. No incluyen los items en mantenimiento de *No data collection*, y la longitud de la cola siempre es cero para estos items, incluso cuando sus valores están retrasados. Los items retrasados en mantenimiento de *With data collection* siguen contándose en la cola.

El proxy de Zabbix no tiene conocimiento de los periodos de mantenimiento porque no existe sincronización de la configuración de mantenimiento entre el server de Zabbix y el proxy. Las comprobaciones internas calculadas en los proxies de Zabbix (por

ejemplo, `zabbix[queue,,]` y `zabbix[stats,,,queue,,]`) informan de los items retrasados independientemente del estado de mantenimiento en el server de Zabbix.

Como resultado, el frontend de Zabbix y las comprobaciones internas en los proxies de Zabbix pueden informar de diferentes longitudes de cola para los mismos items en mantenimiento de *No data collection*.

Etiquetado

Las etiquetas tienen múltiples propósitos, sobre todo para marcar eventos. Cuando las entidades están etiquetadas, cualquier nuevo evento relacionado con una entidad etiquetada heredará sus etiquetas. Por ejemplo:

- con templates etiquetadas: cualquier problema de un host (creado por triggers del template) heredará las etiquetas del template.
- con hosts etiquetados: cualquier problema de un host heredará las etiquetas del host.
- con items/escenarios web etiquetados: cualquier problema de un item/escenario web heredará las etiquetas del item/escenario web.
- con triggers etiquetados: cualquier problema creado por el trigger heredará las etiquetas del trigger.

Un evento de problema hereda todas las etiquetas de toda la cadena de entidades: templates, hosts, items/escenarios web, triggers. Las combinaciones idénticas `tag:value` (después de resolver macros) se fusionan en una sola, evitando así la duplicación.

Los eventos de recuperación generados por un cierre manual también incluyen las etiquetas de evento resueltas heredadas de templates, hosts, items/escenarios web y triggers. Estas etiquetas están disponibles en las notificaciones y mediante macros como `{EVENT.RECOVERY.TAGS}` y `{EVENT.RECOVERY.TAGSJSON}`.

Las etiquetas de evento personalizadas ofrecen más flexibilidad. Por ejemplo:

- la correlación de eventos se puede configurar en función de las etiquetas de evento.
- las condiciones de acción se pueden configurar en función de las etiquetas de evento.
- los problemas de item se pueden agrupar en función de las etiquetas de evento.
- las etiquetas de problema se pueden usar para asignar problemas a servicios.

Las entidades pueden etiquetarse con el mismo nombre de etiqueta pero con distintos valores de etiqueta (por ejemplo, `component:memory` y `component:storage`). Del mismo modo, una entidad puede tener una etiqueta sin valor y la misma etiqueta con un valor (por ejemplo, `database` y `database:postgresql`). Estas etiquetas no se consideran duplicadas.

Las etiquetas constan de un nombre de etiqueta y un valor de etiqueta. Al etiquetar entidades, puede usar solo el nombre o combinarlo con un valor (por ejemplo, `mysql`, `jira`, `target:mysql`, `service:jira`, etc.).

Las etiquetas se pueden definir para varias entidades:

- Templates
- Hosts
- Items
- Web scenarios
- Triggers
- Services
- Template items and triggers
- Host, item, and trigger prototypes

En las listas de *Templates*, *Hosts*, *Items*, *Triggers*, *Web scenarios* y sus prototipos, la columna *Tags* muestra tanto las etiquetas normales como las heredadas. Al pasar el cursor sobre una etiqueta *heredada* o hacer clic en ella, aparece un tooltip de JavaScript con el texto "Inherited tag". Si una etiqueta heredada también existe como etiqueta normal, se muestra un texto de tooltip diferente según la lista en la que aparezca (por ejemplo, "Inherited and template tag" en la lista de templates o "Inherited and host tag" en la lista de hosts). En otras partes del frontend, las etiquetas normales y heredadas se muestran juntas (se eliminan los duplicados), sin iconos y sin texto adicional de tooltip.

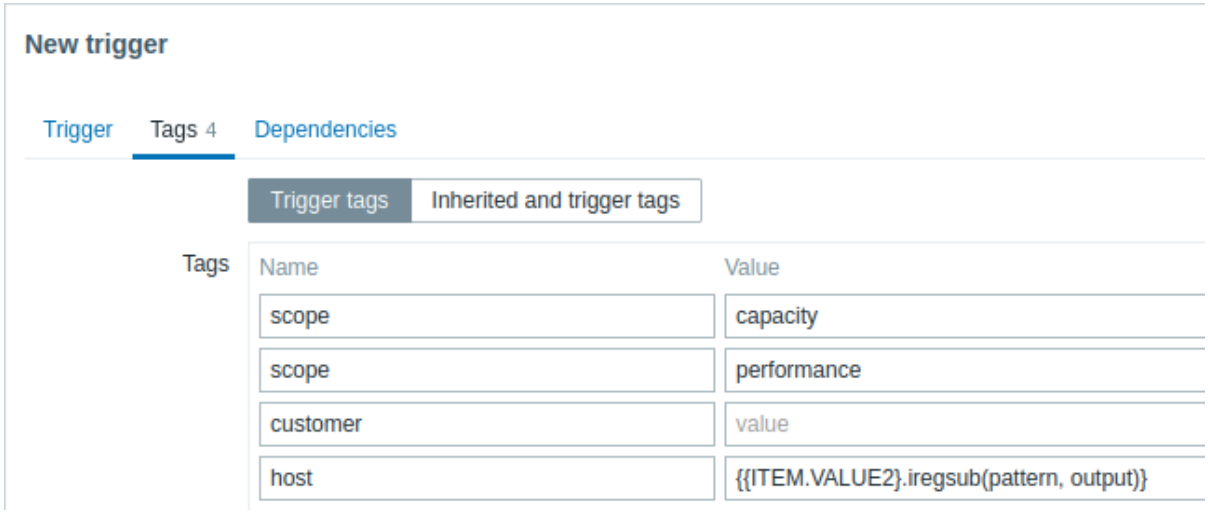
> **Note:**
> Consulte las directrices oficiales de Zabbix para recomendaciones generales sobre la definición de etiquetas, junto con la orientación específica para templates, items, triggers y reglas de descubrimiento de bajo nivel.

Casos de uso Algunos casos de uso comunes para el etiquetado son los siguientes:

1. Marcar eventos de trigger:
   - Defina una etiqueta de trigger (por ejemplo, `scope:performance`).
   - Los problemas creados por este trigger tendrán la etiqueta de trigger.

2. Marcar problemas heredados de template:
    - Defina una etiqueta de template (por ejemplo, `target:mysql`).
    - Los problemas creados por triggers de este template tendrán la etiqueta de template.
3. Marcar problemas de host:
    - Defina una etiqueta de host (por ejemplo, `service:jira`).
    - Los problemas creados por triggers de este host tendrán la etiqueta de host.
4. Filtrar items relacionados:
    - Defina una etiqueta de item (por ejemplo, `component:cpu`).
    - En *Monitoring* > *Latest data*, los items se pueden filtrar por la etiqueta `component:cpu`.
5. Usar información extraída del valor del item como valor de la etiqueta:
    - Defina una etiqueta con una macro como valor de la etiqueta (por ejemplo, `tag-name:{{ITEM.VALUE<N>}.regsub()}`).
    - En *Monitoring* > *Problems*, los problemas tendrán el valor de la etiqueta resuelto a partir de los datos extraídos del valor del item.
6. Identificar problemas en un archivo de registro y cerrarlos por separado:
    - Defina una etiqueta de trigger para el item de monitoreo de logs que extraerá valores del valor del item usando una macro (por ejemplo, `service:{{ITEM.VALUE<N>}.regsub()}`).
    - En la configuración del trigger, configure la correlación de eventos:
        - establezca *PROBLEM event generation mode* en "Multiple";
        - establezca *OK event closes* en "All problems if tag values match";
        - establezca la etiqueta para la coincidencia.
    - Los problemas creados por el trigger del item de log tendrán la etiqueta de trigger y se cerrarán individualmente.
7. Filtrar notificaciones:
    - Defina etiquetas de trigger (por ejemplo, `scope:security` para trigger1 y `scope:availability` para trigger2).
    - Use el filtrado por etiquetas en las condiciones de acción para recibir notificaciones solo sobre los eventos que coincidan con los datos de la etiqueta.
8. Identificar problemas en las notificaciones:
    - Defina etiquetas de trigger.
    - Use la macro {EVENT.TAGS} en la notificación del problema.
    - La notificación del problema contendrá las etiquetas de trigger, lo que facilitará identificar a qué aplicación/servicio pertenece la notificación.
9. Simplificar tareas de configuración usando etiquetas de template:
    - Defina una etiqueta de trigger de template.
    - Los triggers creados a partir de este trigger de template tendrán su etiqueta.
10. Crear triggers con etiquetas desde low-level discovery (LLD):
    - Defina una etiqueta de prototipo de trigger con una macro LLD en el nombre o valor de la etiqueta (por ejemplo, `scope:{#FSNAME}`).
    - Los triggers creados a partir del prototipo de trigger tendrán su etiqueta.
11. Coincidir servicios usando etiquetas de servicio:
    - Defina etiquetas de servicio.
    - Configure acciones de servicio para servicios con etiquetas coincidentes.
    - Además, use etiquetas de servicio para vincular un servicio a un SLA para los cálculos de SLA.
12. Vincular servicios a problemas usando etiquetas de problema de servicio:
    - Defina una etiqueta de problema en la configuración del servicio (por ejemplo, `target:mysql`).
    - Los problemas con una etiqueta coincidente se correlacionarán automáticamente con el servicio, y el estado del servicio cambiará según las reglas de cálculo del estado del servicio configuradas.
13. Suprimir problemas cuando un host está en modo de mantenimiento:
    - Defina etiquetas en la configuración del período de mantenimiento.
    - Los problemas con las etiquetas definidas se suprimirán.
14. Conceder acceso a grupos de usuarios:
    - Defina etiquetas en la configuración del grupo de usuarios.
    - Los usuarios del grupo de usuarios podrán ver solo los problemas con las etiquetas definidas.

Configuración Las etiquetas pueden definirse en una pestaña dedicada, por ejemplo, en configuración de trigger:



Compatibilidad con macros Las macros integradas y las macros de usuario en las etiquetas se resuelven en el momento del evento. Hasta que el evento haya ocurrido, estas macros se mostrarán en Zabbix frontend sin resolver.

Las macros de descubrimiento de bajo nivel se resuelven durante el proceso de descubrimiento.

Las siguientes macros se pueden usar en los nombres y valores de etiquetas de trigger:

- {ITEM.VALUE}, {ITEM.VALUE.AGE}, {ITEM.VALUE.DATE}, {ITEM.VALUE.TIME}, {ITEM.VALUE.TIMESTAMP}, {ITEM.LASTVALUE}, {ITEM.LASTVALUE.AGE}, {ITEM.LASTVALUE.DATE}, {ITEM.LASTVALUE.TIME}, {ITEM.LASTVALUE.TIMESTAMP}, {HOST.HOST}, {HOST.NAME}, {HOST.CONN}, {HOST.DNS}, {HOST.IP}, {HOST.PORT} y {HOST.ID} macros integradas
- {INVENTORY.*} macros integradas (para hacer referencia a valores de inventario del host de uno o varios hosts en una expresión de trigger)
- Macros de usuario y macros de usuario con contexto (el contexto puede incluir macros de descubrimiento de bajo nivel)
- Macros de descubrimiento de bajo nivel (solo en etiquetas de prototipo de trigger)

Las siguientes macros se pueden usar en los nombres y valores de etiquetas de template, host y item/escenario web:

- {HOST.HOST}, {HOST.NAME}, {HOST.CONN}, {HOST.DNS}, {HOST.IP}, {HOST.PORT} y {HOST.ID} macros integradas
- {INVENTORY.*} macros integradas
- Macros de usuario
- Macros de descubrimiento de bajo nivel (solo en etiquetas de prototipo de host y item)

Las siguientes macros se pueden usar en notificaciones basadas en trigger:

- {EVENT.TAGS} y {EVENT.RECOVERY.TAGS} macros integradas (estas macros se resolverán en una lista separada por comas de etiquetas de evento o etiquetas de evento de recuperación)
- {EVENT.TAGSJSON} y {EVENT.RECOVERY.TAGSJSON} macros integradas (estas macros se resolverán en un array JSON que contiene objetos de etiqueta de evento objects u objetos de etiqueta de evento de recuperación)

Extracción de subcadenas en etiquetas de trigger

Se admite la extracción de subcadenas para rellenar el nombre o el valor de la etiqueta, utilizando una función de macro. La función aplica una expresión regular al valor obtenido por la macro compatible. Por ejemplo:

```
{{ITEM.VALUE}.regsub(pattern, output)}
{{ITEM.VALUE}.iregsub(pattern, output)}

{{#LLDMACRO}.regsub(pattern, output)}
{{#LLDMACRO}.iregsub(pattern, output)}
```

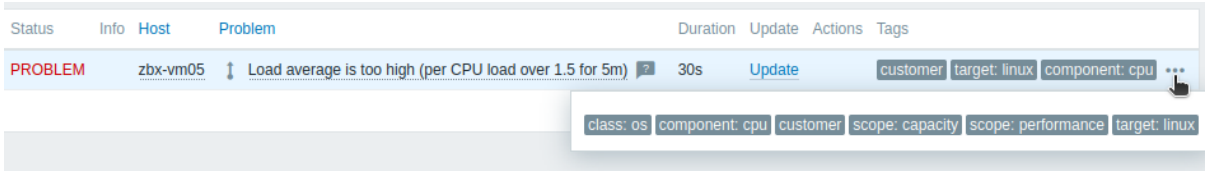
Si el nombre o el valor de la etiqueta supera los 255 caracteres después de la resolución de la macro, se truncará a 255 caracteres.

Véase también: Uso de funciones de macro en macros de descubrimiento de bajo nivel para el etiquetado de eventos.

Visualización de etiquetas de evento Las etiquetas, si están definidas, se pueden ver con los nuevos eventos en:

- *Monitoring* > *Problems*
- *Monitoring* > *Problems* > *Event details*
- *Dashboards* > *Problems* widget

El orden y el número de etiquetas mostradas se determinan mediante las opciones de filtrado *Tag display priority* y *Show tags* en *Monitoring* > *Problems* o en el widget de dashboard *Problems*. Tenga en cuenta que se pueden mostrar como máximo tres etiquetas; si hay más etiquetas, al pasar el cursor sobre los tres puntos se muestran todas las etiquetas en una ventana emergente.



Complex event processing

Overview Complex event processing (CEP) performs rule-based processing for Zabbix events.

For example, you might want to:

- group related events and mark first event in a series as cause (the following as symptoms)
- specify a time-to-live period for events
- specify a maximum number for certain events
- be notified about similar events by location
- be notified if some event happens too often (or not at all) within a time period

In addition to single event processing, you may define **time windows** for event analysis. Within a time window, all matching events form a single context and can be analyzed as a whole.

A sliding time window moves forward with the time. Older events leave the window (are ”evicted”), while new events may enter.

A tumbling time window does not move forward. When the window expires, a new window with the same duration is started. *Cause and symptom grouping* windows are a tumbling time window.

The time window concept presents logical points for executing operations. For example, an event can be transformed or ”enriched” when it enters the window - severity can be raised or lowered, tags can be added or adjusted, suppression can be activated, etc.

Similar operations can be executed when the event leaves the window or when a pattern is found.

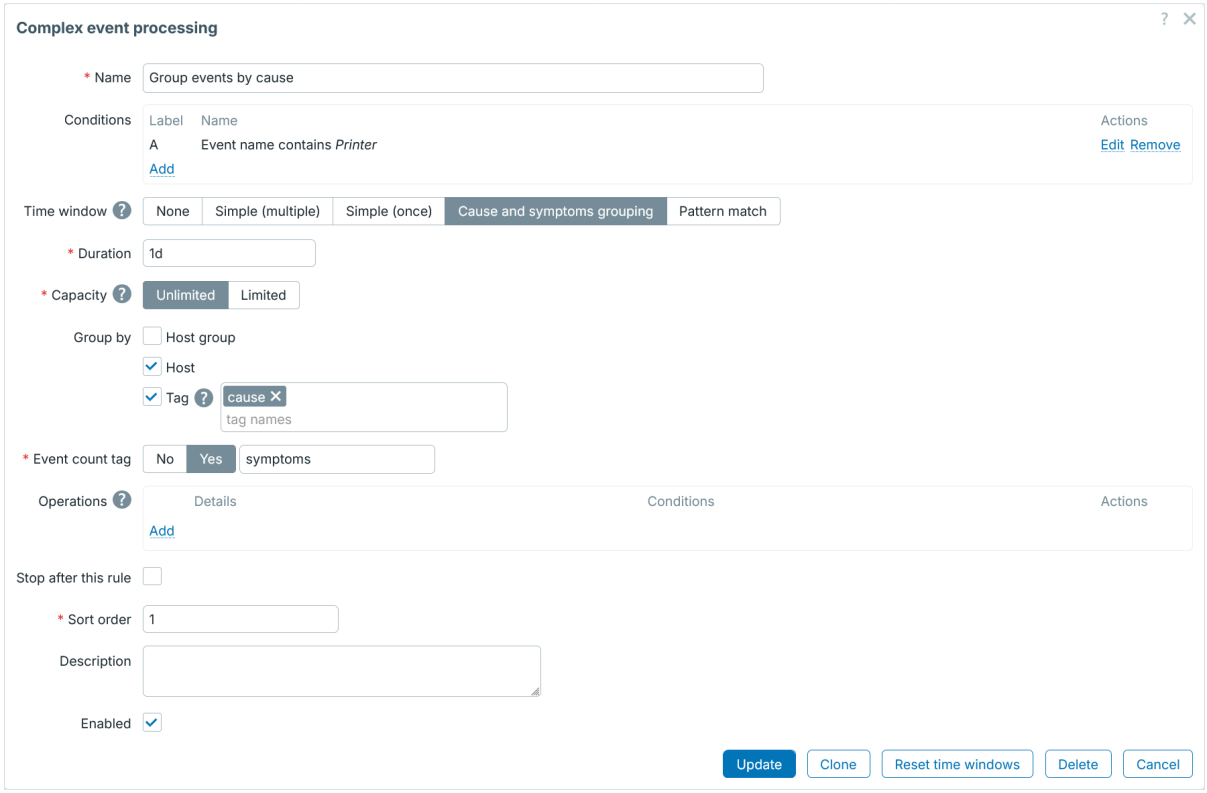
Event stream analysis is defined by CEP rules. A CEP rule consists of a time window (optional), conditions and operations. Operations are tied to an execution point.

Use cases

- Group related events, with the first such event automatically marked as cause (all following as symptoms).
- Enrich event with higher severity/specific tag if too many 401 events happen within a minute (possible bot activity).

Configuration To create a CEP rule in Zabbix frontend, do the following:

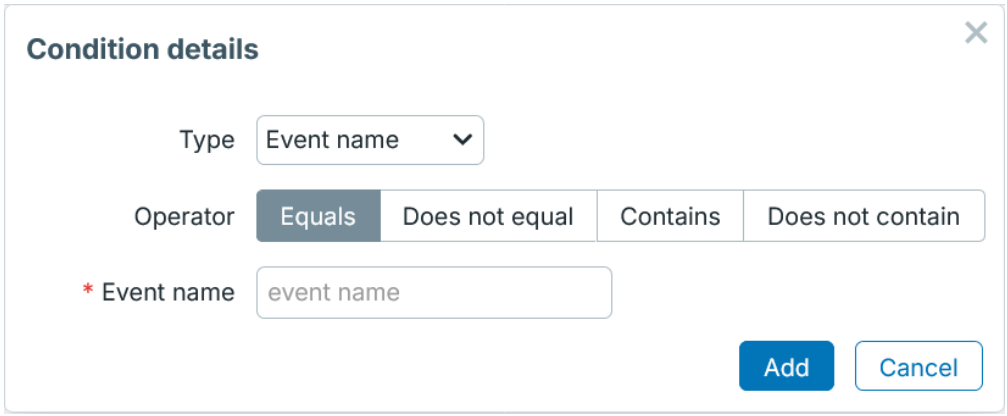
1. Go to *Data collection* > *Event processing*.
2. Click *Create complex event processing* in the upper-right corner of the screen.
3. Enter parameters of the rule in the form.



Parameter	Description
Name	Specify a unique CEP rule name.
Conditions	Specify conditions for associating events with this CEP rule.
Time window	Select the time window type: **None** - process single events with no time window (default) **Simple (multiple)** - process events in a sliding time window (event can be added to multiple windows of this type) **Simple (once)** - process events in a sliding time window (event can be added only to one window of this type) **Cause and symptom grouping** - process first event in time window as cause, consecutive events as symptoms **Pattern match** - analyze sliding time window events based on a JavaScript script. The window is examined every second; events older than the duration are evicted, and the script is then run over the events that remain in it. When the script reports a match, the pattern match operations are executed for every event the window holds; the events stay in the window and are examined again a second later, until they age out. A window can be forcibly closed by the ”Close window” operation.
Duration	Limit the time window duration (in seconds, 365 days max). The window duration is started by the first matching event. In *Simple* and *Pattern match* windows, the window is moving so older events will leave the window (are evicted) after the duration passes. In *Cause and symptom grouping* windows, window duration is a fixed timeframe for setting cause and symptom relationship. The window is started when the first event matches. When the duration expires, the window is closed and a new one is started. No event is evicted when the duration expires; periods run back to back. Time suffixes are supported (`10m`, `1h`, etc). User macros are supported.
Capacity	Specify the event capacity within the time window: **Unlimited** - event count is not limited **Limited** - event count is limited by the specified number Once the limit is surpassed, the latest event is evicted. User macros are supported.
Group by	Group events by host group, host, and/or event tag. This option is available for CEP rules with a time window. If specified, the rule will create multiple windows based on specified criteria; otherwise events will be grouped into a single window group.

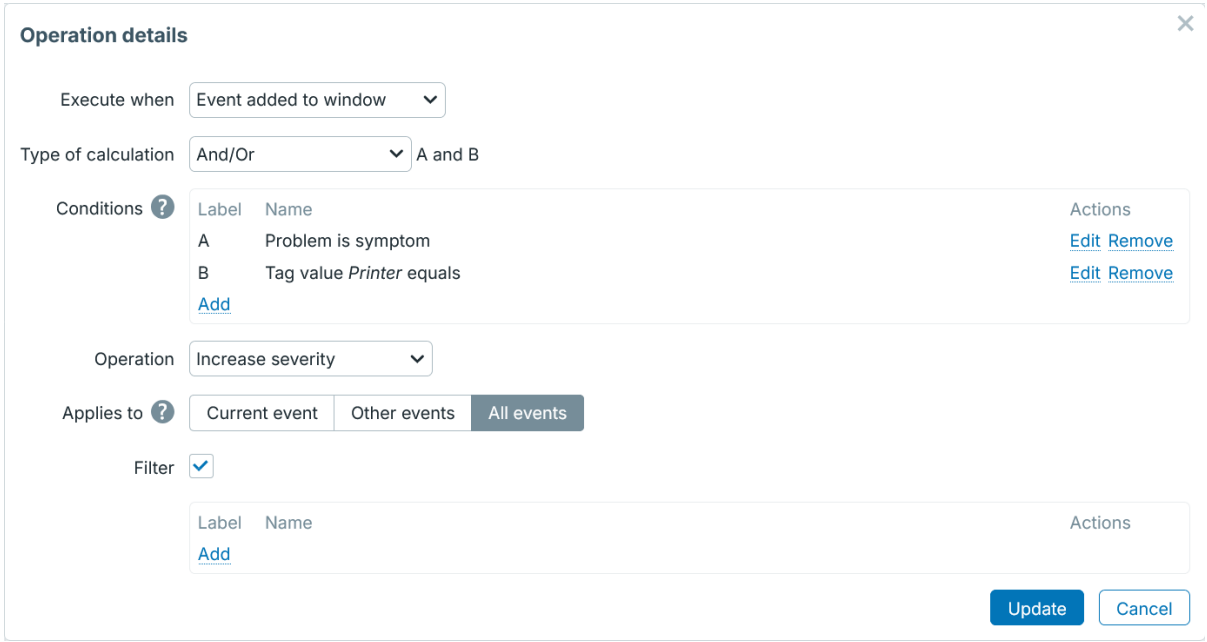
Parameter	Description
Event count tag	Specify name for a tag displaying symptom event count. Symptom event count will be displayed as `<tag name>:<event count>` (for example: `symptoms:2`). This field is only available if *Cause and symptom grouping* is selected as time window type.
Script	Specify the script for identifying event patterns. This field is only available if *Pattern match* is selected as time window type.
Operations	Click on *Add* to specify operations.
Stop after this rule	Mark the checkbox if no further processing is required after this rule. Otherwise, continue with the next CEP rule according to the sort order.
Sort order	Specify position of this CEP rule in the processing order. If the order priority is the same, the rules will be executed alphabetically.
Description	Add the CEP rule description.

CEP rule condition



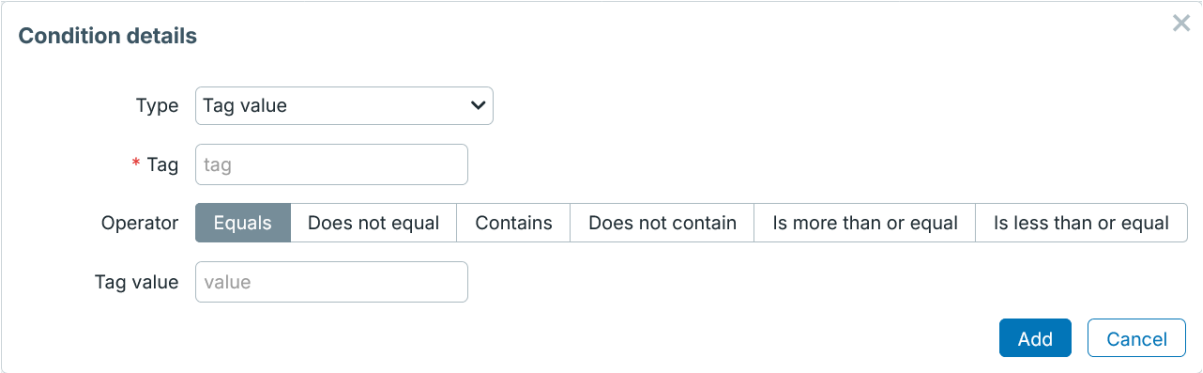
Parameter	Description
Type	Select condition type. Event filtering can be based on event name, tag name, tag value, severity, host group, host (visible name), or time period.

CEP rule operation



Parameter	Description
Execute when	Select execution point: **Event occurred** - when event occurs. This is the only stage where *Discard* operation is supported, since event is not actually created yet and can be discarded. Window-related operations are not available at this point. Changes made to the event in *Event occurred* phase (except closing) are visible to the next CEP rules trying to match this event. For example, if rule A added tag X and rule B has condition ”Tag X exists”, then it will match rule A and B. Of course, rule A should not be configured to stop after matching. **Event added to window** - when event is added to time window. Operations are executed after event is added to matched CEP-rule windows. If window is at its capacity, then *Event evicted* operations are executed instead. **Event evicted** - when event leaves the window. Operations are executed before event is evicted from a window because of duration or capacity. **Event closed** - when event is closed. **Window closed** - when time window is closed. Operations are executed on all window events when window is closed because of *Close window* operation or as expired cause-symptom window. **Pattern matched** - when pattern is matched. Operations are executed on all window events when the pattern match JavaScript returns `true`. Note that the *Event occurred*, *Event added to window* and *Event evicted* have fixed execution order, while *Window closed* and *Pattern matched* operate independently.
Conditions	Select additional conditions for the operation.
Operation	Select operation. **Events**: *Set name* - set event name *Close* - close event *Discard* - discard event. Only available at *Event occurred* stage. *Set severity* - set event severity *Increase severity* - increase event severity *Decrease severity* - decrease event severity *Suppress* - suppress event *Unsuppress* - unsuppress event *Clone first* - clone first event. Useful for producing a summary event based on the first event. Only available at *Event evicted*, *Window closed*, or *Pattern matched* stages. *Clone last* - clone last event. Useful for producing a summary event based on the last event. Only available at *Event evicted*, *Window closed*, or *Pattern matched* stages. For event cloning to work either the source (first/last) event must be resolved or its trigger must have multi-event generation enabled. **Tags**: *Add* - add tag *Set* - set tag and value *Set value* - set tag value *Increase value* - increase tag value *Decrease value* - decrease tag value *Rename* - rename tag *Remove* - remove tag **Window**: *Close (window)* - close event window
Applies to	Select operation scope: **Current event** - apply operation to the current event **Other events** - apply operation to all events except the current one **All events** - apply operation to all events
Filter	Select additional filter for the operation scope. This field is available if *Applies to* is set ”Other events” or ”All events”.

CEP rule operation condition



Parameter	Description
Type	Select condition type. Event filtering can be based on tag name, tag value, problem is opened/closed, problem is symptom/not symptom, first event in time window, last event in time window, problem is suppressed/not suppressed, event is cloned/not cloned.

Correlación de eventos

Descripción general

La correlación de eventos permite correlacionar los eventos de problema con su resolución de una manera muy precisa y flexible.

La correlación de eventos se puede definir:

- En el nivel del trigger: un trigger puede utilizarse para relacionar problemas independientes con su solución;
- Globalmente: los problemas pueden correlacionarse con su solución desde un trigger o método de sondeo diferente mediante reglas de correlación globales.

Correlación de eventos basada en trigger

Descripción general

La correlación de eventos basada en triggers permite correlacionar problemas independientes notificados por un trigger.

Aunque, por lo general, un evento OK puede cerrar todos los eventos de problema creados por un trigger, en algunos casos se necesita un enfoque más detallado. Por ejemplo, al supervisar archivos de registro, es posible que desee detectar determinados problemas en un archivo de registro y cerrarlos individualmente en lugar de cerrarlos todos juntos.

Este es el caso de los triggers cuyo parámetro *PROBLEM event generation mode* está establecido en *Multiple*. Estos triggers se utilizan normalmente para la supervisión de registros, el procesamiento de traps, etc.

En Zabbix es posible relacionar eventos de problema mediante el etiquetado. Las etiquetas se utilizan para extraer valores y crear una identificación para los eventos de problema. Aprovechando esta posibilidad, los problemas también se pueden cerrar individualmente en función de la coincidencia de una etiqueta.

En otras palabras, el mismo trigger puede crear eventos independientes identificados por la etiqueta del evento. Por lo tanto, los eventos de problema se pueden identificar uno por uno y cerrar por separado según la identificación proporcionada por la etiqueta del evento.

Cómo funciona

En la monitorización de logs puede encontrarse con líneas similares a estas:

```
Line1: Service 1 stopped
Line2: Service 2 stopped
Line3: Service 1 was restarted
Line4: Service 2 was restarted
```

La idea de la correlación de eventos es poder asociar el evento de problema de la Line1 con la resolución de la Line3 y el evento de problema de la Line2 con la resolución de la Line4, y cerrar estos problemas uno por uno:

```
Line1: Service 1 stopped
Line3: Service 1 was restarted #problem from Line 1 closed

Line2: Service 2 stopped
Line4: Service 2 was restarted #problem from Line 2 closed
```

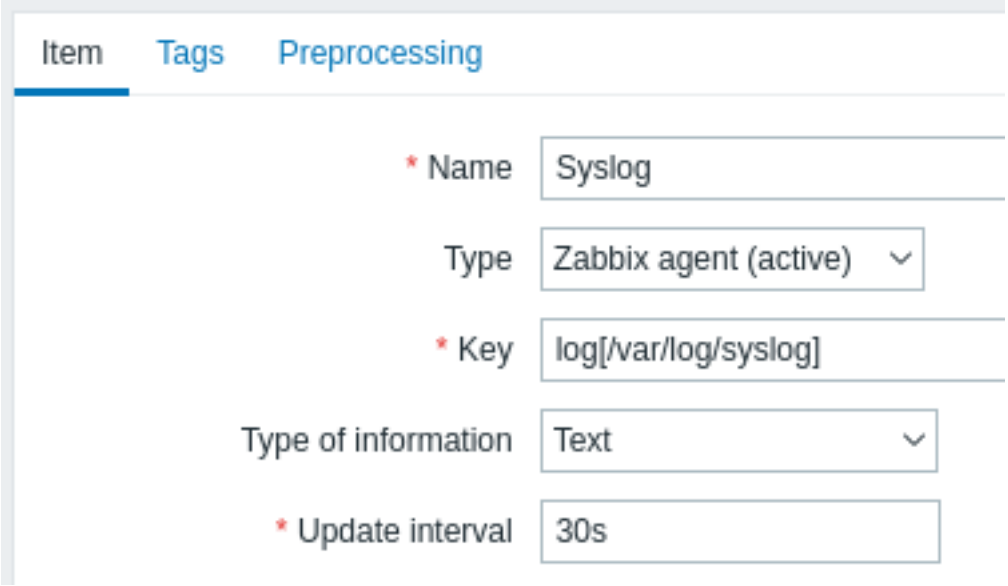
Para ello, necesita etiquetar estos eventos relacionados como, por ejemplo, ”Service 1” y ”Service 2”. Esto puede hacerse aplicando una expresión regular a la línea del log para extraer el valor de la etiqueta. Luego, cuando se crean los eventos, se etiquetan respectivamente como ”Service 1” y ”Service 2” y el problema puede asociarse con la resolución.

Configuración

Item

Para empezar, es posible que desee configurar un item que supervise un archivo de registro, por ejemplo:

```
log[/var/log/syslog]
```



Una vez configurado el item, espere un minuto para que se apliquen los cambios de configuración y luego vaya a Latest data para asegurarse de que el item haya comenzado a recopilar datos.

Trigger

Con el item en funcionamiento, necesitas configurar el trigger. Es importante decidir qué entradas del archivo de registro merecen atención. Por ejemplo, la siguiente expresión de trigger buscará una cadena como ’Stopping’ para señalar posibles problemas:

```
find(/My host/log[/var/log/syslog],,"regexp","Stopping")=1
```

Attention:
Para asegurarte de que cada línea que contenga la cadena ”Stopping” se considere un problema, configura también el *Problem event generation mode* en la configuración del trigger en ’Multiple’.

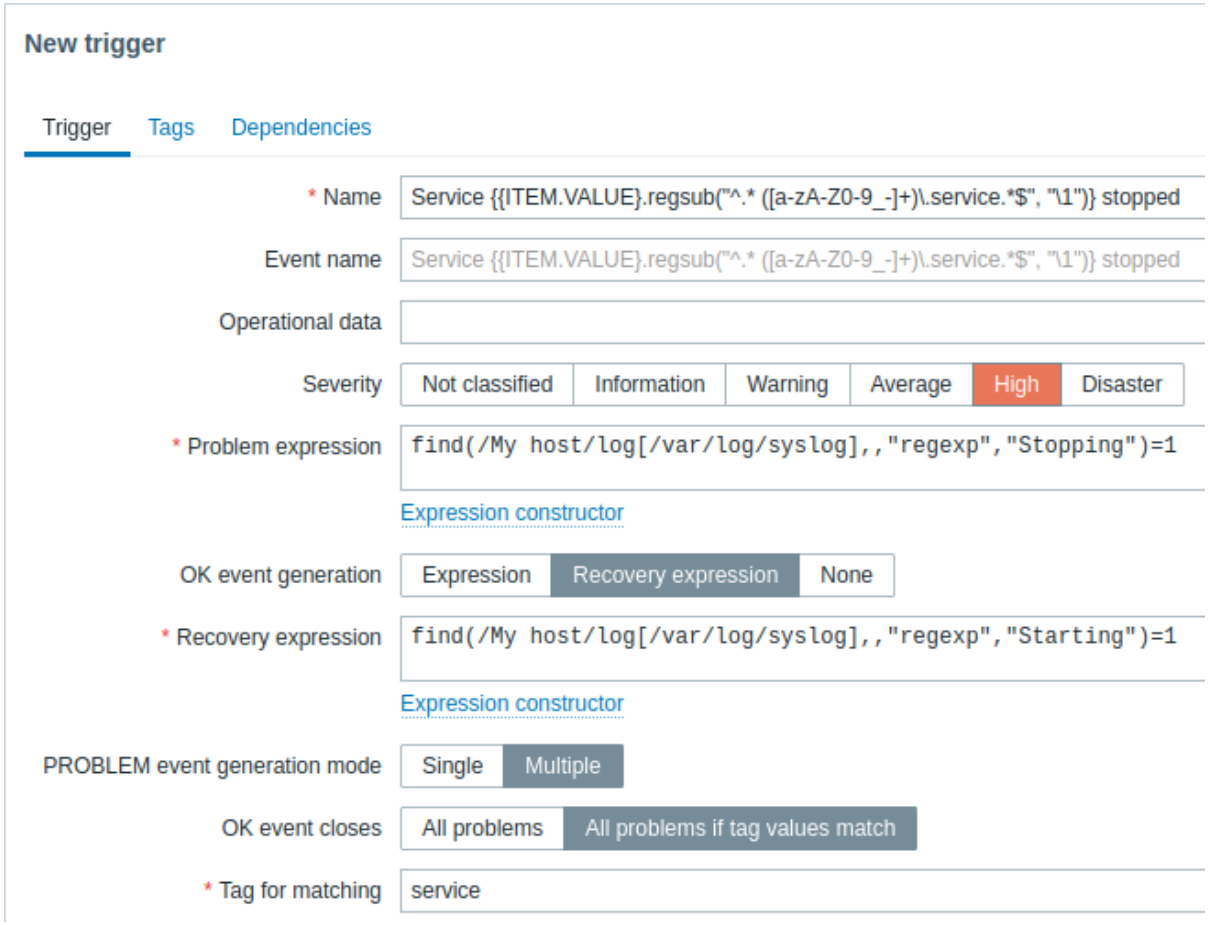
Luego define una expresión de recuperación. La siguiente expresión de recuperación resolverá todos los problemas si se encuentra una línea de registro que contenga la cadena ”Starting”:

```
find(/My host/log[/var/log/syslog],,"regexp","Starting")=1
```

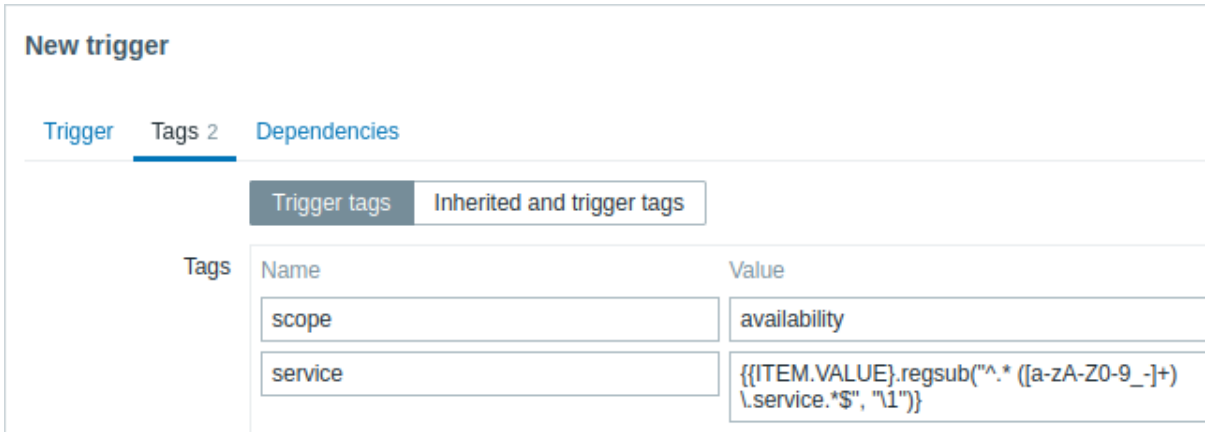
Como no queremos eso, es importante asegurarse de alguna manera de que los problemas raíz correspondientes se cierren, no solo todos los problemas. Ahí es donde el etiquetado puede ayudar.

Los problemas y las resoluciones se pueden emparejar especificando una etiqueta en la configuración del trigger. Deben realizarse los siguientes ajustes:

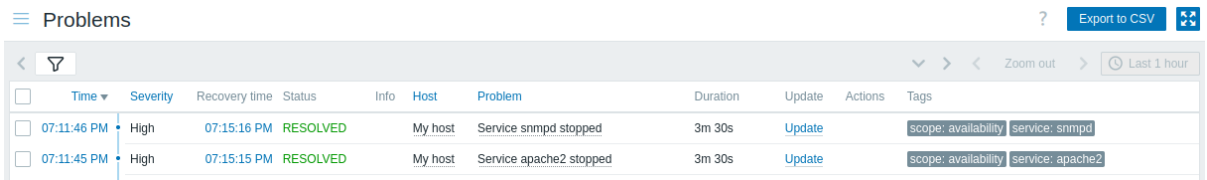
- *Problem event generation mode*: Multiple
- *OK event closes*: All problems if tag values match
- Introduce el nombre de la etiqueta para la coincidencia de eventos



- configura las tags para extraer valores de etiqueta de las líneas de registro



Si se configura correctamente, podrás ver eventos de problema etiquetados por aplicación y emparejados con su resolución en *Monitoring* > *Problems*.



Warning:
Dado que es posible una configuración incorrecta, cuando se puedan crear etiquetas de evento similares para problemas **no relacionados**, revisa los casos descritos a continuación.

- Los macros indexados siempre hacen referencia al campo *Expression* de la configuración del trigger, no a la *Recovery expression*. Por ejemplo, en un evento de recuperación, **{ITEM.VALUE1}** resolverá al último valor del primer item **en la expresión del problema** en el momento de la recuperación. Si la expresión de recuperación se basa en un item diferente y el valor del item de la expresión del problema cambia para el momento de la recuperación, los eventos tendrán etiquetas

diferentes y no se correlacionarán.

- Con dos aplicaciones escribiendo mensajes de error y de recuperación en el mismo archivo de registro, un usuario puede decidir usar dos etiquetas de *service* en el mismo trigger con diferentes valores de etiqueta, usando expresiones regulares separadas en los valores de etiqueta para extraer los nombres de, por ejemplo, service A y service B del macro {ITEM.VALUE} (por ejemplo, cuando los formatos de mensaje difieren). Sin embargo, esto puede no funcionar como se esperaba si no hay coincidencia con las expresiones regulares. Las expresiones regulares que no coincidan producirán valores de etiqueta vacíos y un único valor de etiqueta vacío tanto en los eventos de problema como en los de OK es suficiente para correlacionarlos. Así, un mensaje de recuperación de service A podría cerrar accidentalmente un mensaje de error de service B.
- Las etiquetas reales y los valores de etiqueta solo se vuelven visibles cuando se dispara un trigger. Si la expresión regular utilizada no es válida, se reemplaza silenciosamente por una cadena *UNKNOWN*. Si se omite el evento de problema inicial con un valor de etiqueta *UNKNOWN*, pueden aparecer eventos OK posteriores con el mismo valor de etiqueta *UNKNOWN* que podrían cerrar eventos de problema que no deberían haber cerrado.
- Si un usuario usa el macro {ITEM.VALUE} sin funciones de macro como valor de etiqueta, se aplica la limitación de 255 caracteres. Cuando los mensajes de registro son largos y los primeros 255 caracteres no son específicos, esto también puede dar lugar a etiquetas de evento similares para problemas no relacionados.

Correlación global de eventos

Descripción general

La correlación global de eventos permite abarcar todas las métricas supervisadas por Zabbix y crear correlaciones.

Es posible correlacionar eventos creados por triggers completamente diferentes y aplicarles las mismas operaciones. Al crear reglas de correlación inteligentes, es posible evitar miles de notificaciones repetitivas y centrarse en las causas raíz de un problema.

La correlación global de eventos es un mecanismo potente que permite desvincularse de la lógica de problemas y resoluciones basada en un solo trigger. Hasta ahora, un solo evento de problema era creado por un trigger y dependíamos de ese mismo trigger para resolver el problema. No podíamos resolver un problema creado por un trigger con otro trigger. Sin embargo, con la correlación de eventos basada en etiquetas de eventos, sí podemos hacerlo.

Por ejemplo, un trigger de registro puede informar de problemas de la aplicación, mientras que un trigger de sondeo puede informar de que la aplicación está activa y funcionando. Aprovechando las etiquetas de eventos, puede etiquetar el trigger de registro como *status:down* y el trigger de sondeo como *status:up*. A continuación, en una regla de correlación global, puede relacionar estos triggers y asignar una operación adecuada a esta correlación, como cerrar los eventos antiguos.

En otro caso de uso, la correlación global puede identificar triggers similares y aplicarles la misma operación. ¿Qué ocurriría si pudiéramos obtener un solo informe de problema por cada problema de puerto de red? No sería necesario informar de todos ellos. Esto también es posible con la correlación global de eventos.

La correlación global de eventos se configura mediante **reglas de correlación**. Una regla de correlación define cómo se emparejan los nuevos eventos de problema con los eventos de problema existentes y qué hacer en caso de coincidencia (cerrar el nuevo evento o cerrar los eventos antiguos coincidentes generando los eventos OK correspondientes). Si un problema se cierra mediante la correlación global, se informa de ello en la columna *Info* de *Monitoring* > *Problems*.

La configuración de reglas de correlación global está disponible únicamente para usuarios con nivel de Super Admin.

> **Attention:**
> La correlación de eventos debe configurarse con mucho cuidado, ya que puede afectar negativamente al rendimiento del procesamiento de eventos o, si se configura incorrectamente, cerrar más eventos de los previstos (en el peor de los casos, podrían cerrarse todos los eventos de problema).

Para configurar las reglas de correlación global de forma **segura**, tenga en cuenta las siguientes recomendaciones importantes:

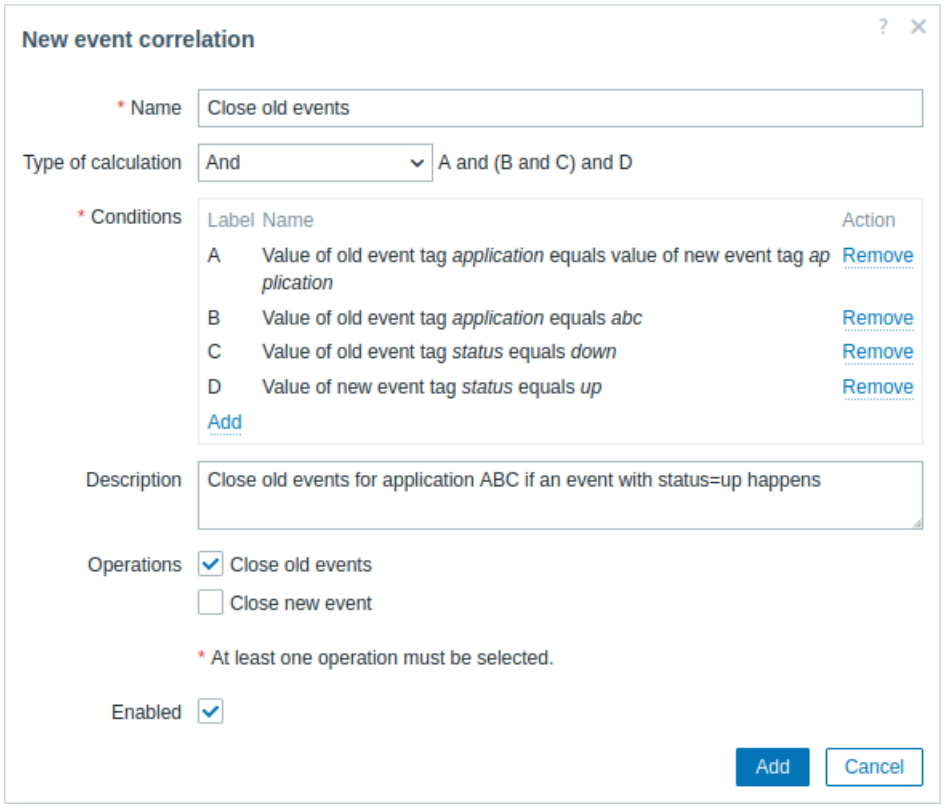
- Reduzca el alcance de la correlación. Establezca siempre una etiqueta única para el nuevo evento que se empareja con eventos antiguos y utilice la condición de correlación *New event tag name*.
- Añada una condición explícita para los eventos antiguos al utilizar **Close old events**. Añada siempre al menos una condición de *Old event* (por ejemplo, *Old event tag name*, *Old event tag value* o *Event tag pair*) al seleccionar **Close old events**; de lo contrario, la regla podría coincidir con eventos de problema existentes no relacionados y cerrarlos (en el peor de los casos, todos los problemas). Prefiera *Event tag pair* para hacer coincidir valores en tiempo de ejecución (host:port, id de sesión, etc.) y limite aún más la coincidencia por host o grupo de hosts cuando sea posible.
- Evite utilizar nombres de etiquetas comunes que puedan terminar utilizándose en diferentes configuraciones de correlación.
- Mantenga limitado el número de reglas de correlación a las que realmente necesite.

Consulte también: known issues.

Configuración

Para configurar reglas de correlación de eventos globalmente:

1. Vaya a *Data collection* > *Event correlation*.
2. Haga clic en *Create event correlation* a la derecha (o en el nombre de la correlación para editar una regla existente).
3. Introduzca los parámetros de la regla de correlación en el formulario.

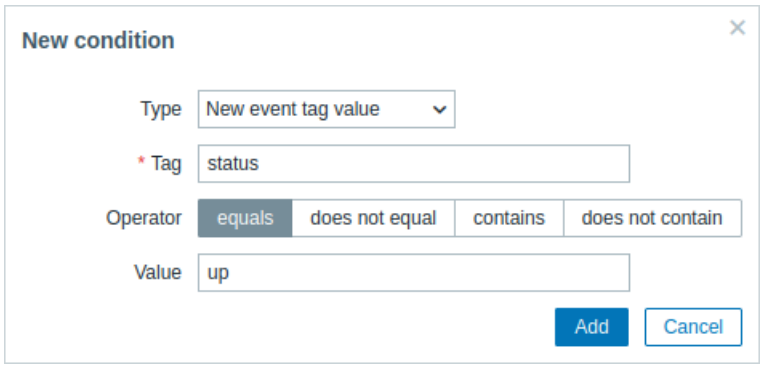


Todos los campos de entrada obligatorios están marcados con un asterisco rojo.

Parameter	Description
Name	Nombre único de la regla de correlación.
Type of calculation	Las siguientes opciones de cálculo de condiciones están disponibles: **And** - todas las condiciones deben cumplirse **Or** - basta con que se cumpla una condición **And/Or** - AND con distintos tipos de condición y OR con el mismo tipo de condición **Custom expression** - una fórmula de cálculo definida por el usuario para evaluar las condiciones de la acción. Debe incluir todas las condiciones (representadas por letras mayúsculas A, B, C, ...) y puede incluir espacios, tabulaciones, paréntesis ( ), **and** (sensible a mayúsculas y minúsculas), **or** (sensible a mayúsculas y minúsculas), **not** (sensible a mayúsculas y minúsculas).
Conditions	Lista de condiciones. Consulte más abajo los detalles sobre cómo configurar una condición.
Description	Descripción de la regla de correlación.

Parameter	Description
Operations	Marque la casilla de la operación que se realizará cuando el evento se correlacione. Las siguientes operaciones están disponibles: **Close old events** - cerrar eventos antiguos cuando ocurre un nuevo evento. Añada siempre una condición basada en el evento antiguo cuando use la operación *Close old events* o podrían cerrarse todos los problemas existentes. **Close new event** - cerrar el nuevo evento cuando ocurra. **Warning!** No deje vacías las condiciones de evento antiguo/nuevo al usar *Close old events*/*Close new event*. Si selecciona la operación *Close old events* sin añadir una condición que coincida con el evento antiguo, Zabbix puede hacer coincidir todos los eventos antiguos existentes y cerrarlos. Añada siempre una condición explícita de evento antiguo (por ejemplo, *Old event tag name* o *Event tag pair*) al usar *Close old events*. Por ejemplo, una regla que use solo una *New event condition* y la operación *Close old events* hará coincidir todos los eventos antiguos que cumplan los criterios del evento antiguo (ausentes), cerrando en la práctica problemas antiguos.
Enabled	Si marca esta casilla, la regla de correlación se habilitará.

Para configurar los detalles de una nueva condición, haga clic en Add en el bloque Conditions. Se abrirá una ventana emergente donde podrá editar los detalles de la condición.



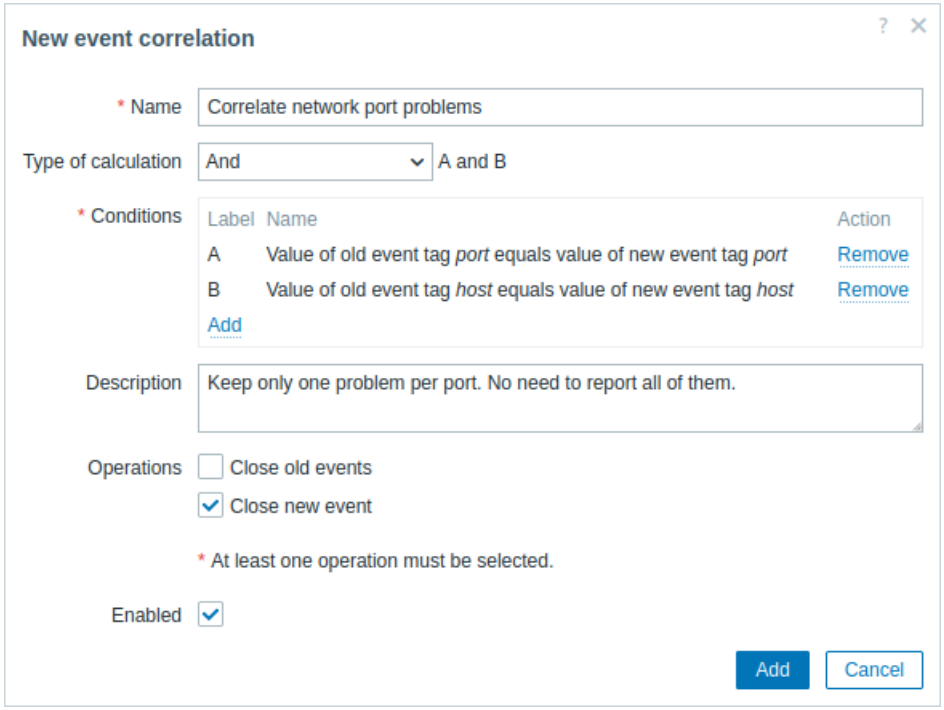
Parameter	Description
New condition	Seleccione una condición para correlacionar eventos. *Note* que si no se especifica ninguna condición de evento antiguo, todos los eventos antiguos pueden coincidir y cerrarse. Del mismo modo, si no se especifica ninguna condición de evento nuevo, todos los eventos nuevos pueden coincidir y cerrarse. Las siguientes condiciones están disponibles: **Old event tag name** - especifique el nombre de la etiqueta del evento antiguo para la coincidencia. **New event tag name** - especifique el nombre de la etiqueta del evento nuevo para la coincidencia. **New event host group** - especifique el grupo de hosts del evento nuevo para la coincidencia. **Event tag pair** - especifique el nombre de la etiqueta del evento nuevo y el nombre de la etiqueta del evento antiguo para la coincidencia. En este caso habrá coincidencia si los **valores** de las etiquetas en ambos eventos coinciden. Los *nombres* de las etiquetas no tienen por qué coincidir. Esta opción es útil para hacer coincidir valores en tiempo de ejecución, que pueden no conocerse en el momento de la configuración (consulte también Example). **Old event tag value** - especifique el nombre y el valor de la etiqueta del evento antiguo para la coincidencia, usando los siguientes operadores: *equals* - tiene el valor de la etiqueta del evento antiguo *does not equal* - no tiene el valor de la etiqueta del evento antiguo *contains* - contiene la cadena en el valor de la etiqueta del evento antiguo *does not contain* - no contiene la cadena en el valor de la etiqueta del evento antiguo **New event tag value** - especifique el nombre y el valor de la etiqueta del evento nuevo para la coincidencia, usando los siguientes operadores: *equals* - tiene el valor de la etiqueta del evento nuevo *does not equal* - no tiene el valor de la etiqueta del evento nuevo *contains* - contiene la cadena en el valor de la etiqueta del evento nuevo *does not contain* - no contiene la cadena en el valor de la etiqueta del evento nuevo

Warning:
Debido a que es posible una configuración incorrecta, cuando puedan crearse etiquetas de evento similares para problemas **no relacionados**, revise los casos descritos a continuación.

- Las etiquetas y los valores de etiqueta reales solo se hacen visibles cuando se dispara un trigger. Si la expresión regular utilizada no es válida, se reemplaza silenciosamente por una cadena *UNKNOWN*. Si se omite el evento de problema inicial con un valor de etiqueta *UNKNOWN*, pueden aparecer posteriormente eventos OK con el mismo valor de etiqueta *UNKNOWN* que podrían cerrar eventos de problema que no deberían haberse cerrado.
- Si un usuario utiliza la macro {ITEM.VALUE} sin funciones de macro como valor de la etiqueta, se aplica la limitación de 255 caracteres. Cuando los mensajes de registro son largos y los primeros 255 caracteres no son específicos, esto también puede dar lugar a etiquetas de evento similares para problemas no relacionados.

Ejemplo

Evite eventos de problema repetitivos del mismo puerto de red.



Esta regla global de correlación correlacionará los problemas si existen valores de etiquetas *host* y *port* en el trigger y son iguales en el evento original y en el nuevo.

La operación cerrará los nuevos eventos de problema en el mismo puerto de red, manteniendo abierto solo el problema original.

Descubrimiento

Resumen El descubrimiento proporciona formas automáticas de añadir hosts y configurar la monitorización sin necesidad de una configuración manual. Zabbix admite múltiples métodos de descubrimiento para adaptarse a diferentes escenarios.

Métodos de descubrimiento

- Descubrimiento de red: detecta automáticamente los hosts y servicios disponibles en su red.
- Autorregistro de agent activo: permite que los agents activos se registren automáticamente en el server.
- Descubrimiento de bajo nivel: crea automáticamente items, triggers, gráficos y hosts para entidades descubiertas, como sistemas de archivos o interfaces de red, mediante prototipos y reglas de descubrimiento anidadas.

Descubrimiento de red

Descripción general

Zabbix ofrece una funcionalidad de descubrimiento automático de red que es eficaz y muy flexible.

Con el descubrimiento de red correctamente configurado, puede:

- Acelerar la implementación de Zabbix.
- Simplificar la administración.
- Usar Zabbix en entornos que cambian rápidamente sin una administración excesiva.

El descubrimiento de red de Zabbix se basa en la siguiente información:

- Rangos de IP
- Disponibilidad de servicios externos (FTP, SSH, WEB, POP3, IMAP, TCP, etc)
- Información recibida del agent de Zabbix (solo se admite el modo sin cifrar)
- Información recibida del agent SNMP

NO proporciona el descubrimiento de la topología de red.

El descubrimiento de red básicamente consta de dos fases: descubrimiento y acciones.

Discovery

Zabbix analiza periódicamente los rangos de IP definidos en las reglas de descubrimiento de red. La frecuencia de la comprobación se puede configurar de forma individual para cada regla.

Cada regla tiene definido un conjunto de comprobaciones de servicio que se ejecutan para el rango de IP.

Las reglas de discovery son procesadas por el discovery manager. El discovery manager crea un trabajo por cada regla con una lista de tareas (comprobaciones de red). Las comprobaciones de red se ejecutan en paralelo por los discovery workers disponibles (el número se puede configurar en el frontend para cada regla). La excepción son las comprobaciones SNMPv3, que son procesadas por un único worker. Solo las comprobaciones con la misma IP y puerto se programan de forma secuencial, porque algunos dispositivos no aceptan conexiones paralelas en el mismo puerto.

El tamaño de la cola de comprobaciones de red está limitado a 2000000 o aproximadamente 4 GB de memoria. Si la cola se llena, la regla de discovery se omitirá y se imprimirá un mensaje de advertencia en el log. Puede usar el `zabbix[discovery_queue]` item interno para supervisar el número de comprobaciones de discovery en la cola.

Las comprobaciones de discovery se procesan de forma independiente de las demás comprobaciones. Si alguna comprobación no encuentra un servicio (o falla), las demás comprobaciones seguirán procesándose.

Note:
Si una regla de discovery se cambia durante la ejecución, la ejecución actual de discovery se abortará.

Cada comprobación de un servicio y un host (IP) realizada por el módulo de discovery de red genera un evento de discovery.

Event	Check of service result
Service Discovered	El servicio está 'up' después de haber estado 'down' o cuando se descubre por primera vez.
Service Up	El servicio está 'up', después de haber estado ya 'up'.
Service Lost	El servicio está 'down' después de haber estado 'up'.
Service Down	El servicio está 'down', después de haber estado ya 'down'.
Host Discovered	Al menos un servicio de un host está 'up' después de que todos los servicios de ese host estuvieran 'down', o se descubre un servicio que pertenece a un host no registrado.
Host Up	Al menos un servicio de un host está 'up', después de que al menos un servicio ya estuviera 'up'.
Host Lost	Todos los servicios de un host están 'down' después de que al menos uno estuviera 'up'.
Host Down	Todos los servicios de un host están 'down', después de que ya estuvieran 'down'.

Acciones

Los eventos de descubrimiento pueden ser la base de acciones relevantes, tales como:

- Enviar notificaciones
- Añadir/eliminar hosts
- Habilitar/deshabilitar hosts
- Añadir hosts a un grupo
- Eliminar hosts de un grupo
- Añadir etiquetas a un host
- Eliminar etiquetas de un host
- Vincular un template a hosts/desvincular un template de hosts
- Ejecutar scripts remotos

Estas acciones pueden configurarse en función del tipo de dispositivo, IP, estado, tiempo de actividad/inactividad, etc. Para obtener todos los detalles sobre la configuración de acciones para eventos basados en descubrimiento de red, consulte las páginas de operaciones y condiciones de la acción.

Dado que las acciones de descubrimiento de red se basan en eventos, se activarán tanto cuando un host descubierto esté en línea como cuando esté fuera de línea. Se recomienda encarecidamente añadir una condición de acción *Estado de descubrimiento: activo* para evitar que acciones como *Añadir host* se activen en eventos de *Servicio perdido*/*Servicio inactivo*. De lo contrario, si un host descubierto se elimina manualmente, seguirá generando eventos de *Servicio perdido*/*Servicio inactivo* y se volverá a crear durante el siguiente ciclo de descubrimiento.

Note:
La vinculación de templates a un host descubierto fallará en conjunto si alguno de los templates vinculables tiene una entidad única (por ejemplo, clave de item) que sea igual a una entidad única (por ejemplo, clave de item) que ya exista en el host o en otro de los templates vinculables.

Creación de host

Se añade un host si se selecciona la operación *Add host*. También se añade un host, incluso si falta la operación *Add host*, si selecciona operaciones que dan lugar a acciones sobre un host. Estas operaciones son:

- Enable host.
- Disable host.
- Add host to a host group.
- Link template to a host.

Los hosts creados se añaden al grupo *Discovered hosts* (de forma predeterminada, configurable en *Administration* > *General* > *Other*). Si desea que los hosts se añadan a otro grupo, agregue una operación *Remove from host groups* (especificando "Discovered hosts") y también agregue una operación *Add to host groups* (especificando otro host group), porque un host debe pertenecer a un host group.

La dirección IP del dispositivo descubierto, junto con la fuente de descubrimiento (Zabbix server, Zabbix proxy o grupo de proxy) y el tipo de interfaz, se utiliza como criterio para encontrar un host en el sistema. Si ya existe un host con la misma dirección IP, tipo de interfaz y fuente de descubrimiento, ese host será el destino para ejecutar las operaciones. Si la fuente de descubrimiento difiere, la entidad descubierta se trata como un host diferente y se puede crear un nuevo host.

Si la dirección IP del host descubierto cambia o la interfaz se elimina, se creará un nuevo host en el siguiente descubrimiento.

Nomenclatura de hosts

Al agregar hosts, el nombre del host es el resultado de una búsqueda inversa de DNS o la dirección IP si la búsqueda inversa falla. La búsqueda se realiza desde el Zabbix server o el Zabbix proxy, dependiendo de cuál esté realizando el descubrimiento. Si la búsqueda falla en el proxy, no se vuelve a intentar en el server. Si ya existe un host con ese nombre, el siguiente host tendrá **_2** añadido al nombre, luego **_3** y así sucesivamente.

También es posible omitir la búsqueda de DNS/IP y, en su lugar, utilizar el valor de un item como nombre del host, por ejemplo:

- Puede descubrir múltiples servidores con Zabbix agent en ejecución utilizando un item de Zabbix agent para el descubrimiento y asignarles nombres adecuados automáticamente, basándose en el valor de cadena devuelto por este item
- Puede descubrir múltiples dispositivos de red SNMP utilizando un item de SNMP agent para el descubrimiento y asignarles nombres adecuados automáticamente, basándose en el valor de cadena devuelto por este item

Si el nombre del host se ha establecido utilizando el valor de un item, no se actualiza durante las siguientes comprobaciones de descubrimiento. Si no es posible establecer el nombre del host utilizando el valor de un item, se utiliza el valor predeterminado (nombre DNS).

Si ya existe un host con la dirección IP descubierta y la fuente de descubrimiento (Zabbix server, proxy o grupo de proxy) no ha cambiado, no se crea un nuevo host. Si la fuente de descubrimiento difiere, la entidad descubierta se trata como distinta y se puede crear un nuevo host. Sin embargo, si la acción de descubrimiento contiene operaciones (enlazar template, añadir a grupo de hosts, etc.), se realizan en el host existente que coincide por dirección IP, tipo de interfaz y fuente de descubrimiento.

Eliminación de host

Los hosts descubiertos por una regla de descubrimiento de red se eliminan automáticamente de *Monitoreo* > *Descubrimiento* si una entidad descubierta ya no está en el rango de IP de la regla. Los hosts se eliminan inmediatamente.

Creación de interfaces al añadir hosts

Cuando se añaden hosts como resultado del descubrimiento de red, se crean interfaces según estas reglas:

- Los servicios detectados; por ejemplo, si una comprobación SNMP tuvo éxito, se creará una interfaz SNMP.
- Si un host respondió tanto a solicitudes de Zabbix agent como de SNMP, se crearán ambos tipos de interfaces.
- Si los criterios de unicidad son Zabbix agent o datos devueltos por SNMP, la primera interfaz encontrada para un host se creará como la predeterminada. Las otras direcciones IP se ignorarán, a menos que la interfaz predeterminada deje de funcionar. En caso de que la interfaz predeterminada esté caída, se utilizará como reemplazo de la interfaz predeterminada la dirección IP más pequeña de las otras direcciones IP. Las condiciones de la acción (como Host IP) no afectan a la adición de interfaces. *Tenga en cuenta* que esto funcionará si todas las interfaces son descubiertas por la misma regla de descubrimiento. Si una regla de descubrimiento diferente descubre una interfaz distinta del mismo host, se añadirá un host adicional.
- Si un host respondió solo a comprobaciones de agent, se creará únicamente con una interfaz agent. Si más adelante empezara a responder a SNMP, se añadirían interfaces SNMP adicionales.
- Si inicialmente se crearon 3 hosts separados, descubiertos mediante el criterio de unicidad "IP", y luego se modifica la regla de descubrimiento de modo que los hosts A, B y C tengan el mismo resultado de criterio de unicidad, B y C se crean como interfaces adicionales para A, el primer host. Los hosts individuales B y C permanecen. En *Monitoring* > *Discovery*, las interfaces añadidas se mostrarán en la columna "Discovered device", en fuente negra y con sangría, pero la columna "Monitored host" solo mostrará A, el primer host creado. "No se mide" el "Uptime/Downtime" para las IP que se consideran interfaces adicionales.

Cambio de la configuración de proxy

Los hosts descubiertos por diferentes proxies no siempre se tratan como hosts distintos. Las comprobaciones de descubrimiento y unicidad dependen de la estructura del grupo de proxy: cuando un proxy ejecuta una regla de descubrimiento y crea un host, ese host se agrega al grupo de proxy padre del proxy, no se asigna al propio proxy. Cuando Zabbix evalúa la unicidad de IP durante el descubrimiento, comprueba los hosts monitorizados por el grupo de proxy padre. Los hosts monitorizados por proxies individuales dentro de ese grupo (incluido el proxy que ejecutó el descubrimiento) se ignoran para la comprobación de unicidad, lo que puede provocar hosts duplicados si varios proxies monitorizan subredes superpuestas.

Aunque este comportamiento permite que el descubrimiento funcione en rangos de IP superpuestos utilizados por diferentes subredes, cambiar el proxy asignado a una subred ya monitorizada es más complicado porque los cambios de proxy deben aplicarse de forma coherente a los hosts descubiertos y a la pertenencia al grupo de proxy padre para evitar duplicados.

Por ejemplo, los pasos para reemplazar el proxy en una regla de descubrimiento:

1. Deshabilite la regla de descubrimiento.
2. Sincronice la configuración del proxy.
3. Reemplace el proxy en la regla de descubrimiento.
4. Reemplace el proxy para todos los hosts descubiertos por esta regla (asegúrese de actualizar los hosts del grupo de proxy padre y cualquier host monitorizado por proxies individuales de ese grupo para evitar duplicados).
5. Habilite la regla de descubrimiento.

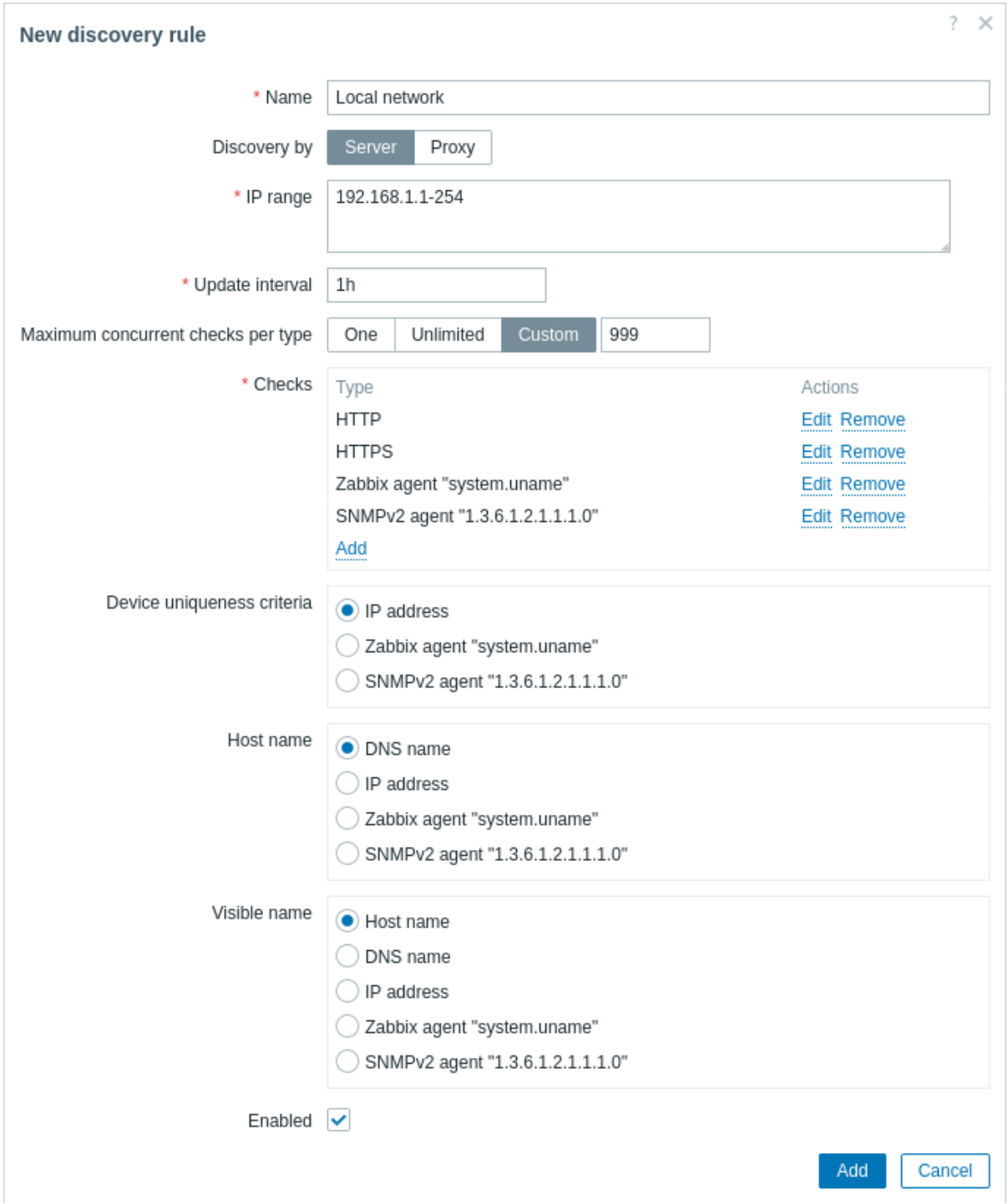
Configuración de una regla de descubrimiento de red

Resumen

Para configurar una regla de descubrimiento de red utilizada por Zabbix para descubrir hosts y servicios:

1. Vaya a *Data collection* > *Discovery*.
2. Haga clic en *Create discovery rule* (o en el nombre de la regla para editar una existente).
3. Edite los atributos de la regla de descubrimiento.

Atributos de la regla



Todos los campos de entrada obligatorios están marcados con un asterisco rojo.

Parameter	Description
Name	Nombre único de la regla. Por ejemplo, ”Red local”.
Discovery by	El descubrimiento se realiza mediante: **Server** - por Zabbix server **Proxy** - por Zabbix proxy (seleccionado en el campo de nombre del proxy)
IP range	El rango de direcciones IP para el descubrimiento. Puede tener los siguientes formatos: IP única: 192.168.1.33 Rango de direcciones IP: 192.168.1-10.1-255. El rango está limitado por el número total de direcciones cubiertas (menos de 64K). Máscara IP: 192.168.4.0/24 máscaras IP compatibles: /16 - /30 para direcciones IPv4 /112 - /128 para direcciones IPv6 Lista: 192.168.1.1-255, 192.168.2.1-100, 192.168.2.200, 192.168.4.0/24 Este campo admite espacios, tabulación y varias líneas.

Parameter	Description
Update interval	Este parámetro define con qué frecuencia Zabbix ejecutará la regla. El intervalo se mide después de que finaliza la ejecución de la instancia de descubrimiento anterior, por lo que no hay solapamiento. Se admiten sufijos de tiempo, por ejemplo, 30s, 1m, 2h, 1d. Se admiten macros de usuario. *Nota* que si se usa una macro de usuario y su valor cambia (por ejemplo, 1w → 1h), la siguiente comprobación se ejecutará según el valor anterior (muy en el futuro con los valores del ejemplo).
Maximum concurrent checks per type	Establezca el número máximo de hilos de descubrimiento (workers) por comprobación de servicio para el procesamiento en paralelo de las comprobaciones de descubrimiento: **One** - un hilo **Unlimited** - número ilimitado de hilos (pero no más que en el parámetro StartDiscoverers) **Custom** - establezca un número personalizado de hilos (0-999). Las comprobaciones de servicio asíncronas SNMPv1 y SNMPv2 se procesan de forma concurrente por los workers disponibles. Todas las reglas de descubrimiento con comprobaciones de servicio asíncronas SNMPv3 son procesadas por un solo worker debido a las particularidades de la implementación de libsnmp; por lo tanto, aumentar el número de workers no incrementará la velocidad de descubrimiento.
Checks	Zabbix utilizará esta lista de comprobaciones para el descubrimiento. Haga clic en Add para configurar una nueva comprobación en una ventana emergente. Comprobaciones compatibles: SSH, LDAP, SMTP, FTP, HTTP, HTTPS, POP, NNTP, IMAP, TCP, Telnet, Zabbix agent, SNMPv1 agent, SNMPv2 agent, SNMPv3 agent, ICMP ping. Un descubrimiento basado en protocolo utiliza la funcionalidad **net.tcp.service[]** para probar cada host, excepto SNMP, que consulta un OID SNMP. Zabbix agent se prueba consultando un item en modo no cifrado. Consulte agent items para obtener más detalles. El parámetro 'Ports' puede ser uno de los siguientes: Puerto único: 22 Rango de puertos: 22-45 Lista: 22-45,55,60-70 Desde Zabbix 7.0, todas las comprobaciones de servicio se realizan de forma asíncrona, excepto las comprobaciones LDAP. Desde Zabbix 7.0, la comprobación HTTP/HTTPS se realiza mediante libcurl. Si Zabbix server/proxy se compila sin libcurl, entonces las comprobaciones HTTP funcionarán como en versiones anteriores (es decir, como comprobaciones TCP), pero las comprobaciones HTTPS no funcionarán.
Device uniqueness criteria	Los criterios de unicidad pueden ser: **IP address** - no procese varios dispositivos de IP única. Si ya existe un dispositivo con la misma IP, se considerará ya descubierto y no se añadirá un nuevo host. **<discovery check>** - comprobación de Zabbix agent o SNMP agent. Tenga en cuenta que los criterios de unicidad utilizados durante el descubrimiento no son los mismos que la identificación de host en el sistema al realizar acciones. Los criterios de unicidad durante el descubrimiento definen si dos o más dispositivos descubiertos son iguales (o diferentes), mientras que solo la dirección IP es el criterio para la identificación de host en Zabbix (consulte Host creation).
Host name	Establezca el nombre técnico del host creado usando: **DNS name** - nombre DNS (predeterminado) **IP address** - dirección IP **<discovery check>** - valor de cadena recibido de la comprobación de descubrimiento (por ejemplo, comprobación de Zabbix agent, SNMP agent) Véase también: Host naming.
Visible name	Establezca el nombre visible del host creado usando: **Host name** - nombre técnico del host (predeterminado) **DNS name** - nombre DNS **IP address** - dirección IP **<discovery check>** - valor de cadena recibido de la comprobación de descubrimiento (por ejemplo, comprobación de Zabbix agent, SNMP agent) Véase también: Host naming.
Enabled	Con la casilla marcada, la regla está activa y será ejecutada por Zabbix server. Si no está marcada, la regla no está activa. No se ejecutará.

Excediendo el límite de descriptores de archivo

En caso de un gran número de comprobaciones concurrentes, es posible agotar el límite de descriptores de archivo para el discovery manager.

El número de descriptores de archivo necesarios para la detección equivale al número de discovery workers * 1000. De forma predeterminada, hay 5 discovery workers, mientras que el límite blando del sistema es aproximadamente 1024.

Si se aproxima a este límite, Zabbix reducirá el número predeterminado de comprobaciones concurrentes por tipo para cada worker y escribirá una advertencia en el archivo de registro. Pero, si el usuario ha establecido un valor superior para *Maximum concurrent checks per type* que el valor calculado por Zabbix, Zabbix utilizará el valor definido por el usuario para un worker.

Un escenario de la vida real

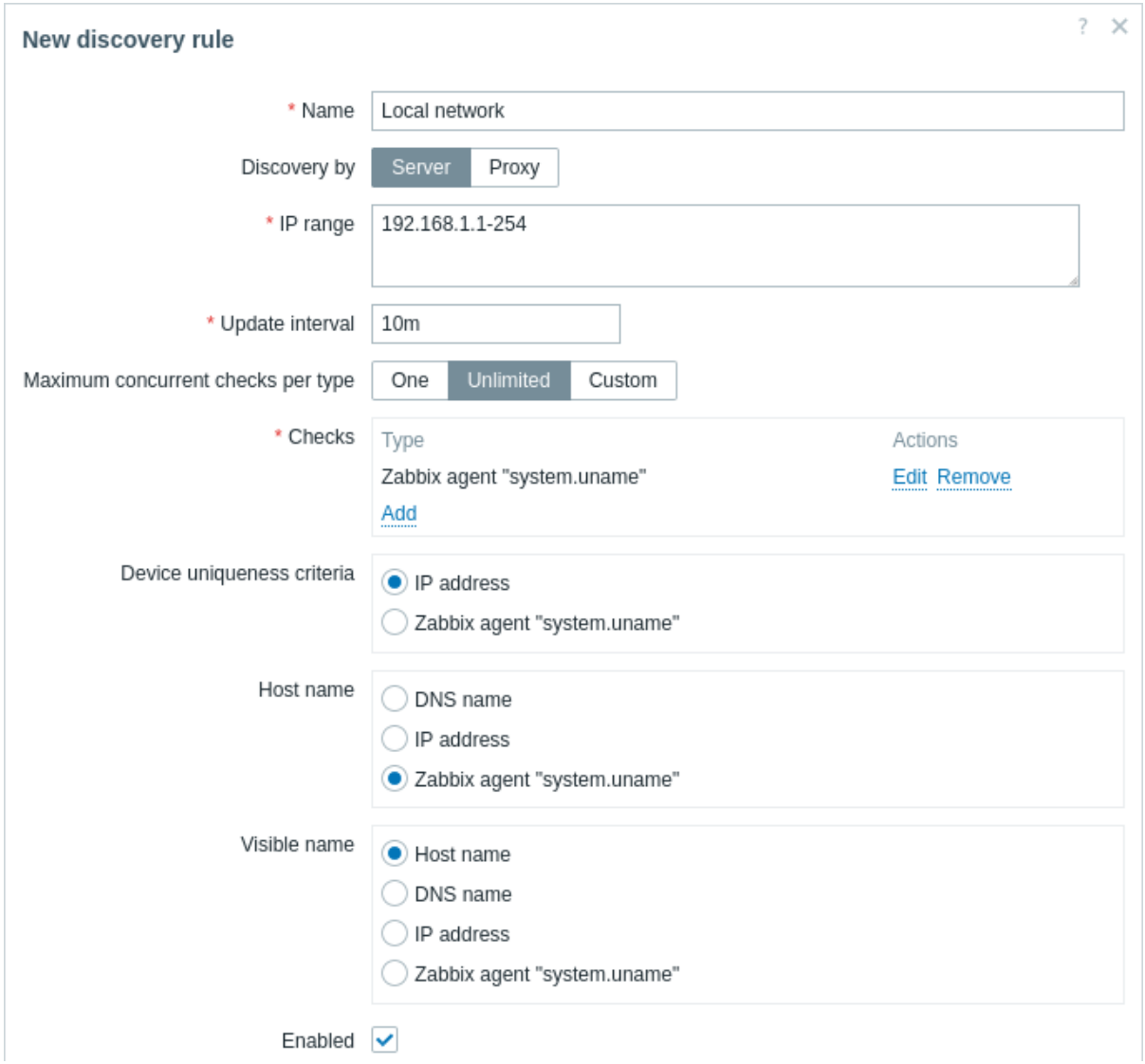
En este ejemplo, queremos configurar el descubrimiento de red para la red local con un rango de IP de 192.168.1.1-192.168.1.254.

En nuestro escenario queremos:

- descubrir aquellos equipos que tengan el agente de Zabbix en ejecución
- ejecutar el descubrimiento cada 10 minutos
- añadir un equipo a la monitorización si el tiempo de actividad del equipo es superior a 1 hora
- eliminar equipos si el tiempo de inactividad del equipo es superior a 24 horas
- añadir equipos Linux al grupo ”Servidores Linux”
- añadir equipos Windows al grupo ”Servidores Windows”
- usar la plantilla *Linux* para equipos Linux
- usar la plantilla *Windows* para equipos Windows

Paso 1

Definir una regla de descubrimiento de red para nuestro rango de IPs.



Zabbix intentará descubrir equipos en el rango de IPs 192.168.1.1-192.168.1.254 conectándose a los agentes de Zabbix y obteniendo el valor de la clave **system.uname**. El valor recibido del agente puede usarse para nombrar los equipos y también para

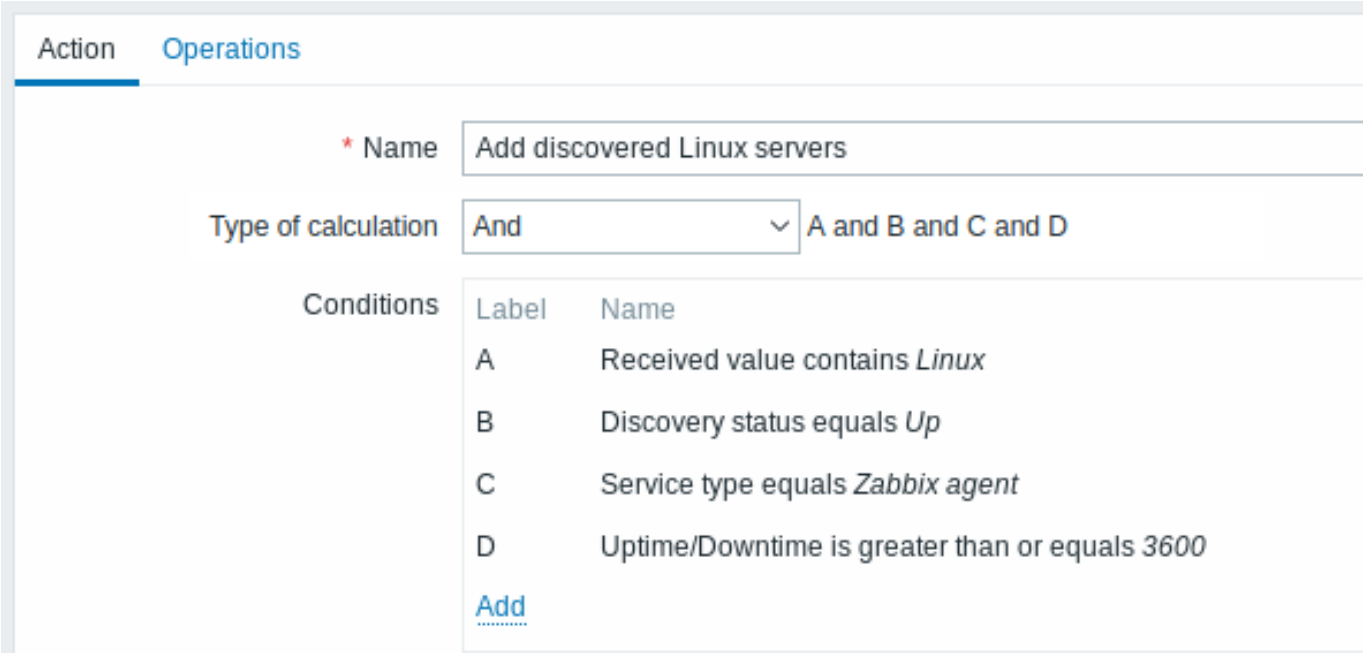
aplicar diferentes acciones para diferentes sistemas operativos. Por ejemplo, vincular los servidores Windows a la plantilla *Windows*, los servidores Linux a la plantilla *Linux*.

La regla se ejecutará cada 10 minutos.

Cuando se añade esta regla, Zabbix iniciará automáticamente el descubrimiento y la generación de eventos basados en el descubrimiento para su posterior procesamiento.

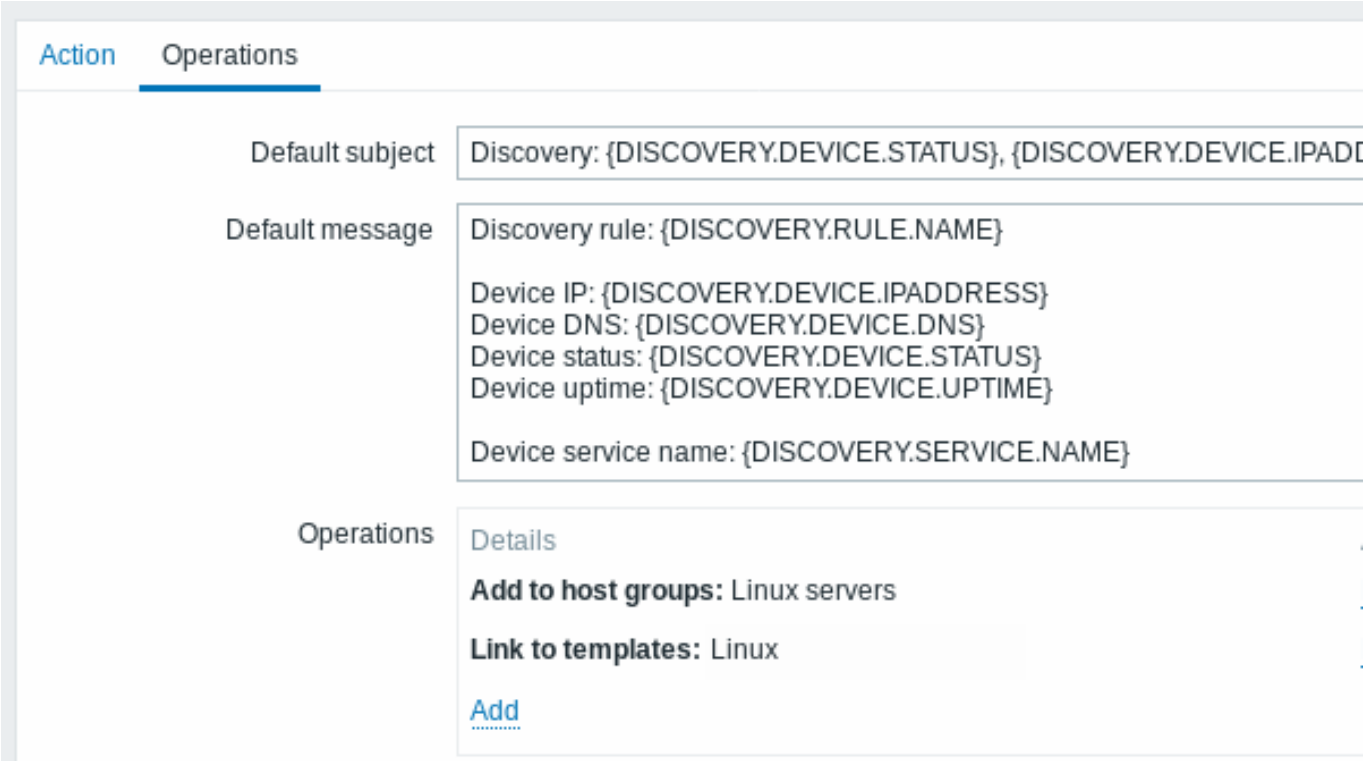
Paso 2

Definir una acción de descubrimiento para agregar los servidores Linux descubiertos al grupo/plantilla correspondiente.



La acción se activará si:

- el servicio ”Zabbix agent” está ”arriba”
- el valor de system.uname (la clave del agente Zabbix que usamos en la definición de la regla) contiene ”Linux”
- El tiempo de actividad es de 1 hora (3600 segundos) o más



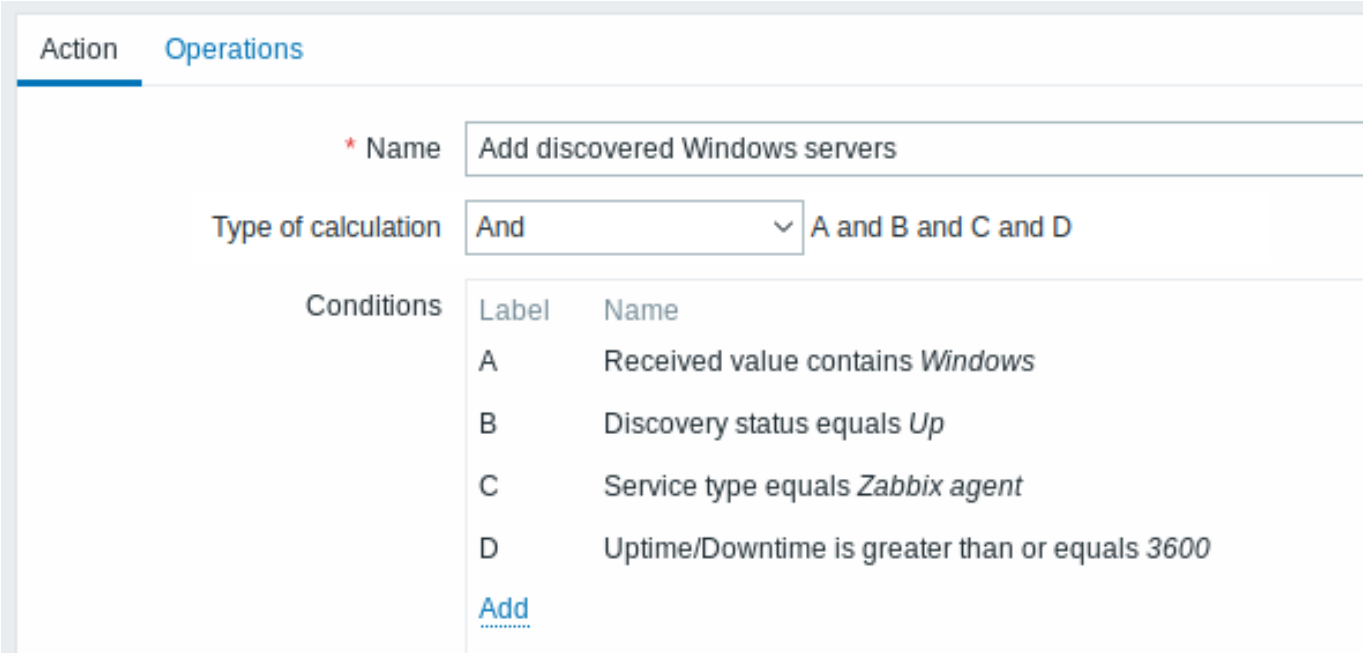
La acción ejecutará las siguientes operaciones:

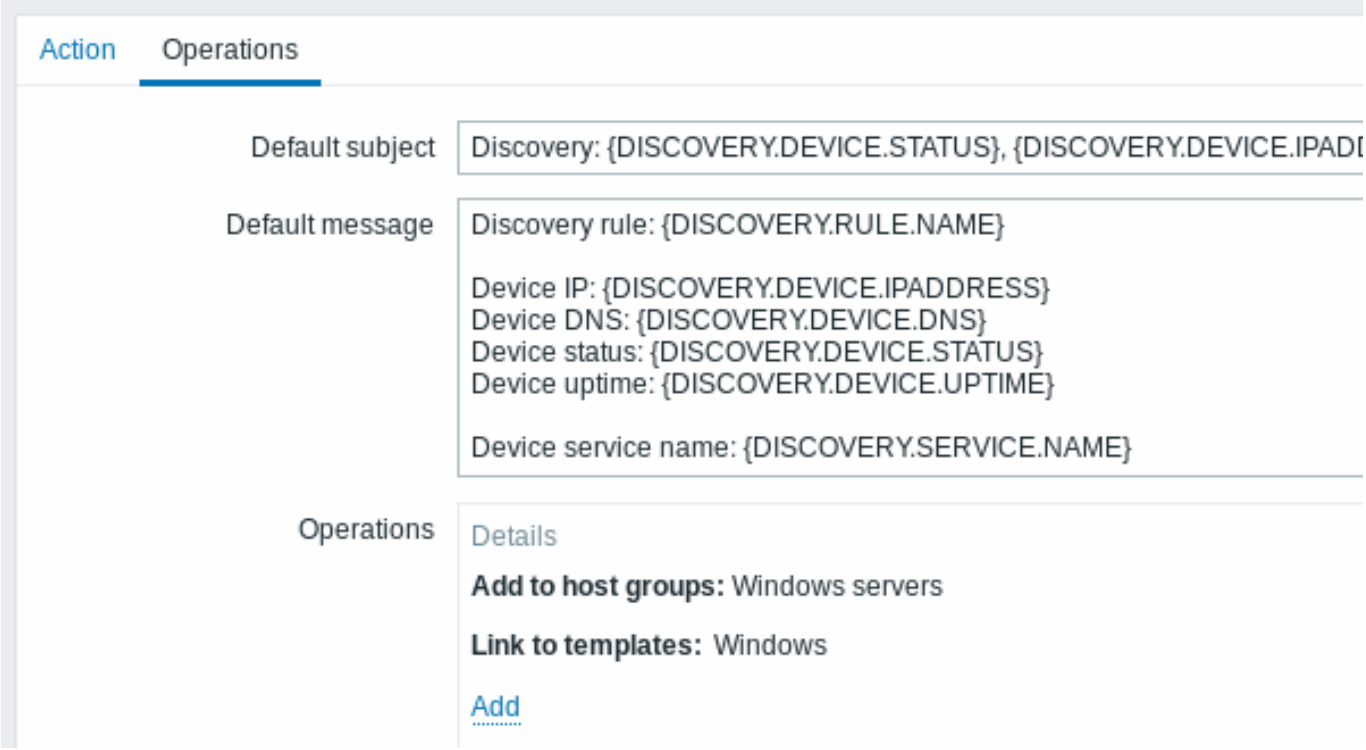
- agregar el equipo descubierto al grupo ”Linux servers” (y también agregar el equipo si no se había agregado previamente)

- vincular el equipo a la plantilla *Linux*. Zabbix comenzará automáticamente a monitorizar el equipo utilizando métricas y disparadores de la plantilla ”Linux”.

Paso 3

Definir una acción de descubrimiento para añadir los servidores Windows descubiertos al grupo/plantilla correspondiente.





Paso 4

Definir una acción de descubrimiento para eliminar servidores perdidos.

Un servidor será eliminado si el servicio ”Zabbix agent” está ’caído’ durante más de 24 horas (86400 segundos).

Autoregistro del agent activo

Overview

Es posible permitir la autoregistración del agent activo de Zabbix, tras lo cual el server puede empezar a monitorizarlos. De este modo, se pueden añadir nuevos hosts para su monitorización sin configurarlos manualmente en el server.

La autoregistración puede producirse cuando un agent activo previamente desconocido solicita comprobaciones.

Esta función puede ser muy útil para la monitorización automática de nuevos nodos en la nube. En cuanto tenga un nuevo nodo en Zabbix Cloud, se iniciará automáticamente la recopilación de datos de rendimiento y disponibilidad del host.

La autoregistración del agent activo también admite la monitorización de hosts añadidos con comprobaciones pasivas. Cuando el agent activo solicita comprobaciones, si tiene definidos los parámetros de configuración `ListenIP` o `ListenPort` en el archivo de configuración, estos se envían al server. Si se especifican varias direcciones IP, se envía la primera al server.

El server, al añadir el nuevo host autoregistrado, utiliza la dirección IP y el puerto recibidos para configurar el agent. Si no se recibe ningún valor de dirección IP, se utiliza el de la conexión entrante. Si no se recibe ningún valor de puerto, se usa 10050.

Es posible especificar que el host deba autoregistrarse con un nombre DNS como interfaz predeterminada del agent.

La autoregistración se vuelve a ejecutar:

- si cambia la información de metadatos del host:
  - debido a que cambió HostMetadata y se reinició el agent
  - debido a que cambió el valor devuelto por HostMetadataItem
- para hosts creados manualmente sin metadatos
- si un host se cambia manualmente para ser monitorizado por otro Zabbix proxy
- si la autoregistración del mismo host proviene de un nuevo Zabbix proxy

El intervalo de latido de la autoregistración del agent activo para Zabbix server y Zabbix proxy es de 120 segundos. Por lo tanto, en caso de que se elimine un host descubierto, la autoregistración se volverá a ejecutar en 120 segundos.

Configuración

Especificar el server

Asegúrese de que el server de Zabbix esté identificado en el archivo de configuración del agent (`zabbix_agentd.conf`):

```
ServerActive=10.0.0.1
```

A menos que defina específicamente un *Hostname* en `zabbix_agentd.conf`, el server utilizará el nombre de host del sistema donde se encuentra el agent para asignar el nombre al host.
En Linux, puede obtener el nombre de host del sistema ejecutando el comando `hostname`.

Si *Hostname* está definido en la configuración del agent de Zabbix como una lista de hosts separados por comas, se crearán hosts para todos los nombres de host indicados.

Reinicie el agent después de realizar cualquier cambio en el archivo de configuración.

Acción para el autoregistro de agentes activos

Cuando el server recibe una solicitud de autoregistro de un agent, llama a una acción. Debe configurarse una acción con el origen de evento "Autoregistration" para el autoregistro de agentes.

> **Note:**
> No es necesario configurar el descubrimiento de red para que los agentes activos se autoregistren.

En el frontend de Zabbix:

1. Vaya a *Alerts* > *Actions*.
2. Seleccione *Autoregistration actions*.
3. Haga clic en *Create action*.
4. En la pestaña *Action*, asigne un nombre a la acción.
5. Opcionalmente, especifique condiciones. En las condiciones para el nombre del host o los metadatos del host, puede utilizar una coincidencia de subcadena o una coincidencia mediante expresión regular. Si va a utilizar el tipo de condición *Host metadata*, consulte Uso de metadatos del host.
6. En la pestaña *Operations*, añada las operaciones pertinentes, como *Add host*, *Add to host group* (por ejemplo, *Discovered hosts*), *Link templates*, etc.

> **Note:**
> Si es probable que los hosts que se autoregistren solo sean compatibles con la monitorización activa (por ejemplo, hosts protegidos por un firewall frente a su Zabbix server), puede que desee crear un template específico, como *Template_Linux-active*, para vincularlo.

Los hosts creados se añaden al grupo *Discovered hosts* (de forma predeterminada, configurable en *Administration* > *General* > *Other*). Si desea que los hosts se añadan a otro grupo, añada una operación *Remove from host group* (especificando "Discovered

hosts") y también una operación *Add to host group* (especificando otro grupo de hosts), ya que un host debe pertenecer a un grupo de hosts.

Autoregistro seguro

Es posible utilizar un método seguro de autoregistro configurando la autenticación basada en PSK con conexiones cifradas.

El nivel de cifrado se configura globalmente en *Administración* > *General* > *Autoregistro*. Es posible seleccionar sin cifrado, cifrado TLS con autenticación PSK o ambas opciones (de modo que algunos hosts puedan registrarse sin cifrado y otros mediante una conexión cifrada).

Zabbix server verifica la autenticación mediante PSK antes de añadir un host. Si la verificación es correcta, el host se añade y *Conexiones desde/hacia el host* se establecen en «PSK» únicamente con la identidad y la clave precompartida que coincidan con las de la configuración global de autoregistro.

Attention:
Para garantizar la seguridad del autoregistro en instalaciones que utilizan proxies, debe habilitarse el cifrado entre Zabbix server y el proxy.

Uso de DNS como interfaz predeterminada

HostInterface y HostInterfaceItem parámetros de configuración permiten especificar un valor personalizado para la interfaz del host durante el registro automático.

Más concretamente, resultan útiles si el host debe registrarse automáticamente con un nombre DNS como interfaz predeterminada del agent en lugar de su dirección IP. En ese caso, el nombre DNS debe especificarse o devolverse como valor de los parámetros HostInterface o HostInterfaceItem. Si el valor de uno de estos parámetros cambia —por ejemplo, de una dirección IP a un nombre DNS o viceversa—, la interfaz predeterminada del host registrado automáticamente se actualizará en consecuencia. Esta actualización se aplica al host existente, no mediante la creación de uno nuevo. Para enviar el nuevo valor, es necesario reiniciar el agent para que vuelva a iniciar el proceso de registro automático.

Note:
Si los parámetros HostInterface o HostInterfaceItem no están configurados, se utiliza el parámetro listen_dns. Este valor se determina realizando una búsqueda DNS inversa de la dirección IP del agent. Si el DNS inverso no está configurado correctamente o devuelve un nombre no válido, el registro automático puede ser incorrecto o fallar debido a un valor de interfaz no válido.

Uso de los metadatos del host

Cuando el agent envía una solicitud de autorregistro al server, envía su nombre de host. En algunos casos (por ejemplo, nodos de Amazon Cloud), un nombre de host no es suficiente para que el server de Zabbix diferencie los hosts descubiertos. Los metadatos del host se pueden utilizar opcionalmente para enviar otra información desde un agent al server.

Los metadatos del host se configuran en el archivo de configuración del agent (`zabbix_agentd.conf`). Hay 2 formas de especificar los metadatos del host en el archivo de configuración:

```
HostMetadata
HostMetadataItem
```

Consulte la descripción de las opciones en el enlace anterior.

El parámetro HostMetadataItem puede devolver hasta 65535 puntos de código UTF-8. Un valor más largo se truncará.

Tenga en cuenta que, en MySQL, la longitud máxima efectiva en caracteres será menor si el valor devuelto contiene caracteres multibyte. Por ejemplo, un valor que contenga únicamente caracteres de 3 bytes estará limitado a un total de 21844 caracteres, mientras que un valor que contenga únicamente caracteres de 4 bytes estará limitado a 16383 símbolos.

Attention:
Se produce un intento de autorregistro cada vez que un agent activo envía una solicitud al server para actualizar las comprobaciones activas. El intervalo entre solicitudes se especifica en el parámetro RefreshActiveChecks del agent. La primera solicitud se envía inmediatamente después de reiniciar el agent.

Ejemplos

Autoregistro por SO usando HostMetadata

Supongamos que desea que los hosts se autoregistren en el servidor Zabbix. Tiene agentes Zabbix activos (ver la sección "Configuración" arriba) en su red. Hay hosts Windows y Linux en su red y tiene disponibles las plantillas "Linux by Zabbix agent" y "Windows by Zabbix agent" en su frontend de Zabbix. Por lo tanto, al registrar el host, le gustaría que la plantilla adecuada de

Linux/Windows se aplique al host que se está registrando. Por defecto, solo el nombre del host se envía al servidor durante el autoregistro, lo que podría no ser suficiente. Para asegurarse de que la plantilla adecuada se aplique al host, debe usar metadatos del host.

Configuración del frontend

Lo primero que hay que hacer es configurar el frontend. Cree 2 acciones. La primera acción:

- Nombre: Autoregistro de host Linux
- Condiciones: Los metadatos del host contienen *Linux*
- Operaciones: Vincular templates: Linux por Zabbix agent

> **Note:**
> En este caso puede omitir la operación "Añadir host". Vincular un template a un host requiere añadir primero el host, por lo que el server lo hará automáticamente.

La segunda acción:

- Nombre: Autoregistro de host Windows
- Condiciones: Los metadatos del host contienen *Windows*
- Operaciones: Vincular templates: Windows por Zabbix agent

Configuración del agent

Ahora necesita configurar los agents. Agregue la siguiente línea a los archivos de configuración del agent:

```
HostMetadataItem=system.uname
```

De esta manera, se asegura de que los metadatos del host contengan "Linux" o "Windows" dependiendo del host en el que se esté ejecutando el agent. Un ejemplo de metadatos del host en este caso:

```
Linux: Linux server3 3.2.0-4-686-pae #1 SMP Debian 3.2.41-2 i686 GNU/Linux
Windows: Windows WIN-0PXGGSTYNHO 6.0.6001 Windows Server 2008 Service Pack 1 Intel IA-32
```

No olvide reiniciar el agent después de realizar cualquier cambio en el archivo de configuración.

Usar HostMetadata para controlar el registro automático y las plantillas

Paso 1 - Proteger el registro con HostMetadata

Utilizar metadatos de host para permitir cierta protección básica contra el registro de hosts no deseados.

Configuración del frontend

Cree una acción en el frontend, utilizando algún código secreto difícil de adivinar para evitar hosts no deseados:

- Nombre: Acción de autorregistro Linux
- Condiciones:
  - Tipo de cálculo: AND
  - Condición (A): Los metadatos del host contienen //Linux//
  - Condición (B): Los metadatos del host contienen //21df83bf21bf0be663090bb8d4128558ab9b95fba66a6dbf834f8b91ae5e08ae//
- Operaciones:
  - Enviar mensaje a los usuarios: Admin a través de todos los medios
  - Añadir a los grupos de hosts: Servidores Linux
  - Vincular templates: Linux por Zabbix agent

Tenga en cuenta que este método por sí solo no proporciona una protección sólida porque los datos se transmiten en texto sin formato. Se requiere recargar la caché de configuración para que los cambios tengan efecto inmediato.

Configuración del agent

Agregue la siguiente línea al archivo de configuración del agent:

```
HostMetadata=Linux    21df83bf21bf0be663090bb8d4128558ab9b95fba66a6dbf834f8b91ae5e08ae
```

donde "Linux" es una plataforma y el resto de la cadena es el texto secreto difícil de adivinar.

No olvide reiniciar el agent después de realizar cualquier cambio en el archivo de configuración.

Paso 2 - Añadir template al host registrado

Es posible añadir templates adicionales a un host ya registrado. En este caso, el template MySQL by Zabbix agent se vinculará solo a los hosts cuyo HostMetadata contenga el token `MySQL`.

Configuración del frontend

Actualice la acción en el frontend:

- Nombre: Autoregistration action Linux
- Condiciones:
  - Tipo de cálculo: AND
  - Condición (A): Host metadata contiene Linux
  - Condición (B): Host metadata contiene 21df83bf21bf0be663090bb8d4128558ab9b95fba66a6dbf834f8b91ae5e08ae
  - Condición (C): Host metadata contiene MySQL
- Operaciones:
  - Enviar mensaje a usuarios: Admin a través de todos los medios
  - Añadir a grupos de hosts: Linux servers
  - Vincular templates: Linux by Zabbix agent
  - Vincular templates: MySQL by Zabbix Agent

Configuración del agent

Actualice la siguiente línea en el archivo de configuración del agent:

```
HostMetadata=MySQL on Linux 21df83bf21bf0be663090bb8d4128558ab9b95fba66a6dbf834f8b91ae5e08ae
```

No olvide reiniciar el agent después de realizar cualquier cambio en el archivo de configuración.

Descubrimiento de bajo nivel

Descripción general La detección de bajo nivel (LLD) descubre automáticamente entidades (por ejemplo, sistemas de archivos, interfaces de red) en un host y crea los items, triggers y gráficos correspondientes sin necesidad de configuración manual para cada una.

Para usar LLD, se crea una **regla de descubrimiento** que recopila datos JSON que describen entidades, y **prototipos** (item, trigger y gráfico) para cada entidad.

El JSON devuelto por la regla debe ser un array de objetos, donde cada objeto representa una entidad descubierta mediante pares clave-valor. Por ejemplo, una regla de descubrimiento con `net.if.discovery` podría devolver:

```
[
  {"{#IFNAME}": "lo"},
  {"{#IFNAME}": "eth0"}
]
```

A continuación, se crea un conjunto correspondiente de items, triggers y gráficos a partir de los prototipos: uno por cada interfaz (`lo` y `eth0`).

Si los objetos contienen claves en el formato `{#MACRO}`, esas macros se usan directamente en los prototipos. Si no es así, o si se requieren macros adicionales/personalizadas, las macros pueden definirse manualmente y asignarse a campos JSON mediante JSONPath.

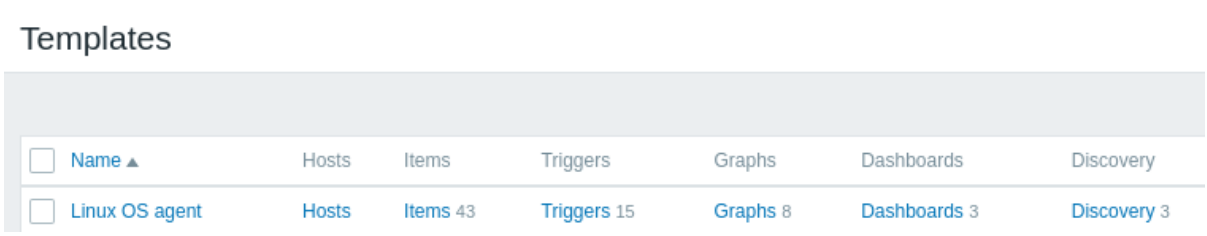
LLD también puede crear hosts (por ejemplo, para máquinas virtuales descubiertas en un hipervisor) y admite reglas de descubrimiento anidadas para el descubrimiento multinivel.

Las entidades que ya no se descubren pueden deshabilitarse o eliminarse automáticamente.

Configuración del descubrimiento de bajo nivel Ilustraremos el descubrimiento de bajo nivel con un ejemplo de descubrimiento del sistema de archivos.

Para configurar el descubrimiento, haga lo siguiente:

1. Vaya a *Data collection* > *Templates* o *Hosts*.
2. Haga clic en *Discovery* en la fila de la template/host correspondiente.



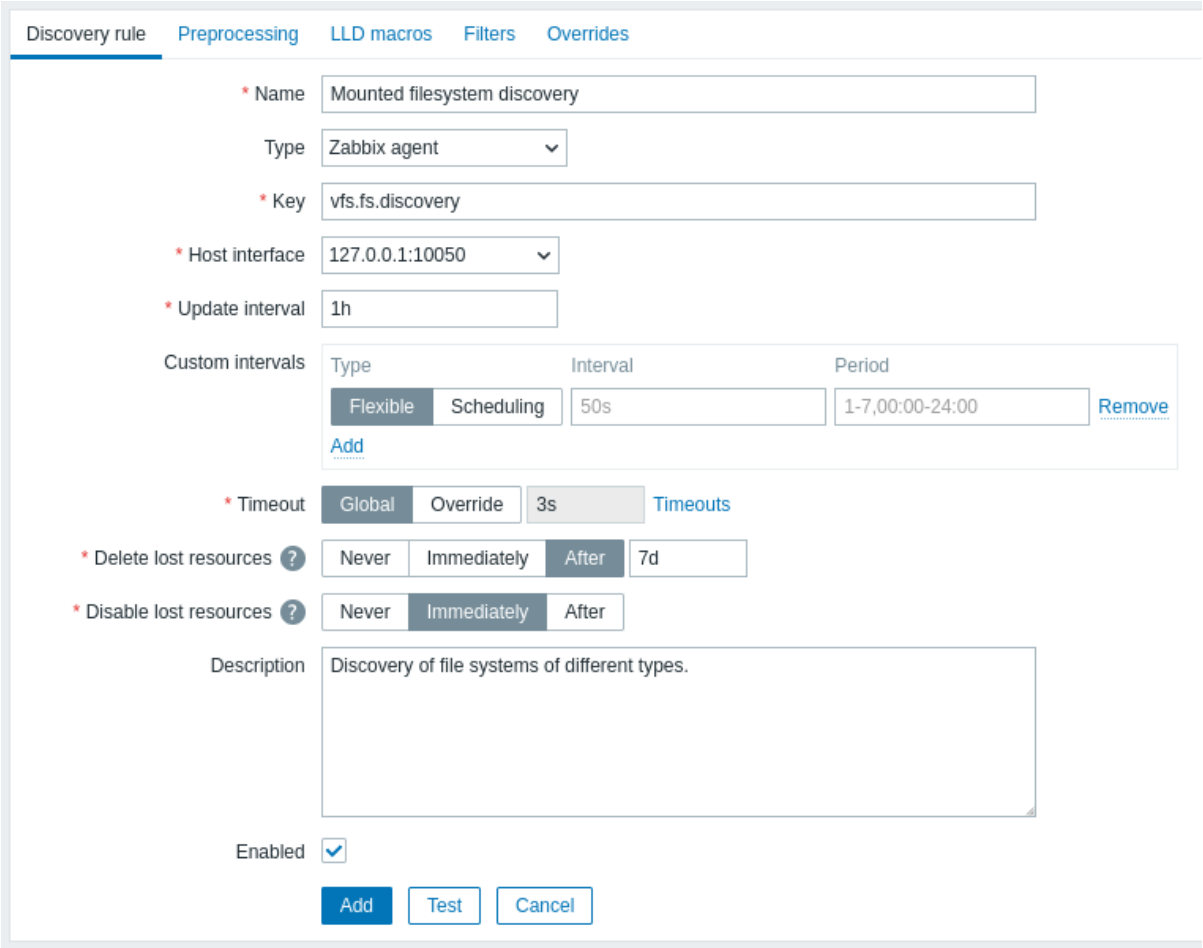
3. Haga clic en *Create discovery rule* en la esquina superior derecha de la pantalla.
4. Complete el formulario de la regla de descubrimiento con los detalles requeridos.

Regla de descubrimiento

El formulario de la regla de descubrimiento contiene cinco pestañas que representan, de izquierda a derecha, el flujo de datos durante el descubrimiento:

- *Regla de descubrimiento*: especifique, principalmente, el item integrado o el script personalizado que se utilizará para obtener los datos de descubrimiento.
- *Preprocesamiento*: aplique algún preprocesamiento a los datos descubiertos.
- *Macros LLD*: permiten extraer valores de macros para utilizarlos en items descubiertos, triggers, etc.
- *Filtros*: permiten filtrar los valores descubiertos.
- *Anulaciones*: permiten modificar items, triggers, gráficos o prototipos de host al aplicarlos a objetos descubiertos específicos.

La pestaña **Regla de descubrimiento** contiene la clave del item que se utilizará para el descubrimiento, así como algunos atributos generales de la regla de descubrimiento:



Todos los campos de entrada obligatorios están marcados con un asterisco rojo.

Parámetro	Descripción
Nombre	Nombre de la regla de descubrimiento.
Tipo	Tipo de comprobación que se realizará para el descubrimiento. En este ejemplo utilizamos un tipo de item *Zabbix agent*. La regla de descubrimiento también puede ser un item dependiente, que dependa de un item normal. No puede depender de otra regla de descubrimiento. Para un item dependiente, seleccione el tipo correspondiente (*Dependent item*) y especifique el item maestro en el campo «Master item». El item maestro debe existir.
Clave	Introduzca la clave del item de descubrimiento (hasta 2048 caracteres). Por ejemplo, puede utilizar la clave de item integrada «vfs.fs.discovery» para devolver una cadena JSON con la lista de sistemas de archivos presentes en el equipo, sus tipos y opciones de montaje. Tenga en cuenta que otra opción para descubrir sistemas de archivos consiste en utilizar los resultados del descubrimiento mediante la clave de agent «vfs.fs.get» (consulte el ejemplo).

Parámetro	Descripción
Intervalo de actualización	Este campo especifica con qué frecuencia Zabbix realiza el descubrimiento. Al principio, cuando esté configurando el descubrimiento de sistemas de archivos, puede establecer un intervalo pequeño, pero cuando compruebe que funciona, puede establecerlo en 30 minutos o más, ya que los sistemas de archivos normalmente no cambian con mucha frecuencia. Se admiten sufijos de tiempo, por ejemplo, 30s, 1m, 2h, 1d. Se admiten macros de usuario. *Nota*: el intervalo de actualización solo puede establecerse en «0» si existen intervalos personalizados con un valor distinto de cero. Si se establece en «0» y existe un intervalo personalizado (flexible o programado) con un valor distinto de cero, el item se consultará durante la duración del intervalo personalizado. Las nuevas reglas de descubrimiento se comprobarán en un plazo de 60 segundos desde su creación, a menos que tengan un intervalo de actualización programado o flexible y *Intervalo de actualización* esté establecido en 0. *Nota*: en una regla de descubrimiento existente, el descubrimiento puede realizarse inmediatamente pulsando el botón *Ejecutar ahora* button.
Intervalos personalizados	Puede crear reglas personalizadas para comprobar el item: **Flexible**: crea una excepción al *Intervalo de actualización* (un intervalo con una frecuencia diferente). **Programación**: crea una programación de consulta personalizada. Para obtener información detallada, consulte Intervalos personalizados.
Tiempo de espera	Establezca el tiempo de espera de la comprobación de descubrimiento. Seleccione la opción de tiempo de espera: **Global**: se utiliza el tiempo de espera del proxy/global (se muestra en el campo *Tiempo de espera* atenuado); **Anulación**: se utiliza un tiempo de espera personalizado (establecido en el campo *Tiempo de espera*; rango permitido: 1 - 600s). Se admiten sufijos de tiempo, por ejemplo, 30s y 1m, así como macros de usuario. Al hacer clic en el enlace *Tiempos de espera*, puede configurar los tiempos de espera del proxy o los tiempos de espera globales (si no se utiliza un proxy). Tenga en cuenta que el enlace *Tiempos de espera* solo está visible para los usuarios de tipo *Super admin* con permisos para las secciones del frontend *Administración* > *General* o *Administración* > *Proxies*.
Eliminar recursos perdidos	Especifique cuánto tiempo debe transcurrir antes de eliminar la entidad descubierta una vez que su estado de descubrimiento cambie a «Ya no se descubre»: *Nunca*: no se eliminará; *Inmediatamente*: se eliminará de inmediato; *Después de*: se eliminará una vez transcurrido el período especificado. El valor debe ser mayor que el valor de *Deshabilitar recursos perdidos*. Se admiten sufijos de tiempo, por ejemplo, 2h y 1d. Se admiten macros de usuario. *Nota*: no se recomienda utilizar «Inmediatamente», ya que una edición incorrecta del filtro podría provocar que la entidad se elimine junto con todos sus datos históricos. Tenga en cuenta que los recursos deshabilitados manualmente no serán eliminados por el descubrimiento de bajo nivel.
Deshabilitar recursos perdidos	Especifique cuánto tiempo debe transcurrir antes de deshabilitar la entidad descubierta una vez que su estado de descubrimiento cambie a «Ya no se descubre»: *Nunca*: no se deshabilitará; *Inmediatamente*: se deshabilitará de inmediato; *Después de*: se deshabilitará una vez transcurrido el período especificado. El valor debe ser mayor que el intervalo de actualización de la regla de descubrimiento. Tenga en cuenta que los recursos deshabilitados automáticamente volverán a habilitarse si el descubrimiento de bajo nivel los descubre de nuevo. Los recursos deshabilitados manualmente no volverán a habilitarse si se descubren de nuevo. Este campo no se muestra si *Eliminar recursos perdidos* está establecido en «Inmediatamente». Se admiten sufijos de tiempo, por ejemplo, 2h y 1d. Se admiten macros de usuario.
Descripción	Introduzca una descripción.
Habilitado	Si se selecciona, la regla se procesará.

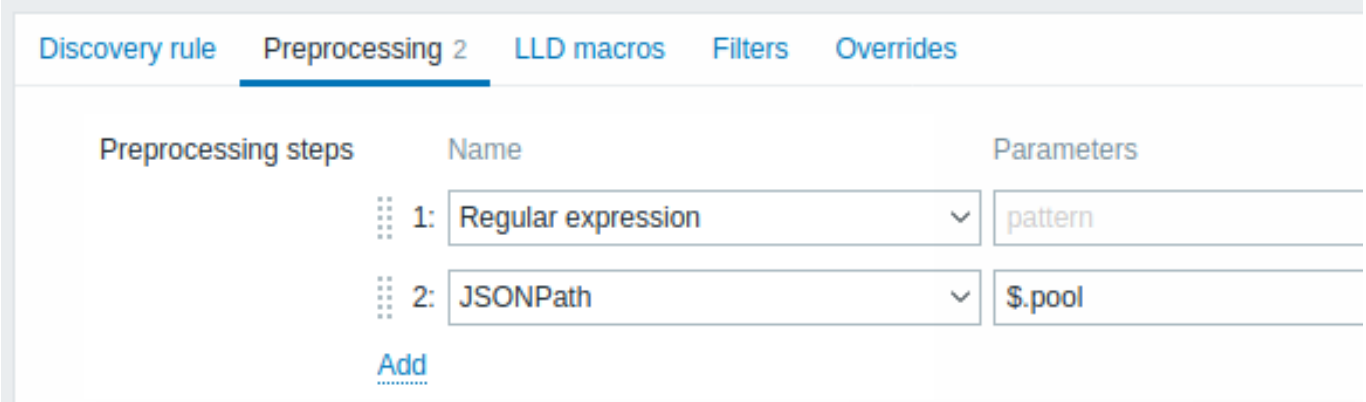
Note:
El historial de la regla de descubrimiento no se conserva.

Preprocesamiento

La pestaña **Preprocesamiento** permite definir reglas de transformación que se aplicarán al resultado del descubrimiento. En este paso se pueden realizar una o varias transformaciones. Las transformaciones se ejecutan en el orden en que se definen. Todo el preprocesamiento lo realiza Zabbix server.

Consulte también:

- Detalles del preprocesamiento
- Pruebas de preprocesamiento



Tipo		
	Transformación	Descripción
Texto		
	Expresión regular	Compara el valor recibido con la expresión regular <pattern> y reemplaza el valor por el <output> extraído. La expresión regular admite la extracción de un máximo de 10 grupos capturados mediante la secuencia \N. Parámetros: **pattern** - expresión regular **output** - template de formato de salida. Una secuencia de escape \N (donde N=1…9) se reemplaza por el grupo coincidente N. Una secuencia de escape \0 se reemplaza por el texto coincidente. Si marca la casilla *Personalizado en caso de error*, puede especificar opciones personalizadas para gestionar errores: descartar el valor, establecer un valor especificado o establecer un mensaje de error especificado.
	Reemplazar	Busca la cadena de búsqueda y la reemplaza por otra cadena (o por nada). Se reemplazarán todas las apariciones de la cadena de búsqueda. Parámetros: **search string** - cadena que se buscará y reemplazará; distingue entre mayúsculas y minúsculas (obligatorio) **replacement** - cadena por la que se reemplazará la cadena de búsqueda. La cadena de reemplazo también puede estar vacía, lo que permite eliminar la cadena de búsqueda cuando se encuentra. Es posible utilizar secuencias de escape para buscar o reemplazar saltos de línea, retornos de carro, tabulaciones y espacios "\n \r \t \s"; la barra inversa se puede escapar como "\\" y las secuencias de escape se pueden escapar como "\\n". El escape de saltos de línea, retornos de carro y tabulaciones se realiza automáticamente durante el descubrimiento de bajo nivel.
Datos estructurados		
	JSONPath	Extrae un valor o fragmento de datos JSON mediante la funcionalidad JSONPath. Si marca la casilla *Personalizado en caso de error*, puede especificar opciones personalizadas para gestionar errores: descartar el valor, establecer un valor especificado o establecer un mensaje de error especificado.

Tipo		

	XML XPath	Extrae un valor o fragmento de datos XML mediante la funcionalidad XPath.
		Para que esta opción funcione, Zabbix server debe compilarse con compatibilidad con libxml.
		Ejemplos:
		`number(/document/item/value)` extraerá 10 de `<document><item><value>10</value></item></document>`
		`number(/document/item/@attribute)` extraerá 10 de `<document><item attribute="10"></item></document>`
		`/document/item` extraerá `<item><value>10</value></item>` de `<document><item><value>10</value></item></document>`
		Tenga en cuenta que los espacios de nombres no son compatibles.
		Si marca la casilla *Personalizado en caso de error*, puede especificar opciones personalizadas para gestionar errores: descartar el valor, establecer un valor especificado o establecer un mensaje de error especificado.

	CSV a JSON	Convierte los datos de un archivo CSV al formato JSON. Para obtener más información, consulte: Preprocesamiento de CSV a JSON.
	XML a JSON	Convierte datos en formato XML a JSON. Para obtener más información, consulte: Reglas de serialización. Si marca la casilla *Personalizado en caso de error*, puede especificar opciones personalizadas para gestionar errores: descartar el valor, establecer un valor especificado o establecer un mensaje de error especificado.
SNMP		
	Valor de recorrido SNMP	Extrae el valor mediante el nombre OID/MIB especificado y aplica opciones de formato: **Sin cambios** - devuelve HEX-STRING como una cadena hexadecimal sin escapar (*nota*: las sugerencias de visualización se siguen aplicando); **UTF-8 desde HEX-STRING** - convierte HEX-STRING en una cadena UTF-8; **MAC desde HEX-STRING** - convierte HEX-STRING en una cadena de dirección MAC (en la que `' '` se reemplazará por `':'`); **Entero desde BITS** - convierte los primeros 8 bytes de una cadena de bits expresada como una secuencia de caracteres hexadecimales (por ejemplo, ”1A 2B 3C 4D”) en un entero sin signo de 64 bits. En cadenas de bits de más de 8 bytes, se ignorarán los bytes posteriores. Si marca la casilla *Personalizado en caso de error*, puede especificar opciones personalizadas para gestionar errores: descartar el valor, establecer un valor especificado o establecer un mensaje de error especificado.
	Recorrido SNMP a JSON	Convierte valores SNMP a JSON. Especifique un nombre de campo en JSON y la ruta OID SNMP correspondiente. Los valores de los campos se rellenarán con los valores de la ruta OID SNMP especificada. Puede utilizar este paso de preprocesamiento para el descubrimiento de OID SNMP. Están disponibles opciones de formato de valores similares a las del paso *Valor de recorrido SNMP*. Si marca la casilla *Personalizado en caso de error*, puede especificar opciones personalizadas para gestionar errores: descartar el valor, establecer un valor especificado o establecer un mensaje de error especificado.
	Obtener valor SNMP	Aplica opciones de formato al valor obtenido mediante SNMP: **UTF-8 desde HEX-STRING** - convierte HEX-STRING en una cadena UTF-8; **MAC desde HEX-STRING** - convierte HEX-STRING en una cadena de dirección MAC (en la que `' '` se reemplazará por `':'`); **Entero desde BITS** - convierte los primeros 8 bytes de una cadena de bits expresada como una secuencia de caracteres hexadecimales (por ejemplo, ”1A 2B 3C 4D”) en un entero sin signo de 64 bits. En cadenas de bits de más de 8 bytes, se ignorarán los bytes posteriores. Si marca la casilla *Personalizado en caso de error*, puede especificar opciones personalizadas para gestionar errores: descartar el valor, establecer un valor especificado o establecer un mensaje de error especificado.
Scripts personalizados		

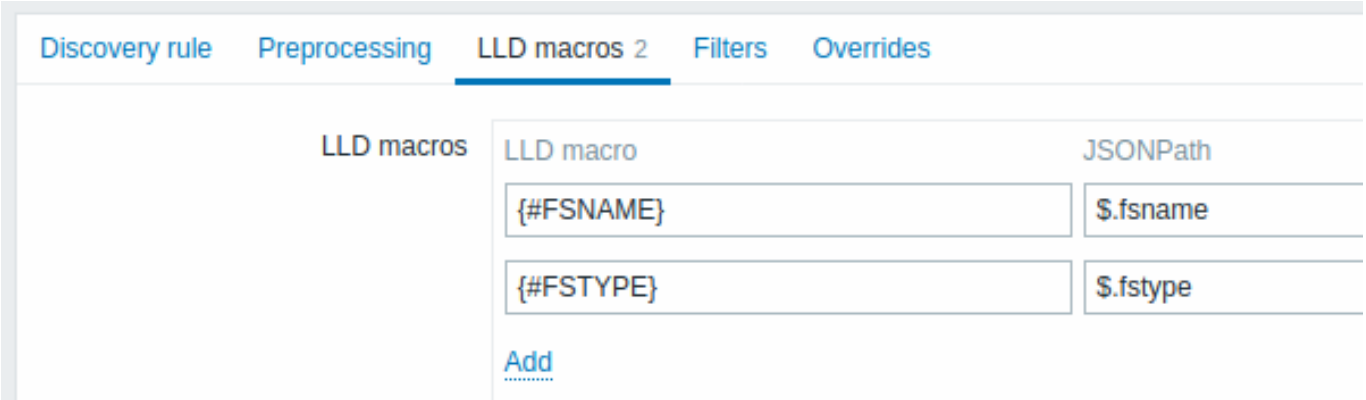
Tipo	
JavaScript	Introduzca código JavaScript en el editor modal que se abre al hacer clic en el campo de parámetros o en el icono del lápiz situado junto a él. Tenga en cuenta que la longitud de JavaScript disponible depende de la base de datos utilizada. Para obtener más información, consulte: Preprocesamiento de Javascript
Validación	
No coincide con la expresión regular	Especifique una expresión regular con la que el valor no debe coincidir. Por ejemplo, `Error:(.*?)\.` Si marca la casilla *Personalizado en caso de error*, puede especificar opciones personalizadas para gestionar errores: descartar el valor, establecer un valor especificado o establecer un mensaje de error especificado.
Comprobar si hay un error en JSON	Comprueba si hay un mensaje de error a nivel de aplicación ubicado en JSONPath. Detiene el procesamiento si la comprobación tiene éxito y el mensaje no está vacío; de lo contrario, continúa el procesamiento con el valor que existía antes de este paso de preprocesamiento. Tenga en cuenta que estos errores del servicio externo se notifican al usuario tal cual, sin añadir información sobre el paso de preprocesamiento. Por ejemplo, `$.errors`. Si se recibe un JSON como `{"errors":"e1"}`, no se ejecutará el siguiente paso de preprocesamiento. Si marca la casilla *Personalizado en caso de error*, puede especificar opciones personalizadas para gestionar errores: descartar el valor, establecer un valor especificado o establecer un mensaje de error especificado.
Comprobar si hay un error en XML	Comprueba si hay un mensaje de error a nivel de aplicación ubicado en XPath. Detiene el procesamiento si la comprobación tiene éxito y el mensaje no está vacío; de lo contrario, continúa el procesamiento con el valor que existía antes de este paso de preprocesamiento. Tenga en cuenta que estos errores del servicio externo se notifican al usuario tal cual, sin añadir información sobre el paso de preprocesamiento. No se notificará ningún error si no se puede analizar XML no válido. Si marca la casilla *Personalizado en caso de error*, puede especificar opciones personalizadas para gestionar errores: descartar el valor, establecer un valor especificado o establecer un mensaje de error especificado.
Coincide con la expresión regular	Especifique una expresión regular con la que el valor debe coincidir. Si marca la casilla *Personalizado en caso de error*, puede especificar opciones personalizadas para gestionar errores: descartar el valor, establecer un valor especificado o establecer un mensaje de error especificado.
Limitación	
Descartar los valores sin cambios con heartbeat	Descarta un valor si no ha cambiado durante el período definido (en segundos). Se admiten valores enteros positivos para especificar los segundos (mínimo: 1 segundo). En este campo se pueden utilizar sufijos de tiempo (por ejemplo, `30s`, `1m`, `2h`, `1d`). En este campo se pueden utilizar macros de usuario y macros de descubrimiento de bajo nivel. Solo se puede especificar una opción de limitación para un item de descubrimiento. Por ejemplo, `1m`. Si se pasa texto idéntico a esta regla dos veces en un intervalo de 60 segundos, se descartará. *Nota*: Cambiar los prototipos de item no restablece la limitación. La limitación solo se restablece cuando se modifican los pasos de preprocesamiento.
Prometheus	
Prometheus a JSON	Convierte las métricas de Prometheus requeridas a JSON. Consulte Comprobaciones de Prometheus para obtener más información.

Tenga en cuenta que, si la regla de descubrimiento se ha aplicado al host mediante un template, el contenido de esta pestaña es de solo lectura.

Macros personalizadas

La pestaña **LLD macros** permite especificar LLD macros personalizadas cuando el JSON devuelto no contiene aún claves en el formato `{#MACRO}`.

Por ejemplo, la clave de item del agent `vfs.fs.get` devuelve datos JSON de sistema de archivos sin ninguna LLD macro predefinida.
Aquí, define las macros manualmente y las asigna a valores en el JSON mediante expresiones de JSONPath:



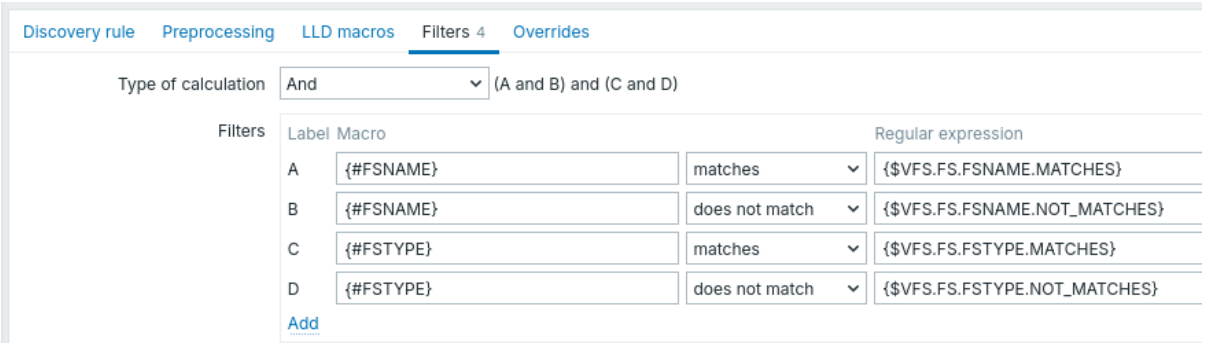
Parameter	Description
LLD macro *JSONPath*	Nombre de la LLD macro en el formato `{#MACRO}`. Expresión que define qué valor extraer de cada entidad descubierta en el array JSON. Por ejemplo, `$.foo` extrae `"bar"` y `"baz"` de `[{"foo":"bar"}, {"foo":"baz"}]`. JSONPath admite tanto la notación con punto como la notación con corchetes. La notación con corchetes debe usarse para claves que contengan caracteres especiales o Unicode, por ejemplo, `$['unicode + special chars #1']['unicode + special chars #2']`.

La regla de descubrimiento debe devolver un array JSON en la raíz (`$.`).
Por compatibilidad con versiones anteriores, también se acepta un objeto JSON que contenga un único array `"data"`. En este caso, el array `"data"` se desempaqueta automáticamente:

```
{"data":[{"foo":"bar"}, {"foo":"baz"}]}
```

Filtro

Se puede utilizar un filtro para generar items, triggers y gráficos reales únicamente para las entidades que coincidan con los criterios. La pestaña **Filtros** contiene definiciones de filtros de reglas de descubrimiento que permiten filtrar los valores descubiertos:



Parámetro	Descripción
Tipo de cálculo	Están disponibles las siguientes opciones para calcular los filtros: **And**: deben cumplirse todos los filtros; **Or**: es suficiente con que se cumpla un filtro; **And/Or**: utiliza *And* con nombres de macros diferentes y *Or* con el mismo nombre de macro; **Expresión personalizada**: permite definir un cálculo personalizado de los filtros. La fórmula debe incluir todos los filtros de la lista. Limitada a 255 símbolos.

Parámetro	Descripción
Filtros	Están disponibles los siguientes operadores de condición de filtro: *coincide*, *no coincide*, *existe*, *no existe*. Los operadores *coincide* y *no coincide* esperan una expresión regular compatible con Perl (PCRE). Puede introducir una expresión regular o hacer referencia a una expresión regular global en el campo *Expresión regular*. Los operadores *existe* y *no existe* permiten filtrar entidades según la presencia o ausencia de la macro de LLD especificada en la respuesta. Tenga en cuenta que, si falta en la respuesta una macro del filtro, la entidad encontrada se ignorará, a menos que se especifique una condición "no existe" para esta macro. Se mostrará una advertencia si la ausencia de una macro afecta al resultado de la expresión. Por ejemplo, si `{#B}` falta en: `{#A}` coincide con 1 y `{#B}` coincide con 2: se mostrará una advertencia `{#A}` coincide con 1 o `{#B}` coincide con 2: no se mostrará ninguna advertencia.

Por ejemplo, si solo le interesan los sistemas de archivos C:, D: y E:, puede introducir `{#FSNAME}` en el campo *Macro* y `^C|^D|^E` en el campo *Expresión regular*.
 También es posible filtrar por tipos de sistemas de archivos mediante la macro `{#FSTYPE}` (por ejemplo, `^ext|^reiserfs`) y por tipos de unidad (compatible únicamente con Windows agent) mediante la macro `{#FSDRIVETYPE}` (por ejemplo, `fixed`).

Para probar una expresión regular, puede utilizar `grep -E`, por ejemplo:

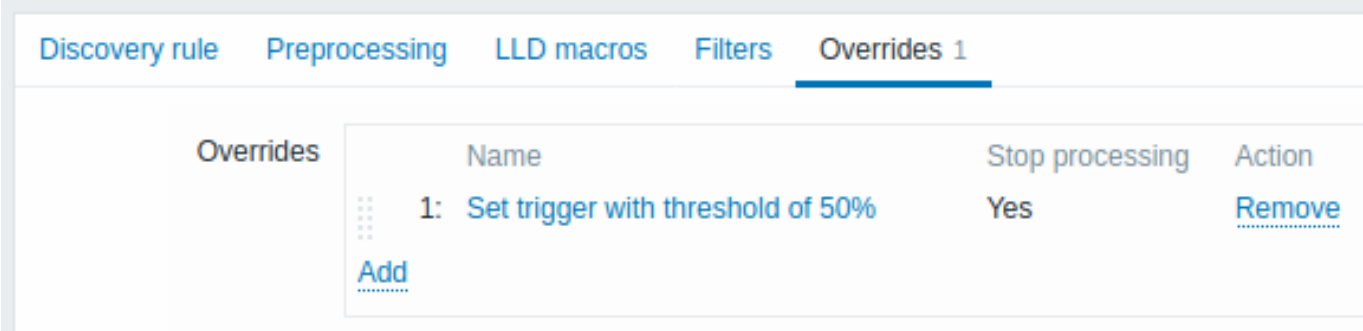
```
for f in ext2 nfs reiserfs smbfs;
do echo $f | grep -E '^ext|^reiserfs' || echo "SKIP: $f";
done
```

Warning:
Un error o una errata en la expresión regular utilizada en la regla de LLD (por ejemplo, una expresión regular incorrecta de "Sistemas de archivos para descubrir") puede provocar la eliminación de miles de elementos de configuración, valores históricos y eventos de muchos hosts.

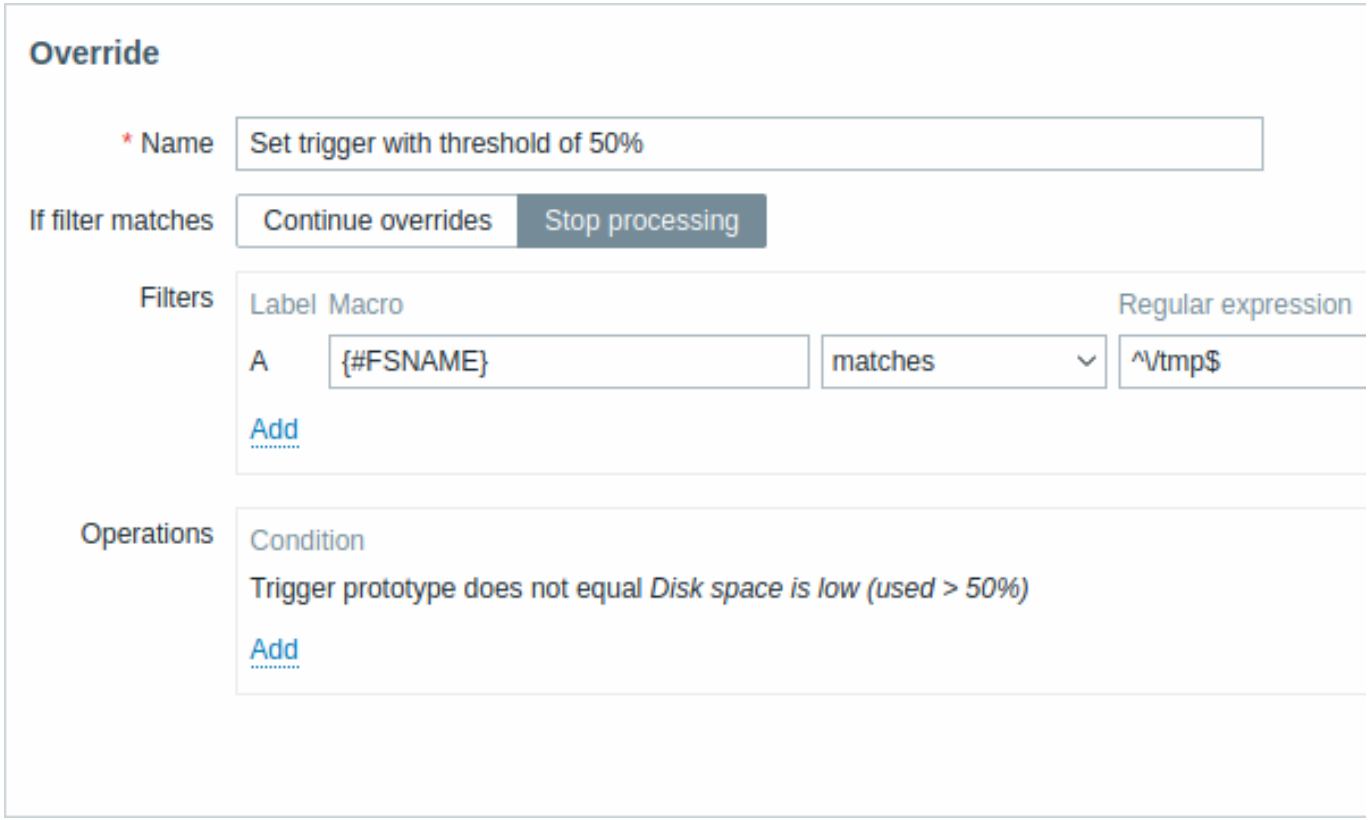
Attention:
La base de datos de Zabbix en MySQL debe crearse distinguiendo mayúsculas y minúsculas para que los nombres de sistemas de archivos que solo difieren en el uso de mayúsculas y minúsculas se descubran correctamente.

Anulación

La pestaña **Anulaciones** permite establecer reglas para modificar la lista de prototipos de item, trigger, gráfico, host y descubrimiento, o sus atributos, para los objetos descubiertos que cumplan determinados criterios.



Las anulaciones, si existen, se muestran en una lista reordenable mediante arrastrar y soltar y se ejecutan en el orden en que están definidas. Para configurar los detalles de una nueva anulación, haga clic en Add en el bloque *Anulaciones*. Para editar una anulación existente, haga clic en su nombre. Se abrirá una ventana emergente que permite editar los detalles de la regla de anulación.

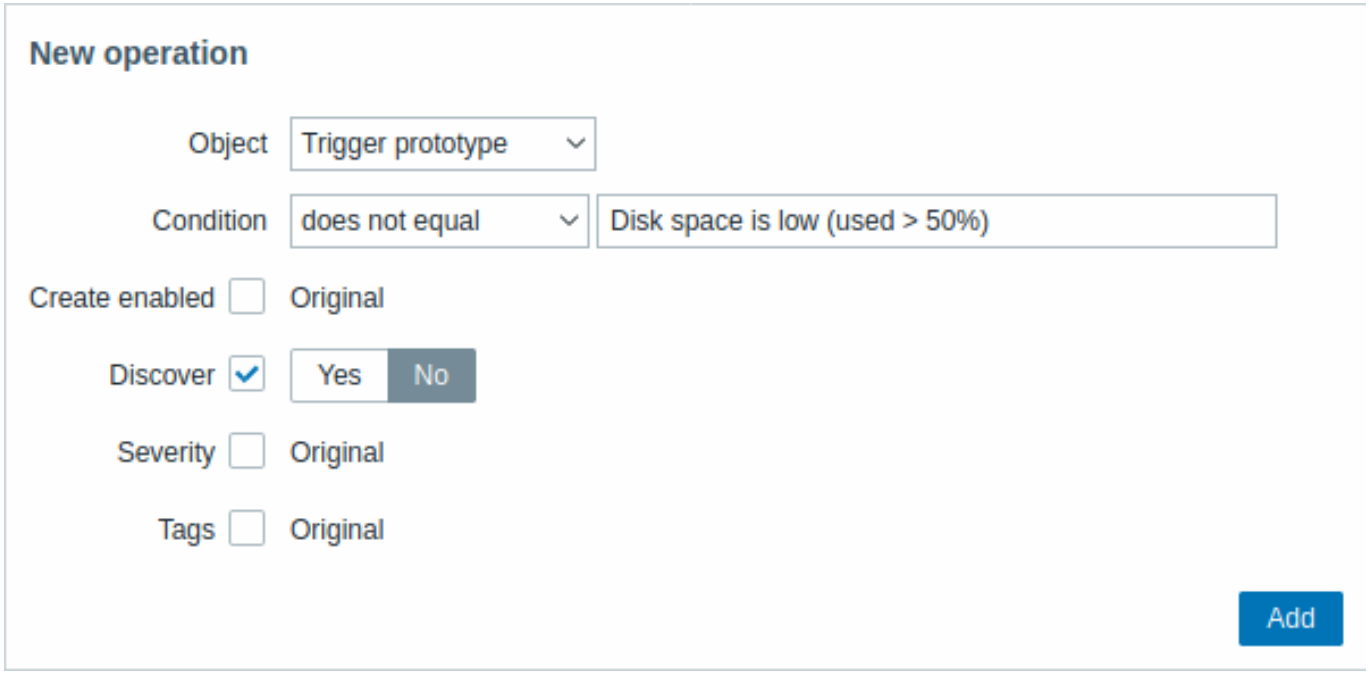


Todos los parámetros obligatorios están marcados con asteriscos rojos.

Parámetro	Descripción
Nombre	Un nombre de anulación único (por regla de LLD).
Si el filtro coincide	Define si deben procesarse las siguientes anulaciones cuando se cumplen las condiciones del filtro: **Continuar con las anulaciones**: se procesarán las anulaciones posteriores. **Detener el procesamiento**: se ejecutarán las operaciones de las anulaciones anteriores, si las hay, y de esta anulación; las anulaciones posteriores se ignorarán para las filas de LLD coincidentes.
Filtros	Determina a qué entidades descubiertas debe aplicarse la anulación. Los filtros de anulación se procesan después de los filtros de la regla de descubrimiento y tienen la misma funcionalidad.
Operaciones	Las operaciones de anulación se muestran con estos detalles: **Condición**: un tipo de objeto y una condición que debe cumplirse para el nombre del objeto; por ejemplo: el prototipo de trigger no es igual a *El espacio en disco es bajo (usado > 50%)*. **Acciones**: se muestran enlaces para editar y eliminar una operación.

Configuración de una operación

Para configurar los detalles de una nueva operación, haga clic en Add en el bloque Operaciones. Para editar una operación existente, haga clic en Edit junto a la operación. Se abrirá una ventana emergente en la que podrá editar los detalles de la operación.



Parámetro	Descripción
Objeto	Hay cinco tipos de objetos disponibles: Prototipo de item Prototipo de trigger Prototipo de gráfico Prototipo de host Prototipo de descubrimiento
Condición	Permite filtrar las entidades a las que debe aplicarse la operación.
Operador	Operadores compatibles: **es igual a**: aplicar a este prototipo **no es igual a**: aplicar a todos los prototipos excepto este **contiene**: aplicar si el nombre del prototipo contiene esta cadena **no contiene**: aplicar si el nombre del prototipo no contiene esta cadena **coincide**: aplicar si el nombre del prototipo coincide con la expresión regular **no coincide**: aplicar si el nombre del prototipo no coincide con la expresión regular
Patrón	Una expresión regular o una cadena que buscar.
Objeto: *Prototipo de item*	
Crear habilitado	Al marcar la casilla, aparecerán botones que permiten anular la configuración original del prototipo de item: *Sí*: el item se añadirá en estado habilitado. *No*: el item se añadirá a una entidad descubierta, pero en estado deshabilitado.
Descubrir	Al marcar la casilla, aparecerán botones que permiten anular la configuración original del prototipo de item: *Sí*: el item se añadirá. *No*: el item no se añadirá.
Intervalo de actualización	Al marcar la casilla, aparecerán dos opciones que permiten establecer un intervalo diferente para el item: *Retraso*: intervalo de actualización del item. Se admiten macros de usuario, macros de LLD y sufijos de tiempo (por ejemplo, 30s, 1m, 2h, 1d). Debe establecerse en 0 si se utiliza *Intervalo personalizado*. *Intervalo personalizado*: haga clic en Add para especificar intervalos flexibles o programados. Para obtener información detallada, consulte Intervalos personalizados.

Parámetro		Descripción
	Historial	Al marcar la casilla, aparecerán botones que permiten establecer diferentes períodos de almacenamiento del historial para el item: *No almacenar*: si se selecciona, el historial no se almacenará. *Almacenar hasta*: si se selecciona, aparecerá a la derecha un campo de entrada para especificar el período de almacenamiento. Se admiten macros de usuario y macros de LLD.
	Tendencias	Al marcar la casilla, aparecerán botones que permiten establecer diferentes períodos de almacenamiento de tendencias para el item: *No almacenar*: si se selecciona, las tendencias no se almacenarán. *Almacenar hasta*: si se selecciona, aparecerá a la derecha un campo de entrada para especificar el período de almacenamiento. Se admiten macros de usuario y macros de LLD.
	Etiquetas	Al marcar la casilla, aparecerá un nuevo bloque que permite especificar pares etiqueta-valor. Se admiten macros de usuario y macros de LLD. Estas etiquetas se añadirán a las etiquetas especificadas en el prototipo de item, incluso si los nombres de las etiquetas coinciden.
Objeto: *Prototipo de trigger*		
	Crear habilitado	Al marcar la casilla, aparecerán botones que permiten anular la configuración original del prototipo de trigger: *Sí*: el trigger se añadirá en estado habilitado. *No*: el trigger se añadirá a una entidad descubierta, pero en estado deshabilitado.
	Descubrir	Al marcar la casilla, aparecerán botones que permiten anular la configuración original del prototipo de trigger: *Sí*: el trigger se añadirá. *No*: el trigger no se añadirá.
	Severidad	Al marcar la casilla, aparecerán botones de severidad del trigger que permiten modificar su severidad.
	Etiquetas	Al marcar la casilla, aparecerá un nuevo bloque que permite especificar pares etiqueta-valor. Se admiten macros de usuario y macros de LLD. Estas etiquetas se añadirán a las etiquetas especificadas en el prototipo de trigger, incluso si los nombres de las etiquetas coinciden.
Objeto: *Prototipo de gráfico*		
	Descubrir	Al marcar la casilla, aparecerán botones que permiten anular la configuración original del prototipo de gráfico: *Sí*: el gráfico se añadirá. *No*: el gráfico no se añadirá.
Objeto: *Prototipo de host*		
	Crear habilitado	Al marcar la casilla, aparecerán botones que permiten anular la configuración original del prototipo de host: *Sí*: el host se creará en estado habilitado. *No*: el host se creará en estado deshabilitado.
	Descubrir	Al marcar la casilla, aparecerán botones que permiten anular la configuración original del prototipo de host: *Sí*: el host se descubrirá. *No*: el host no se descubrirá.

Parámetro		Descripción
	Vincular templates	Al marcar la casilla, aparecerá un campo de entrada para especificar templates. Comience a escribir el nombre del template o haga clic en *Seleccionar* junto al campo y seleccione templates de la lista en una ventana emergente. Los templates de esta anulación se añaden a todos los templates ya vinculados al prototipo de host.
	Etiquetas	Al marcar la casilla, aparecerá un nuevo bloque que permite especificar pares etiqueta-valor. Se admiten macros de usuario y macros de LLD. Estas etiquetas se añadirán a las etiquetas especificadas en el prototipo de host, incluso si los nombres de las etiquetas coinciden.
	Inventario del host	Al marcar la casilla, aparecerán botones que permiten seleccionar un modo de inventario diferente para el prototipo de host: *Deshabilitado*: no completar el inventario del host *Manual*: proporcionar los detalles manualmente *Automatizado*: completar automáticamente los datos del inventario del host según las métricas recopiladas.
Objeto: *Prototipo de descubrimiento*		
	Crear habilitado	Al marcar la casilla, aparecerán botones que permiten anular la configuración original del prototipo de descubrimiento: *Sí*: la regla de descubrimiento se añadirá en estado habilitado. *No*: la regla de descubrimiento se añadirá en estado deshabilitado.
	Descubrir	Al marcar la casilla, aparecerán botones que permiten anular la configuración original del prototipo de descubrimiento: *Sí*: la regla de descubrimiento se añadirá. *No*: la regla de descubrimiento no se añadirá.
	Intervalo de actualización	Al marcar la casilla, aparecerán dos opciones que permiten establecer un intervalo diferente para la regla de descubrimiento: *Retraso*: intervalo de actualización de la regla. Se admiten macros de usuario, macros de LLD y sufijos de tiempo (por ejemplo, 30s, 1m, 2h, 1d). Debe establecerse en 0 si se utiliza *Intervalo personalizado*. *Intervalo personalizado*: haga clic en Add para especificar intervalos flexibles o programados. Para obtener información detallada, consulte Intervalos personalizados.

Botones del formulario

Los botones situados en la parte inferior del formulario permiten realizar varias operaciones.

Add	Añadir una regla de descubrimiento. Este botón solo está disponible para reglas de descubrimiento nuevas.
Update	Actualizar las propiedades de una regla de descubrimiento. Este botón solo está disponible para reglas de descubrimiento existentes.
Clone	Crear otra regla de descubrimiento basada en las propiedades de la regla de descubrimiento actual.
Execute now	Realizar inmediatamente el descubrimiento según la regla de descubrimiento. La regla de descubrimiento ya debe existir. Consulte más detalles. *Nota*: al realizar el descubrimiento inmediatamente, la caché de configuración no se actualiza, por lo que el resultado no reflejará los cambios más recientes en la configuración de la regla de descubrimiento.
Test	Probar la configuración de la regla de descubrimiento. Utilice este botón para verificar los ajustes de configuración (como la conectividad y la corrección de los parámetros) sin aplicar permanentemente ningún cambio.
Delete	Eliminar la regla de descubrimiento.
Cancel	Cancelar la edición de las propiedades de la regla de descubrimiento.

Entidades descubiertas Las siguientes capturas de pantalla ilustran cómo se ven los items, triggers y gráficos descubiertos en la configuración del host. Las entidades descubiertas están precedidas por un enlace naranja a la regla de descubrimiento de la que provienen.

Tenga en cuenta que las entidades descubiertas no se crearán en caso de que ya existan entidades con los mismos criterios de unicidad, por ejemplo, un item con la misma clave o un gráfico con el mismo nombre. En este caso, se muestra un mensaje de error en el frontend indicando que la regla de descubrimiento de bajo nivel no pudo crear ciertas entidades. Sin embargo, la propia regla de descubrimiento no se volverá no soportada porque alguna entidad no pudo ser creada y tuvo que ser omitida. La regla de descubrimiento continuará creando/actualizando otras entidades.

Si una entidad descubierta (host, sistema de archivos, interfaz, etc.) deja de ser descubierta (o ya no pasa el filtro), las entidades que se crearon en base a ella pueden ser deshabilitadas automáticamente y eventualmente eliminadas.

Los recursos perdidos pueden ser deshabilitados automáticamente según el valor del parámetro *Deshabilitar recursos perdidos*. Esto afecta a los hosts, items y triggers perdidos.

Los recursos perdidos pueden ser eliminados automáticamente según el valor del parámetro *Eliminar recursos perdidos*. Esto afecta a los hosts, grupos de hosts, items, triggers y gráficos perdidos.

Cuando las entidades descubiertas pasan a estar 'Ya no descubiertas', se muestra un indicador de tiempo de vida en la lista de entidades. Mueva el puntero del ratón sobre él y se mostrará un mensaje que indica los detalles de su estado.

Si las entidades fueron marcadas para su eliminación, pero no se eliminaron en el momento esperado (regla de descubrimiento o host del item deshabilitado), se eliminarán la próxima vez que se procese la regla de descubrimiento.

Las entidades que contienen otras entidades, que están marcadas para su eliminación, no se actualizarán si se cambian a nivel de la regla de descubrimiento. Por ejemplo, los triggers basados en LLD no se actualizarán si contienen items que están marcados para su eliminación.

Otros tipos de descubrimiento Más detalles y procedimientos sobre otros tipos de descubrimiento listos para usar están disponibles en las siguientes secciones:

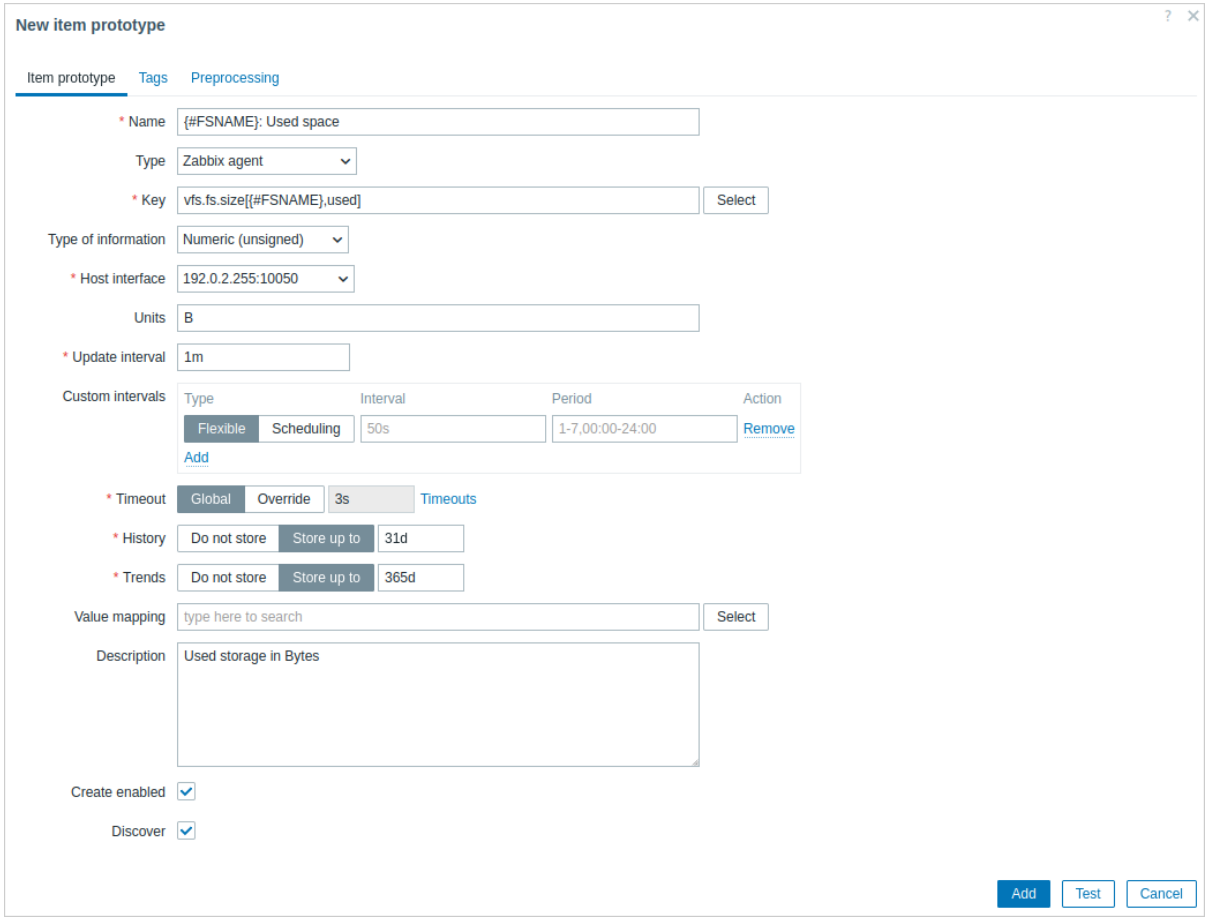
- descubrimiento de interfaces de red
- descubrimiento de CPUs y núcleos de CPU
- descubrimiento de OIDs SNMP
- descubrimiento de objetos JMX;
- descubrimiento usando consultas SQL ODBC
- descubrimiento de servicios de Windows
- descubrimiento de interfaces de host en Zabbix

Para más detalles sobre el formato JSON para items de descubrimiento y un ejemplo de cómo implementar su propio descubridor de sistemas de archivos como un script Perl, consulte creación de reglas LLD personalizadas.

Prototipos de item

Una vez creada una regla, vaya a los items de esa regla y haga clic en *Crear prototipo de item* para crear un prototipo de item.

Observe cómo se utiliza la macro `{#FSNAME}` cuando se requiere un nombre de sistema de archivos. El uso de una macro de descubrimiento de bajo nivel es obligatorio en la clave del item para garantizar que el descubrimiento se procese correctamente. Cuando se procesa la regla de descubrimiento, esta macro se sustituirá por el sistema de archivos descubierto.



Note:
Cuando no se utilizan parámetros de la clave del item, coloque la macro de LLD dentro de los corchetes de la clave del item `[...]` (por ejemplo, `v_[{#MACRO}]`).

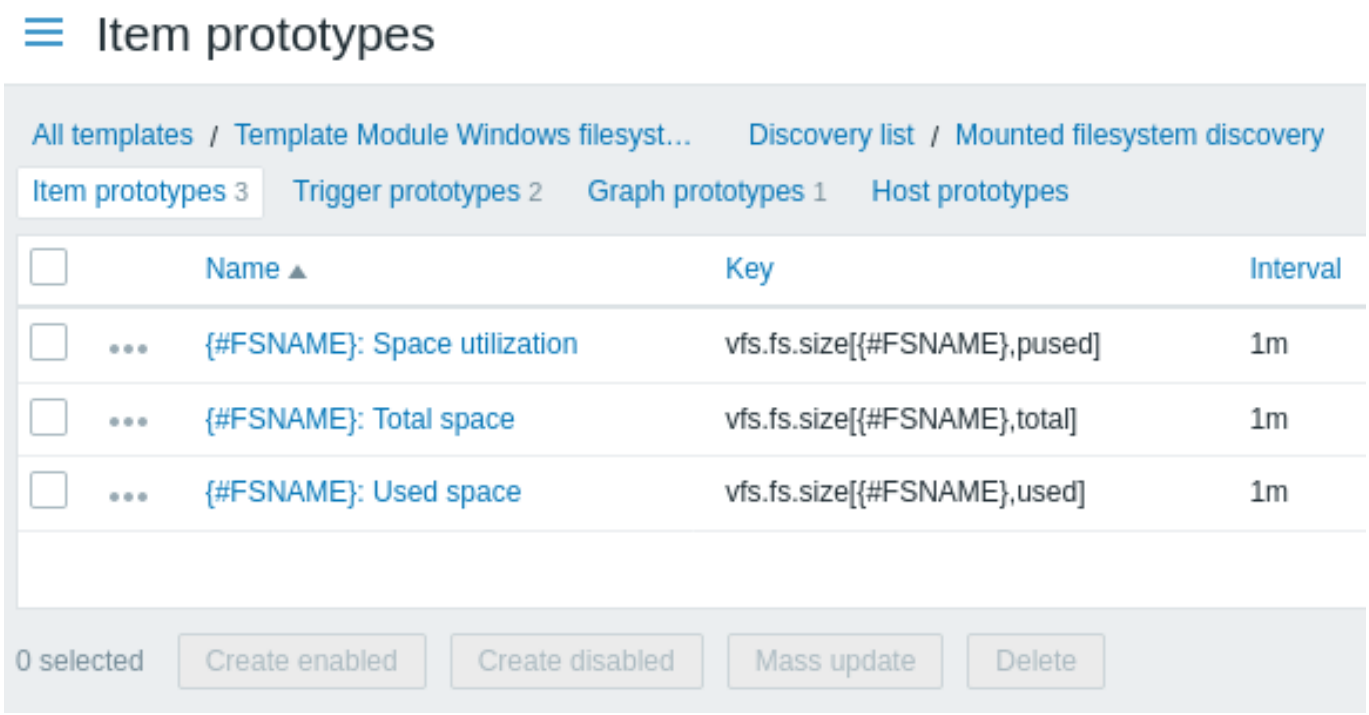
Las macros de descubrimiento de bajo nivel y las macros de usuario son compatibles con la configuración de prototipos de item y con los parámetros de preprocesamiento de valores de item. Tenga en cuenta que, cuando se utilizan en intervalos de actualización, una sola macro debe ocupar todo el campo. No se admiten varias macros en un mismo campo ni macros combinadas con texto.

Note:
El escape contextual de las macros de descubrimiento de bajo nivel se realiza para permitir su uso seguro en parámetros de preprocesamiento de expresiones regulares y XPath.

Atributos específicos de los prototipos de item:

Parámetro	Descripción
Crear habilitado	Si se selecciona, el item se añadirá en estado habilitado. Si no se selecciona, el item se añadirá a una entidad descubierta, pero en estado deshabilitado.
Descubrir	Si se selecciona (opción predeterminada), el item se añadirá a una entidad descubierta. Si no se selecciona, el item no se añadirá a una entidad descubierta, a menos que esta configuración se sobrescriba en la regla de descubrimiento.

Podemos crear varios prototipos de item para cada métrica del sistema de archivos que nos interese:



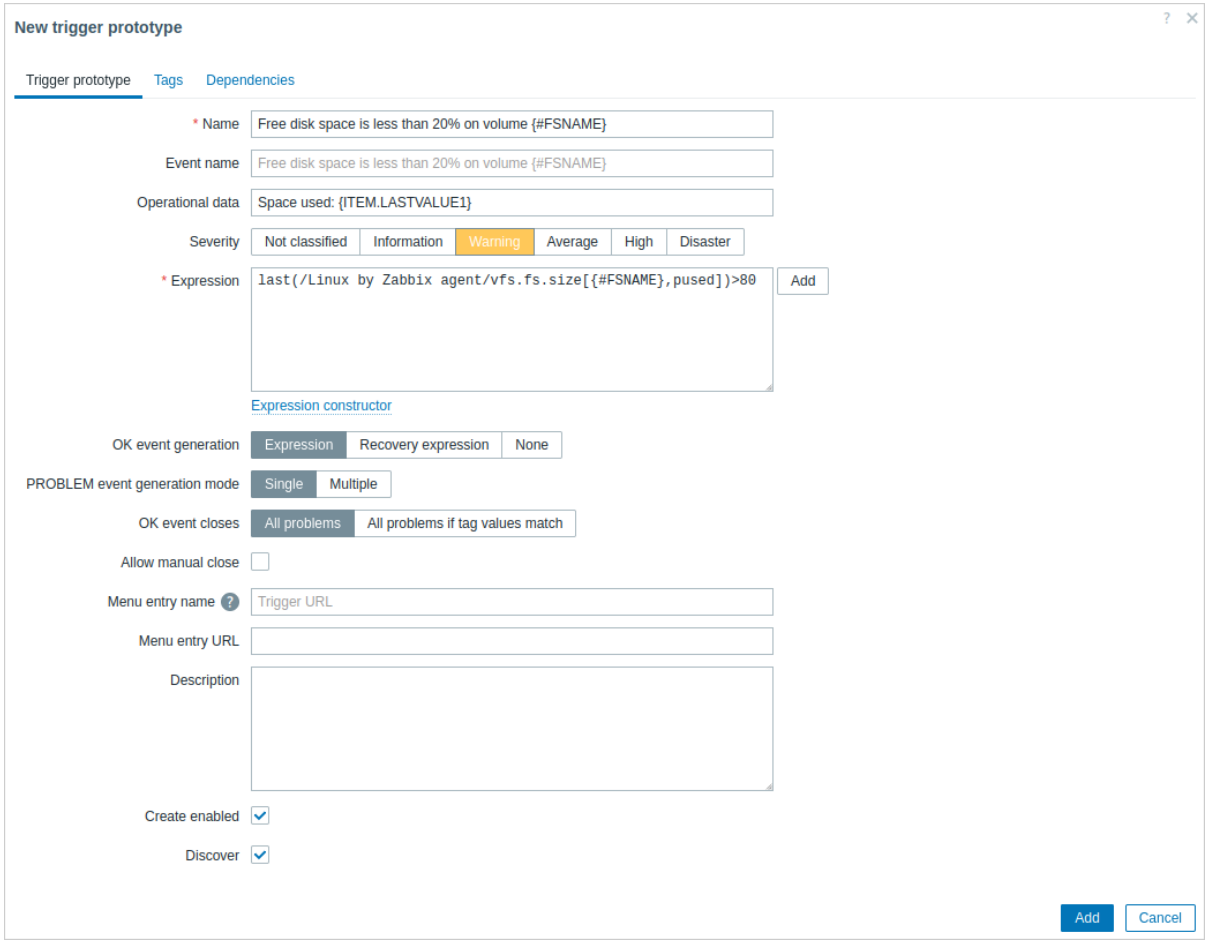
Haga clic en el icono de tres puntos para abrir el menú del prototipo de item específico con estas opciones:

- *Crear prototipo de trigger*: crear un prototipo de trigger basado en este prototipo de item
- *Prototipos de trigger*: hacer clic para ver una lista con enlaces a los prototipos de trigger ya configurados para este prototipo de item
- *Crear item dependiente*: crear un item dependiente para este prototipo de item

La opción *Actualización masiva* está disponible si desea actualizar las propiedades de varios prototipos de item a la vez.

Prototipos de trigger

Creamos prototipos de trigger de una manera similar a los prototipos de item:

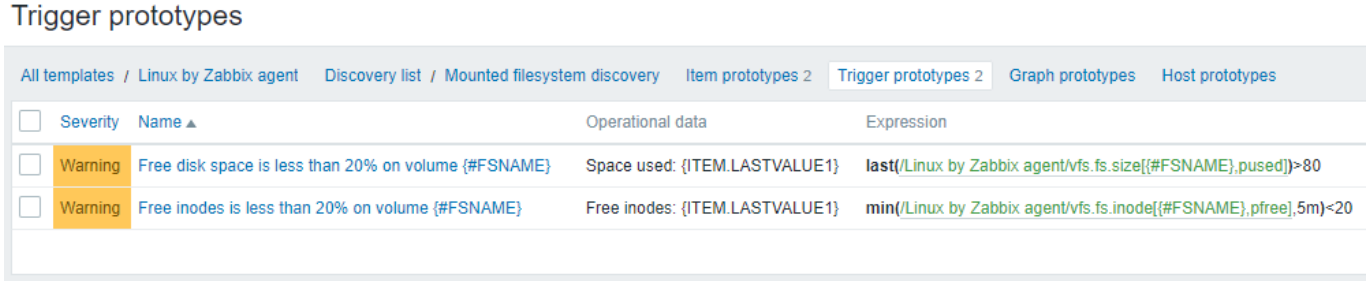


Atributos específicos de los prototipos de trigger:

Parameter	Description
Create enabled	Si está marcado, el trigger se añadirá en estado habilitado. Si no está marcado, el trigger se añadirá a una entidad descubierta, pero en estado deshabilitado.
Discover	Si está marcado (valor predeterminado), el trigger se añadirá a una entidad descubierta. Si no está marcado, el trigger no se añadirá a una entidad descubierta, a menos que esta configuración se sobrescriba en la regla de descubrimiento.

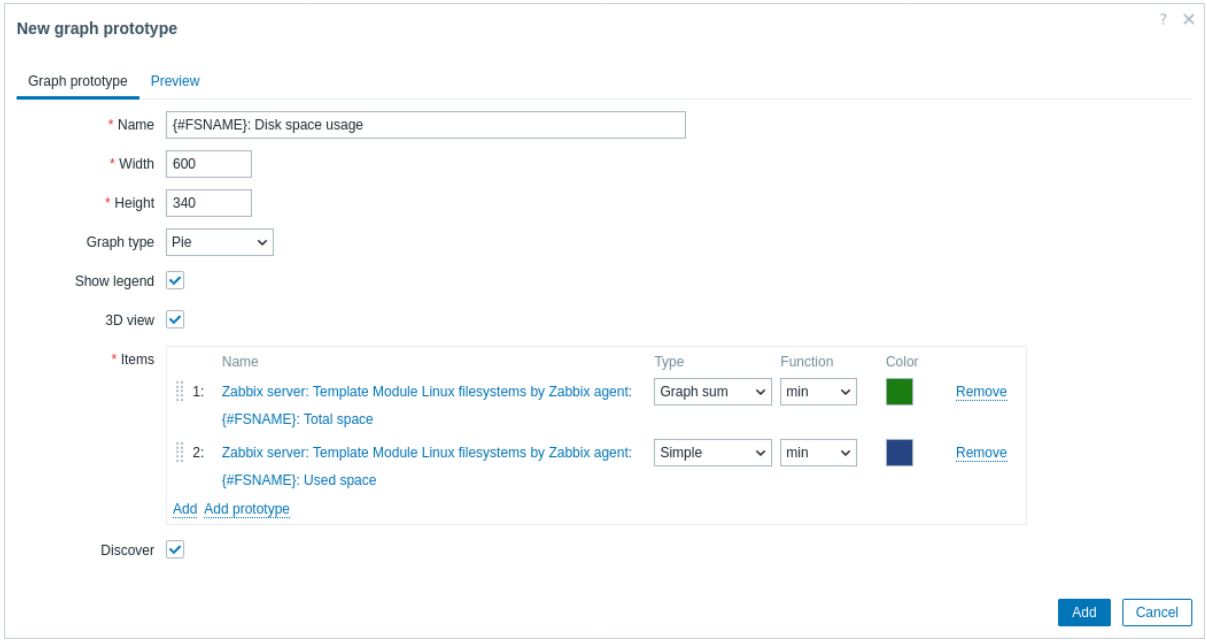
Cuando se crean triggers reales a partir de los prototipos, puede ser necesario tener flexibilidad respecto a qué constante ('20' en nuestro ejemplo) se usa para la comparación en la expresión. Vea cómo los macros de usuario con contexto pueden ser útiles para lograr esa flexibilidad.

Puede definir dependencias entre prototipos de trigger. Para ello, vaya a la pestaña *Dependencies*. Un prototipo de trigger puede depender de otro prototipo de trigger de la misma regla de descubrimiento de bajo nivel (LLD) o de un trigger normal. Un prototipo de trigger no puede depender de un trigger de otro regla LLD ni de un trigger creado a partir de un prototipo de trigger. Un prototipo de trigger de host no puede depender de un trigger de un template.



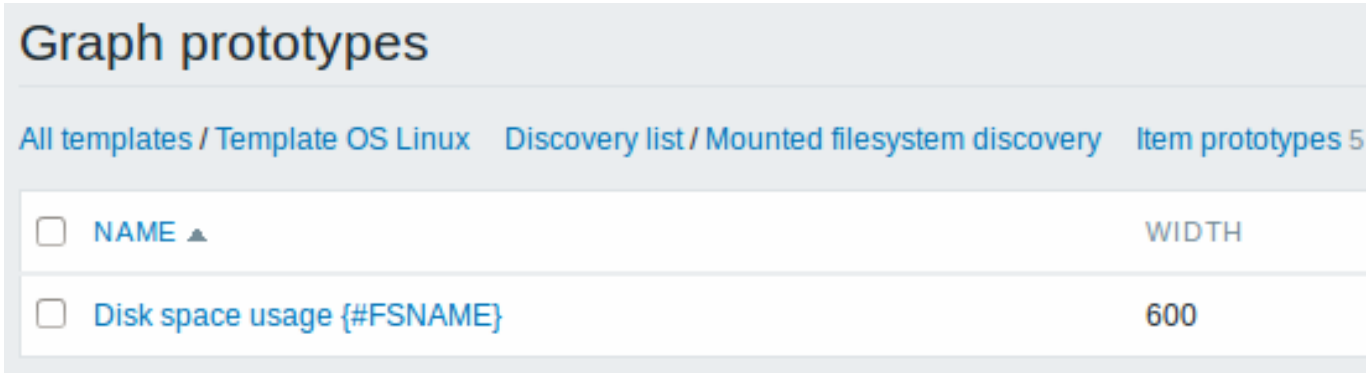
Prototipos de gráfico

También podemos crear prototipos de gráfico:

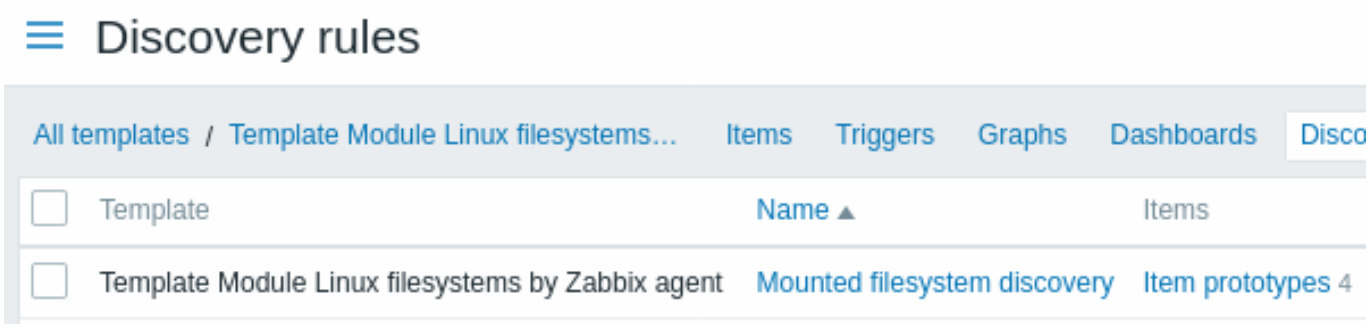


Atributos específicos de los prototipos de gráfico:

Parameter	Description
Discover	Si está marcado (predeterminado), el gráfico se agregará a una entidad descubierta. Si no está marcado, el gráfico no se agregará a una entidad descubierta, a menos que esta configuración se sobrescriba en la regla de descubrimiento.



Por último, hemos creado una regla de descubrimiento que se muestra a continuación. Tiene cinco prototipos de item, dos prototipos de trigger y un prototipo de gráfico.



Prototipos de host

Los prototipos de equipos son modelos para crear equipos mediante reglas de descubrimiento de bajo nivel. Antes de ser descubiertos como equipos, estos prototipos no pueden tener métricas ni disparadores, excepto aquellos vinculados desde plantillas.

Configuración

Los host prototypes se configuran en reglas de descubrimiento de bajo nivel.

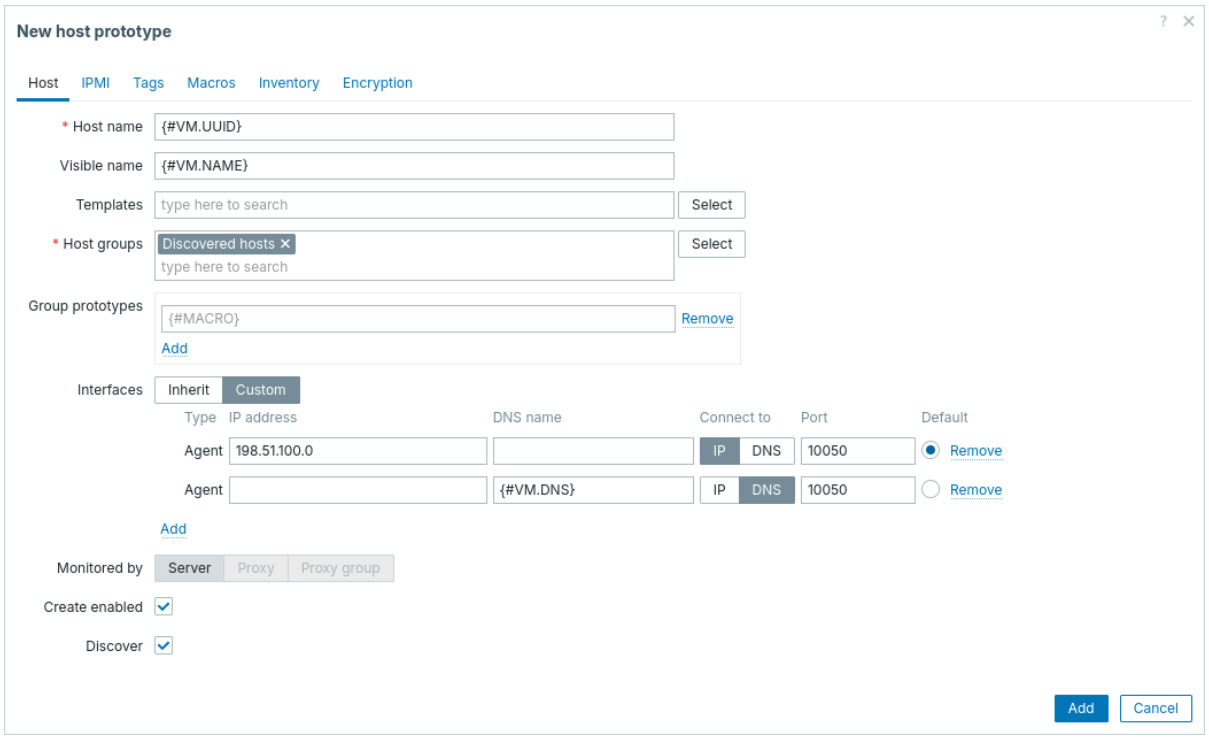
Para crear un host prototype:

1. Vaya a *Recopilación de datos > Hosts*.

2. Haga clic en *Discovery* para el host requerido y vaya a la lista de reglas de descubrimiento de bajo nivel configuradas para ese host.

3. Haga clic en *Host prototypes* para la regla de descubrimiento requerida.

4. Haga clic en el botón *Create host prototype* en la esquina superior derecha y complete el formulario de configuración.



Todos los campos obligatorios están marcados con un asterisco rojo.

Parameter	Description
Host name	Este parámetro debe contener al menos una macro de descubrimiento de bajo nivel para garantizar nombres de host únicos para los hosts creados.
Visible name	Se admiten macros de descubrimiento de bajo nivel.
Group prototypes	Permite especificar prototipos de grupos de host mediante macros de descubrimiento de bajo nivel. Según los prototipos de grupo especificados, se descubrirán, crearán y vincularán a los hosts creados grupos de host; los grupos descubiertos que ya hayan sido creados por otras reglas de descubrimiento de bajo nivel también se vincularán a los hosts creados. Sin embargo, los grupos de host descubiertos que coincidan con grupos de host creados manualmente no se vincularán a los hosts creados.
Interfaces	Defina si los hosts descubiertos heredan la IP del host al que pertenece la regla de descubrimiento (valor predeterminado) o si obtienen interfaces personalizadas. Se admiten macros de descubrimiento de bajo nivel y macros de usuario.
Monitored by	Este campo se sincroniza automáticamente con el campo correspondiente en la configuración del host.
Create enabled	Defina el estado de los hosts descubiertos; si no se marca, los hosts se crearán deshabilitados.
Discover	Defina si los hosts se crearán a partir del host prototype; si no se marca, los hosts no se crearán a partir del host prototype (a menos que esta configuración se sobrescriba en la regla de descubrimiento de bajo nivel).

Note:
También se admiten macros de descubrimiento de bajo nivel para los valores de etiquetas y los valores de macros de usuario del host prototype.
 Los *Value maps* no son compatibles con los host prototypes.

Para ver un ejemplo de cómo configurar un host prototype, consulte *Virtual machine monitoring*.

Interfaces de host

Para añadir interfaces personalizadas, cambie el selector *Interfaces* de "Heredar" a "Personalizado". Haga clic en Add y seleccione el tipo de interfaz: agent Zabbix, SNMP, JMX, IPMI.

Note:
Si se selecciona *Personalizado*, pero no se han configurado interfaces, los hosts se crearán sin interfaces.
 Si se selecciona *Heredar* y la plantilla de host pertenece a una template, todos los hosts descubiertos heredarán la interfaz de host del host al que está vinculada la template.

Si se especifican varias interfaces personalizadas, la interfaz principal puede establecerse en la columna *Por defecto*.

Para ver un ejemplo de cómo configurar una interfaz de host personalizada, consulte *Ejemplo de configuración de monitorización de VMware*.

Warning:
Un host sólo se creará si una interfaz de host contiene datos correctos.

Equipos descubiertos

En la lista de equipos, los equipos descubiertos están precedidos por el nombre de la regla de descubrimiento que los creó.

Los equipos descubiertos heredan la mayoría de los parámetros de los prototipos de equipo como *solo lectura*. Solo se pueden configurar los siguientes parámetros de los equipos descubiertos:

- *Plantillas* - vincular nuevas plantillas o desvincular las añadidas manualmente. Las plantillas heredadas de un prototipo de equipo no se pueden desvincular.
- *Descripción* - añadir o editar la descripción del equipo.
- *Estado* - habilitar o deshabilitar el equipo.
- *Etiquetas* - añadir nuevas etiquetas o eliminar las añadidas manualmente. Las etiquetas heredadas de un prototipo de equipo no se pueden eliminar. Tenga en cuenta que las etiquetas no pueden tener duplicados (etiquetas con el mismo nombre y valor). Si se añade una nueva etiqueta al prototipo de equipo y coincide con una etiqueta añadida manualmente en un equipo descubierto, la etiqueta añadida manualmente será reemplazada por la etiqueta heredada durante el descubrimiento.
- *Macros* - añadir nuevas macros de equipo o eliminar las añadidas manualmente, así como las heredadas de un prototipo de equipo; cambiar los valores y tipos de las macros.

Note:
Los equipos descubiertos también heredan macros de usuario del equipo donde está configurada la regla de descubrimiento. Estas macros heredadas pueden eliminarse, y sus valores y tipos pueden cambiarse.

Los equipos descubiertos también pueden descubrir otros equipos. Por ejemplo, si un prototipo de equipo (utilizado para descubrir hipervisores) tiene una plantilla vinculada que incluye una regla de descubrimiento de bajo nivel con su propio prototipo de equipo (utilizado para descubrir máquinas virtuales), Zabbix descubrirá tanto los hipervisores como sus máquinas virtuales. Para descubrir además contenedores en esas máquinas virtuales, puede crear reglas de descubrimiento de bajo nivel o vincular una nueva plantilla con prototipos de equipo a las máquinas virtuales descubiertas o preconfigurar el prototipo de equipo de la máquina virtual vinculando una plantilla que a su vez incluya prototipos de equipo.

Los equipos descubiertos pueden eliminarse manualmente. Sin embargo, tenga en cuenta que se descubrirán de nuevo si el descubrimiento está habilitado para ellos.

Los equipos que ya no se descubren pueden ser:

- deshabilitados automáticamente (según el valor de *Deshabilitar recursos perdidos* de la regla de descubrimiento)
- eliminados automáticamente (según el valor de *Eliminar recursos perdidos* de la regla de descubrimiento).

Prototipos de descubrimiento

Descripción general

Los prototipos de descubrimiento son reglas de **descubrimiento de bajo nivel anidadas** dentro de una regla de descubrimiento "principal", lo que permite realizar un descubrimiento de varios niveles de objetos con sus propios items, triggers, etc. Por ejemplo, puede querer descubrir todas las instancias de bases de datos en un servidor de bases de datos, después descubrir los espacios de tablas de cada instancia y, finalmente, descubrir las tablas de cada espacio de tablas.

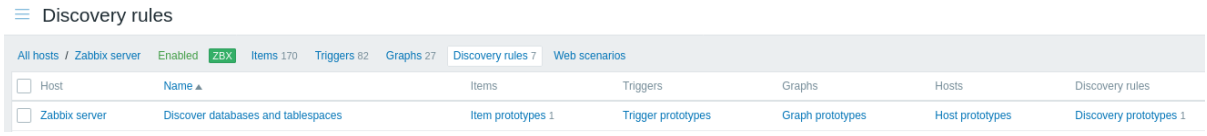
Los prototipos de descubrimiento tienen sus propios prototipos de item, trigger, gráfico, host y descubrimiento. Un prototipo de descubrimiento anidado utilizará el mismo valor JSON que la regla principal si especifica un tipo *Nested*.

Los niveles de anidamiento de los prototipos de descubrimiento son ilimitados.

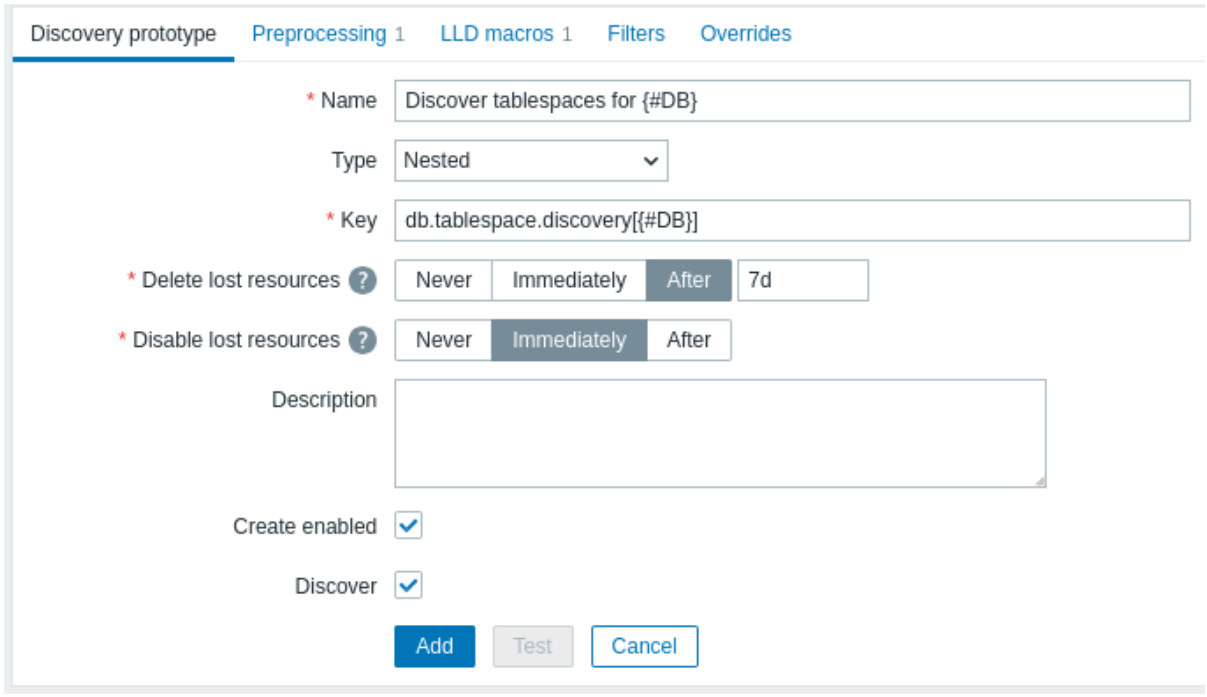
Configuración

Para crear un prototipo de descubrimiento:

- Haga clic en *Prototipos de descubrimiento* en la fila de una regla de descubrimiento existente



- Haga clic en *Crear prototipo de descubrimiento*



Los campos de configuración de este formulario se comparten con el descubrimiento de bajo nivel habitual.

Si selecciona "Nested" como *Type* en el formulario de prototipo de descubrimiento abierto, entonces las reglas de descubrimiento (a partir del prototipo de descubrimiento) se generan en función de un objeto JSON del mismo valor JSON que la regla de descubrimiento principal. Por ejemplo, si el JSON original es [<object A>, <object B>] y hay un prototipo de regla de descubrimiento anidada, se generarán dos reglas de descubrimiento basadas respectivamente en los datos de object A y object B.

En este caso, el prototipo de descubrimiento se activa al mismo tiempo que la regla principal. Por lo tanto, la regla anidada puede usar preprocesamiento para trabajar sobre otro "fragmento" de los mismos datos, ya adquiridos por la regla principal.

Las macros LLD de la regla LLD principal están disponibles para las reglas de descubrimiento anidadas.

Reglas LLD anidadas en equipos descubiertos

Una regla de descubrimiento de bajo nivel *Anidada* puede usarse en una plantilla de equipo asignada a un prototipo de equipo. Si existe una regla de descubrimiento *Anidada* en un equipo descubierto, entonces el objeto JSON utilizado para descubrir el equipo también se envía a todas las reglas LLD de tipo anidado en este equipo. Para más detalles, consulte el ejemplo.

Las macros LLD de la regla de descubrimiento que creó el equipo están disponibles para las reglas de descubrimiento anidadas.

Ejemplo

Ilustremos una posible aplicación de los prototipos de descubrimiento, basada en la recepción del siguiente ejemplo de JSON multinivel.

```
[
  {
    "database": "db1",
    "created_at": "2024-02-01T12:30:00Z",
    "encoding": "UTF8",
    "tablespaces": [
      { "name": "ts1", "max_size": "10GB" },
      { "name": "ts2", "max_size": "20GB" },
      { "name": "ts3", "max_size": "15GB" }
```

```
    ]
  },
  {
    "database": "db2",
    "created_at": "2023-11-15T08:45:00Z",
    "encoding": "UTF16",
    "tablespaces": [
      { "name": "ts1", "max_size": "5GB" },
      { "name": "ts2", "max_size": "25GB" },
      { "name": "ts3", "max_size": "30GB" }
    ]
  },
  {
    "database": "db3",
    "created_at": "2024-01-05T15:10:00Z",
    "encoding": "UTF8",
    "tablespaces": [
      { "name": "ts1", "max_size": "12GB" },
      { "name": "ts2", "max_size": "18GB" },
      { "name": "ts3", "max_size": "22GB" }
    ]
  }
]
```

Caso 1

Descubrir instancias de base de datos en un server de base de datos y, a continuación, descubrir los tablespaces de cada instancia.

1. Tiene al menos un host relacionado con el descubrimiento del server de base de datos.
2. Cree una regla LLD para este host con el nombre *Descubrir bases de datos y tablespaces*.
3. Cambie a la pestaña *Macros LLD* de esta regla y añada la macro `{#DB}=$.database`.
4. Añada un prototipo de item para esta regla con el nombre *Conexiones activas a {#DB}* (Tipo: Agent, Clave: `db.connections[{#DB}]`)
5. Se descubren los items relacionados con cada base de datos:

```
Active connections to db1, Active connections to db2, Active connections to db3.
```

6. Cree un prototipo de descubrimiento para esta regla con el nombre *Descubrir tablespaces para {#DB}* (Tipo: Nested, Clave: `db.tablespace.discovery[{#DB}]`).
7. Cambie a la pestaña *Preprocessing* de este prototipo de descubrimiento y añada el paso `JSONPath=$.tablespaces`.
8. Cambie a la pestaña *Macros LLD* de este prototipo de descubrimiento y añada la macro `{#TSNAME}=$.name`.
9. Cree un prototipo de item para este prototipo de descubrimiento con el nombre *Tamaño del tablespace {#TSNAME} para {#DB}* (Tipo: Agent, Clave: `db.ts.size[{#DB}, {#TSNAME}]`).
10. Se descubren los items relacionados con cada tablespace de cada base de datos:

```
Size of tablespace ts1 for db1, Size of tablespace ts2 for db1, Size of tablespace ts3 for db1,
Size of tablespace ts1 for db2, Size of tablespace ts2 for db2, Size of tablespace ts3 for db2,
Size of tablespace ts1 for db3, Size of tablespace ts2 for db3, Size of tablespace ts3 for db3.
```

con las claves `db.ts.size[db1,ts1]`, `db.ts.size[db1,ts2]`, ... `db.ts.size[db3,ts3]`.

Caso 2

Descubrir instancias de base de datos en el server de base de datos representándolas como hosts descubiertos, y luego descubrir los tablespaces de cada instancia.

1. Tiene al menos un host (host raíz) relacionado con el descubrimiento del server de base de datos.
2. Cree un template para descubrir los tablespaces de cada base de datos.
3. Cree un item en este template llamado *Conexiones activas a {#DB}* (Tipo: Agent, Clave: `db.connections[{#DB}]`).
4. Cree una regla LLD para este template llamada *Descubrir tablespaces* (Tipo: Nested).
5. Cambie a la pestaña *Preprocessing* de esta regla y agregue el paso `JSONPath=$.tablespaces`.

6. Cambie a la pestaña *LLD Macros* de esta regla, agregue la macro `{#TSNAME}=$.name`.
7. Cree un prototipo de item para esta regla llamado *Tamaño del tablespace {#TSNAME} para {#DB}* (Tipo: Agent, Clave: `db.ts.size[{#DB}, {#TSNAME}]`).
8. De vuelta en el host raíz, cree una regla LLD para este host llamada *Descubrir bases de datos y tablespaces*.
9. Cambie a la pestaña *LLD Macros* de esta regla, agregue la macro `{#DB}=$.database`.
10. Agregue un prototipo de host para esta regla llamado *Host para la base de datos {#DB}*.
11. Cambie a la pestaña *Macros* de este prototipo de host, agregue la macro `{$DB}={#DB}` (para el nombre y la clave del item del paso 3).
12. Vincule el template del paso 2 a este prototipo de host.
13. Los hosts descubiertos contienen los items descubiertos relacionados con cada base de datos y sus tablespaces:

Host	Items
Host para la base de datos db1	Conexiones activas a db1
	Tamaño del tablespace ts1 para db1
	Tamaño del tablespace ts2 para db1
	Tamaño del tablespace ts3 para db1
Host para la base de datos db2	Conexiones activas a db2
	Tamaño del tablespace ts1 para db2
	Tamaño del tablespace ts2 para db2
	Tamaño del tablespace ts3 para db2
Host para la base de datos db3	Conexiones activas a db3
	Tamaño del tablespace ts1 para db3
	Tamaño del tablespace ts2 para db3
	Tamaño del tablespace ts3 para db3

Notas sobre la detección de bajo nivel

Uso de macros LLD en contextos de macros de usuario

Las macros LLD pueden utilizarse dentro del contexto de macros de usuario, por ejemplo, en prototipos de trigger.

Varias reglas de LLD para el mismo item

Es posible definir varias reglas de descubrimiento de bajo nivel con el mismo item de descubrimiento.

Para ello, debe definir el parámetro Alias del agent [/manual/concepts/agent/agentd_params], que le permitirá utilizar claves de item de descubrimiento modificadas en diferentes reglas de descubrimiento, por ejemplo, `vfs.fs.discovery[foo]`, `vfs.fs.discovery[bar]`, etc.

Límites de datos para los valores de retorno

No hay límite para los datos JSON de reglas de bajo nivel si se reciben directamente por el servidor Zabbix. Esto se debe a que los valores de retorno se procesan sin almacenarse en una base de datos.

Tampoco hay límite para las reglas de bajo nivel personalizadas. Sin embargo, si los datos de la regla de bajo nivel personalizada se recuperan utilizando un parámetro de usuario, se aplica el límite de valor de retorno del parámetro de usuario.

Si los datos deben pasar por el proxy Zabbix, debe almacenar estos datos en la base de datos. En tal caso, se aplican los límites de la base de datos.

Ejemplos de reglas de descubrimiento

Esta sección proporciona ejemplos prácticos de configuración de reglas de descubrimiento para diferentes tipos de entidades, como sistemas de archivos, interfaces de red, servicios, sensores, objetos SNMP, aplicaciones Java y bases de datos.

Cada ejemplo ilustra cómo se puede estructurar una regla de descubrimiento, incluido el item key o la consulta utilizada, la salida JSON con macros de LLD y cómo se usan esos resultados para crear prototipos de item, trigger y gráfico.

- Descubrimiento de dispositivos de bloque
- Descubrimiento de CPU y núcleos de CPU
- Descubrimiento de interfaces de host en Zabbix
- Descubrimiento de sensores IPMI

- Descubrimiento de objetos JMX
- Descubrimiento de sistemas de archivos montados
- Descubrimiento de interfaces de red
- Descubrimiento de OID SNMP
- Descubrimiento de OID SNMP (heredado)
- Descubrimiento de servicios systemd
- Descubrimiento de instancias de contadores de rendimiento de Windows
- Descubrimiento de servicios de Windows
- Descubrimiento mediante consultas WMI
- Descubrimiento mediante consultas SQL ODBC
- Descubrimiento mediante datos de Prometheus

Reglas LLD personalizadas

Resumen

También es posible crear una regla de LLD completamente personalizada, que descubra cualquier tipo de entidades; por ejemplo, bases de datos en un server de bases de datos.

Para ello, se debe crear un item personalizado que devuelva una cadena JSON, especificando los objetos encontrados y, opcionalmente, algunas de sus propiedades. La cantidad de macros por entidad no está limitada; mientras que las reglas de descubrimiento integradas devuelven una o dos macros (por ejemplo, dos para el descubrimiento de sistemas de archivos), es posible devolver más.

Ejemplo

El formato requerido de la cadena JSON se ilustra mejor con un ejemplo. Suponga que está ejecutando un agent antiguo de Zabbix 1.8 (uno que no admite la clave `vfs.fs.discovery`), pero aun así necesita descubrir sistemas de archivos. Aquí tiene un script Perl simple para Linux que descubre los sistemas de archivos montados y genera la salida en JSON, que incluye tanto el nombre como el tipo del sistema de archivos. Una forma de usarlo sería como un UserParameter con la clave `vfs.fs.discovery_perl`:

```
####!/usr/bin/perl

$first = 1;

print "[\n";

for (`cat /proc/mounts`)
{
    ($fsname, $fstype) = m/\S+ (\S+) (\S+)/;

    print "\t,\n" if not $first;
    $first = 0;

    print "\t{\n";
    print "\t\t\"{#FSNAME}\":\"$fsname\",\n";
    print "\t\t\"{#FSTYPE}\":\"$fstype\"\n";
    print "\t}\n";
}

print "]\n";
```

Attention:
Los símbolos permitidos para los nombres de macros LLD son **0-9** , **A-Z** , **_** , **.**. Las letras minúsculas no son compatibles en los nombres.

A continuación se muestra un ejemplo de su salida (reformateada para mayor claridad). El JSON para las comprobaciones de descubrimiento personalizadas debe seguir el mismo formato.

```
[
    { "{#FSNAME}":"/",                      "{#FSTYPE}":"rootfs"   },
    { "{#FSNAME}":"/sys",                   "{#FSTYPE}":"sysfs"    },
    { "{#FSNAME}":"/proc",                  "{#FSTYPE}":"proc"     },
    { "{#FSNAME}":"/dev",                   "{#FSTYPE}":"devtmpfs" },
    { "{#FSNAME}":"/dev/pts",               "{#FSTYPE}":"devpts"   },
```

```
    { "{#FSNAME}":"/lib/init/rw",               "{#FSTYPE}":"tmpfs"    },
    { "{#FSNAME}":"/dev/shm",                   "{#FSTYPE}":"tmpfs"    },
    { "{#FSNAME}":"/home",                      "{#FSTYPE}":"ext3"     },
    { "{#FSNAME}":"/tmp",                       "{#FSTYPE}":"ext3"     },
    { "{#FSNAME}":"/usr",                       "{#FSTYPE}":"ext3"     },
    { "{#FSNAME}":"/var",                       "{#FSTYPE}":"ext3"     },
    { "{#FSNAME}":"/sys/fs/fuse/connections",   "{#FSTYPE}":"fusectl"  }
]
```

En el ejemplo anterior, se requiere que las claves coincidan con los nombres de macros LLD utilizados en los prototipos; la alternativa es extraer los valores de las macros LLD usando JSONPath {#FSNAME} → $.fsname y {#FSTYPE} → $.fstype, haciendo así posible un script como este:

```
####!/usr/bin/perl

$first = 1;

print "[\n";

for (`cat /proc/mounts`)
{
    ($fsname, $fstype) = m/\S+ (\S+) (\S+)/;

    print "\t,\n" if not $first;
    $first = 0;

    print "\t{\n";
    print "\t\t\"fsname\":\"$fsname\",\n";
    print "\t\t\"fstype\":\"$fstype\"\n";
    print "\t}\n";
}

print "]\n";
```

A continuación se muestra un ejemplo de su salida (reformateada para mayor claridad).

```
[
    { "fsname":"/",                         "fstype":"rootfs"   },
    { "fsname":"/sys",                      "fstype":"sysfs"    },
    { "fsname":"/proc",                     "fstype":"proc"     },
    { "fsname":"/dev",                      "fstype":"devtmpfs" },
    { "fsname":"/dev/pts",                  "fstype":"devpts"   },
    { "fsname":"/lib/init/rw",              "fstype":"tmpfs"    },
    { "fsname":"/dev/shm",                  "fstype":"tmpfs"    },
    { "fsname":"/home",                     "fstype":"ext3"     },
    { "fsname":"/tmp",                      "fstype":"ext3"     },
    { "fsname":"/usr",                      "fstype":"ext3"     },
    { "fsname":"/var",                      "fstype":"ext3"     },
    { "fsname":"/sys/fs/fuse/connections",  "fstype":"fusectl"  }
]
```

Luego, en el campo *Filter* de la regla de descubrimiento, puede especificar {#FSTYPE} como macro y `rootfs|ext3` como expresión regular.

Note:
No tiene que usar nombres de macro como FSNAME/FSTYPE con reglas LLD personalizadas; puede usar libremente los nombres que desee. En caso de que se use JSONPath, la fila LLD será un elemento de matriz que puede ser un objeto, pero también puede ser otra matriz o un valor.

Tenga en cuenta que, si se usa un parámetro de usuario, el valor devuelto está limitado a 16 MB. Para más detalles, consulte los límites de datos para los valores devueltos de LLD.

Supervisión de máquinas virtuales

Descripción general Zabbix puede utilizar reglas de descubrimiento de bajo nivel para descubrir automáticamente hipervisores VMware y máquinas virtuales, y crear equipos para monitorizarlos basándose en prototipos de equipo predefinidos.

Zabbix también incluye plantillas listas para usar para monitorizar hipervisores VMware vCenter o ESXi.

La versión mínima requerida de VMware vCenter o vSphere es la 5.1.

Recopilación de datos La monitorización de máquinas virtuales consta de dos pasos:

1. Los procesos *vmware collector* de Zabbix recopilan datos de máquinas virtuales: los procesos obtienen la información necesaria de los servicios web de VMware a través del protocolo SOAP, la preprocesan y la almacenan en la memoria compartida del servidor Zabbix.
2. Los procesos *poller* de Zabbix recuperan los datos utilizando las claves de métricas de comprobación simple de Zabbix VMware monitoring item keys.

Zabbix divide los datos recopilados en datos de configuración de VMware y datos de contadores de rendimiento de VMware. Ambos tipos de datos se recopilan de forma independiente por los procesos *vmware collector*.

Las siguientes estadísticas están disponibles en función de la información de los contadores de rendimiento de VMware:

- Almacén de datos
- Dispositivo de disco
- CPU
- Energía
- Interfaz de red
- Métricas personalizadas de contadores de rendimiento

Para la lista completa de métricas que obtienen datos de los contadores de rendimiento de VMware, consulte *VMware monitoring item keys*.

Tenga en cuenta que la frecuencia de recuperación de eventos de VMware depende del intervalo de sondeo de vmware.eventlog, pero no puede ser inferior a 5 segundos.

Configuración Si Zabbix server se compila a partir de sources, debe compilarse con las opciones de configuración `--with-libcurl --with-libxml2` para habilitar la supervisión de máquinas virtuales. Los paquetes de Zabbix ya se compilan con estas opciones habilitadas.

Los siguientes parámetros del archivo de configuración de Zabbix server se pueden modificar para la supervisión de máquinas virtuales:

- `StartVMwareCollectors`

Note:
Se recomienda habilitar más collectors que el número de servicios VMware supervisados; de lo contrario, la obtención de estadísticas de contadores de rendimiento de VMware podría retrasarse debido a la obtención de datos de configuración de VMware (lo que lleva un tiempo en instalaciones grandes).

 En general, el valor de `StartVMwareCollectors` no debe ser inferior a 2 ni superior al doble de la cantidad de servicios VMware supervisados: Cantidad de servicios < `StartVMwareCollectors` < (Cantidad de servicios * 2). Por ejemplo, al supervisar un servicio VMware, establezca `StartVMwareCollectors` en 2; al supervisar tres servicios, establezca `StartVMwareCollectors` en 5.

 Tenga en cuenta que el número requerido de collectors también depende del alcance del entorno VMware, así como de los parámetros de configuración `VMwareFrequency` y `VMwarePerfFrequency`.

- `VMwareCacheSize`
- `VMwareFrequency`
- `VMwarePerfFrequency`
- `VMwareTimeout`

Attention:
Para admitir métricas de capacidad de datastore, asegúrese de que el valor de la clave VMware `vpxd.stats.maxQueryMetrics` esté establecido en al menos 64. Para obtener más información, consulte el artículo de la *VMware Knowledge Base*.

Descubrimiento

Zabbix puede usar reglas de descubrimiento de bajo nivel (por ejemplo, `vmware.hv.discovery[{$VMWARE.URL}]`) para descubrir automáticamente hipervisores y máquinas virtuales de VMware. Además, Zabbix puede usar prototipos de host para generar automáticamente hosts reales para las entidades descubiertas. Para obtener más información, consulte *Prototipos de host*.

Ejemplos de configuración

Para un ejemplo básico de cómo configurar Zabbix para supervisar VMware usando el template *VMware FQDN*, consulte *Monitor VMware with Zabbix*.

Para un ejemplo más detallado de cómo crear un host, una regla de descubrimiento de bajo nivel y un prototipo de host para supervisar VMware, consulte *Setup example*.

Registro extendido Los datos recopilados por los procesos de *vmware collector* se pueden registrar para realizar una depuración detallada mediante el nivel de depuración 5.
El nivel de depuración se puede configurar en los archivos de configuración de server y proxy, o mediante la opción de control en tiempo de ejecución `-R log_level_increase="vmware collector,N"`, donde `N` es el número del proceso.

Por ejemplo, para aumentar el nivel de depuración de 4 a 5 para todos los procesos de *vmware collector*, ejecute el siguiente comando:

```
zabbix_server -R log_level_increase="vmware collector"
```

Para aumentar el nivel de depuración de 4 a 5 para el segundo proceso de *vmware collector*, ejecute el siguiente comando:

```
zabbix_server -R log_level_increase="vmware collector,2"
```

Cuando ya no sea necesario el registro extendido de los datos del recopilador de VMware, se recomienda reducir el nivel de depuración al valor predeterminado (3) ejecutando el comando `-R log_level_decrease`.

Solución de problemas

- En caso de que las métricas no estén disponibles, asegúrese de que no hayan sido deshabilitadas o desactivadas de forma predeterminada en versiones recientes de VMware vSphere, o de que no se hayan impuesto límites a las consultas de la base de datos de métricas de rendimiento. Para obtener más información, consulte ZBX-12094.
- Si `config.vpxd.stats.maxQueryMetrics` no es válido o supera el error de número máximo de caracteres permitido, agregue un parámetro `config.vpxd.stats.maxQueryMetrics` a la configuración de vCenter Server. El valor de este parámetro debe ser el mismo que el valor de `maxQuerysize` en el archivo `web.xml` de VMware. Para obtener más información, consulte el artículo de la *Base de conocimientos de VMware*.
- Si sospecha que su instalación de Zabbix está usando demasiada memoria, consulte Perfilado del uso excesivo de memoria con tcmalloc.

Claves de item de monitoreo de VMware

Descripción general Esta página proporciona detalles sobre las comprobaciones simples que se pueden utilizar para monitorizar entornos VMware. Las métricas se agrupan según el objetivo de monitorización.

Claves de item compatibles Las claves de item se enumeran sin parámetros ni información adicional. Haga clic en la clave de item para ver los detalles completos.

Clave de item	Descripción	Grupo de item
vmware.eventlog	El registro de eventos de VMware.	Servicio general
vmware.fullname	El nombre completo del servicio de VMware.	
vmware.version	La versión del servicio de VMware.	
vmware.cl.perfcounter	Las métricas del contador de rendimiento del clúster de VMware.	Clúster
vmware.cluster.alarms.get	Los datos de alarmas del clúster de VMware.	
vmware.cluster.discovery	El descubrimiento de clústeres de VMware.	
vmware.cluster.property	La propiedad del clúster de VMware.	
vmware.cluster.status	El estado del clúster de VMware.	
vmware.cluster.tags.get	El array de etiquetas del clúster de VMware.	
vmware.datastore.alarms.get	Los datos de alarmas del datastore de VMware.	Datastore

Clave de item	Descripción	Grupo de item
vmware.datastore.discovery	El descubrimiento de datastores de VMware.	
vmware.datastore.hv.list	La lista de hipervisores del datastore.	
vmware.datastore.perfcounter	El valor del contador de rendimiento del datastore de VMware.	
vmware.datastore.property	La propiedad del datastore de VMware.	
vmware.datastore.read	El tiempo que tarda una operación de lectura desde el datastore.	
vmware.datastore.size	El espacio del datastore de VMware en bytes o en porcentaje del total.	
vmware.datastore.tags.get	El array de etiquetas del datastore de VMware.	
vmware.datastore.write	El tiempo que tarda una operación de escritura en el datastore.	
vmware.dc.alarms.get	Los datos de alarmas del centro de datos de VMware.	Centro de datos
vmware.dc.discovery	El descubrimiento de centros de datos de VMware.	
vmware.dc.tags.get	El array de etiquetas del centro de datos de VMware.	
vmware.dvswitch.discovery	El descubrimiento de VMware vSphere Distributed Switches.	vSphere Distributed Switch
vmware.dvswitch.fetchportsget	Los datos de puertos de VMware vSphere Distributed Switch.	
vmware.hv.alarms.get	Los datos de alarmas del hipervisor de VMware.	Hipervisor
vmware.hv.cluster.name	El nombre del clúster del hipervisor de VMware.	
vmware.hv.connectionstate	El estado de conexión del hipervisor de VMware.	
vmware.hv.cpu.usage	El uso del procesador del hipervisor de VMware (Hz).	
vmware.hv.cpu.usage.perf	El uso del procesador del hipervisor de VMware como porcentaje durante el intervalo.	
vmware.hv.cpu.utilization	El uso del procesador del hipervisor de VMware como porcentaje durante el intervalo, depende de la gestión de energía o de HT.	
vmware.hv.datacenter.name	El nombre del centro de datos del hipervisor de VMware.	
vmware.hv.datastore.discovery	El descubrimiento de datastores del hipervisor de VMware.	
vmware.hv.datastore.list	La lista de datastores del hipervisor de VMware.	
vmware.hv.datastore.multipath	El número de rutas de datastore disponibles.	
vmware.hv.datastore.read	El tiempo medio que tarda una operación de lectura desde el datastore.	
vmware.hv.datastore.size	El espacio del datastore de VMware en bytes o en porcentaje del total.	
vmware.hv.datastore.write	El tiempo medio que tarda una operación de escritura en el datastore.	
vmware.hv.discovery	El descubrimiento de hipervisores de VMware.	
vmware.hv.diskinfo.get	Los datos de disco del hipervisor de VMware.	
vmware.hv.fullname	El nombre completo del producto, incluida la información de versión.	
vmware.hv.hw.cpu.freq	La frecuencia del procesador del hipervisor de VMware.	
vmware.hv.hw.cpu.model	El modelo del procesador del hipervisor de VMware.	
vmware.hv.hw.cpu.num	El número de núcleos del procesador en el hipervisor de VMware.	
vmware.hv.hw.cpu.threads	El número de hilos del procesador en el hipervisor de VMware.	
vmware.hv.hw.memory	El tamaño total de memoria del hipervisor de VMware.	
vmware.hv.hw.model	El modelo del hipervisor de VMware.	
vmware.hv.hw.sensors.get	El valor de los sensores de hardware del hipervisor de VMware.	
vmware.hv.hw.serialnumber	El número de serie del hipervisor de VMware.	
vmware.hv.hw.uuid	El UUID de BIOS del hipervisor de VMware.	
vmware.hv.hw.vendor	El nombre del proveedor del hipervisor de VMware.	
vmware.hv.maintenance	El estado de mantenimiento del hipervisor de VMware.	
vmware.hv.memory.size.ballooned	El tamaño de la memoria ballooned del hipervisor de VMware.	
vmware.hv.memory.used	El tamaño de la memoria utilizada del hipervisor de VMware.	
vmware.hv.net.if.discovery	El descubrimiento de interfaces de red del hipervisor de VMware.	
vmware.hv.network.in	Las estadísticas de entrada de red del hipervisor de VMware.	
vmware.hv.network.linkspeed	La velocidad de la interfaz de red del hipervisor de VMware.	
vmware.hv.network.out	Las estadísticas de salida de red del hipervisor de VMware.	
vmware.hv.perfcounter	El valor del contador de rendimiento del hipervisor de VMware.	
vmware.hv.property	La propiedad del hipervisor de VMware.	
vmware.hv.power	El uso de energía del hipervisor de VMware.	
vmware.hv.sensor.health.state	El sensor acumulado del estado de salud del hipervisor de VMware.	
vmware.hv.sensors.get	Los sensores de estado del proveedor de HW del hipervisor de VMware.	
vmware.hv.status	El estado del hipervisor de VMware.	
vmware.hv.tags.get	El array de etiquetas del hipervisor de VMware.	
vmware.hv.uptime	El tiempo de actividad del hipervisor de VMware.	
vmware.hv.version	La versión del hipervisor de VMware.	
vmware.hv.vm.num	El número de máquinas virtuales en el hipervisor de VMware.	

Clave de item	Descripción	Grupo de item
vmware.rp.cpu.usage	El uso de CPU en hercios durante el intervalo en VMware Resource Pool.	Resource pool
vmware.rp.memory	Las métricas de memoria del resource pool de VMware.	
vmware.alarms.get	Los datos de alarmas del centro virtual de VMware.	Virtual center
vmware.vm.alarms.get	Los datos de alarmas de la máquina virtual de VMware.	Máquina virtual
vmware.vm.attribute	El valor del atributo personalizado de la máquina virtual de VMware.	
vmware.vm.cluster.name	El nombre de la máquina virtual de VMware.	
vmware.vm.consolidationneeded	El disco de la máquina virtual de VMware requiere consolidación.	
vmware.vm.cpu.latency	El porcentaje de tiempo en que la máquina virtual no puede ejecutarse porque compite por el acceso a la(s) CPU física(s).	
vmware.vm.cpu.num	El número de procesadores en la máquina virtual de VMware.	
vmware.vm.cpu.readiness	El porcentaje de tiempo en que la máquina virtual estaba lista, pero no pudo ser programada para ejecutarse en la CPU física.	
vmware.vm.cpu.ready	El tiempo en que la máquina virtual estaba lista, pero no pudo ser programada para ejecutarse en la CPU física.	
vmware.vm.cpu.swapwait	El porcentaje de tiempo de CPU empleado esperando el swap-in.	
vmware.vm.cpu.usage	El uso del procesador de la máquina virtual de VMware (Hz).	
vmware.vm.cpu.usage.perf	El uso del procesador de la máquina virtual de VMware como porcentaje durante el intervalo.	
vmware.vm.datacenter.name	El nombre del centro de datos de la máquina virtual de VMware.	
vmware.vm.discovery	El descubrimiento de máquinas virtuales de VMware.	
vmware.vm.guest.memory.swapped	La cantidad de memoria física del guest que se ha intercambiado al espacio de swap.	
vmware.vm.guest.osuptime	El tiempo total transcurrido desde el último arranque del sistema operativo.	
vmware.vm.hv.maintenance	El estado de mantenimiento del hipervisor de la máquina virtual de VMware.	
vmware.vm.hv.name	El nombre del hipervisor de la máquina virtual de VMware.	
vmware.vm.memory.size	El tamaño total de memoria de la máquina virtual de VMware.	
vmware.vm.memory.size.ballooned	El tamaño de la memoria ballooned de la máquina virtual de VMware.	
vmware.vm.memory.size.compressed	El tamaño de la memoria comprimida de la máquina virtual de VMware.	
vmware.vm.memory.size.consumed	La cantidad de memoria física del host consumida para respaldar páginas de memoria física del guest.	
vmware.vm.memory.size.private	El tamaño de la memoria privada de la máquina virtual de VMware.	
vmware.vm.memory.size.shared	El tamaño de la memoria compartida de la máquina virtual de VMware.	
vmware.vm.memory.size.swapped	El tamaño de la memoria intercambiada de la máquina virtual de VMware.	
vmware.vm.memory.size.usage.guest	El uso de memoria del guest de la máquina virtual de VMware.	
vmware.vm.memory.size.usage.host	El uso de memoria del host de la máquina virtual de VMware.	
vmware.vm.memory.usage	El porcentaje de memoria física del host que se ha consumido.	
vmware.vm.net.if.discovery	El descubrimiento de interfaces de red de la máquina virtual de VMware.	
vmware.vm.net.if.in	Las estadísticas de entrada de la interfaz de red de la máquina virtual de VMware.	
vmware.vm.net.if.out	Las estadísticas de salida de la interfaz de red de la máquina virtual de VMware.	
vmware.vm.net.if.usage	La utilización de red de la máquina virtual de VMware durante el intervalo.	
vmware.vm.perfcounter	El valor del contador de rendimiento de la máquina virtual de VMware.	
vmware.vm.powerstate	El estado de energía de la máquina virtual de VMware.	
vmware.vm.property	La propiedad de la máquina virtual de VMware.	
vmware.vm.snapshot.get	El estado de instantánea de la máquina virtual de VMware.	
vmware.vm.state	El estado de la máquina virtual de VMware.	
vmware.vm.storage.committed	El espacio de almacenamiento comprometido de la máquina virtual de VMware.	
vmware.vm.storage.readoio	El número medio de solicitudes de lectura pendientes al disco virtual durante el intervalo de recopilación.	
vmware.vm.storage.totalreadlatency	El tiempo medio que tarda una lectura del disco virtual.	
vmware.vm.storage.totalwritelatency	El tiempo medio que tarda una escritura en el disco virtual.	
vmware.vm.storage.uncommitted	El espacio de almacenamiento no comprometido de la máquina virtual de VMware.	
vmware.vm.storage.unshared	El espacio de almacenamiento no compartido de la máquina virtual de VMware.	
vmware.vm.storage.writeoio	El número medio de solicitudes de escritura pendientes al disco virtual durante el intervalo de recopilación.	
vmware.vm.tags.get	El array de etiquetas de la máquina virtual de VMware.	
vmware.vm.tools	El estado o la versión de las herramientas del guest de la máquina virtual de VMware.	
vmware.vm.uptime	El tiempo de actividad de la máquina virtual de VMware.	

Clave de item	Descripción	Grupo de item
vmware.vm.vfs.dev.discovery	El descubrimiento de dispositivos de disco de la máquina virtual de VMware.	
vmware.vm.vfs.dev.read	Las estadísticas de lectura del dispositivo de disco de la máquina virtual de VMware.	
vmware.vm.vfs.dev.write	Las estadísticas de escritura del dispositivo de disco de la máquina virtual de VMware.	
vmware.vm.vfs.fs.discovery	El descubrimiento de sistemas de archivos de la máquina virtual de VMware.	
vmware.vm.vfs.fs.size	Las estadísticas del sistema de archivos de la máquina virtual de VMware.	

Detalles de la clave de la métrica Los parámetros sin corchetes angulares son obligatorios. Los parámetros marcados con corchetes angulares **< >** son opcionales.

vmware.eventlog[url,<mode>,<severity>]

 El registro de eventos de VMware.
 Valor de retorno: *Log*.

Parámetros:

- **url** - la URL del servicio VMware;
- **mode** - *all* (predeterminado) o *skip* - omitir el procesamiento de datos antiguos;
- **severity** - filtrar por gravedad: *error*, *warning*, *info* o *user*. Este parámetro debe ir entre comillas si se especifica más de una gravedad en una lista delimitada por comas (p. ej.`"error,warning,info,user"`). Deshabilitado de forma predeterminada.

Comentarios:

- Debe haber solo una clave de item `vmware.eventlog` por URL;
- Véase también el ejemplo de filtrado de registros del registro de eventos de VMware.

vmware.fullnameurl

 El nombre completo del servicio VMware.
 Valor de retorno: *String*.

Parámetros:

- **url** - la URL del servicio VMware.

vmware.versionurl

 La versión del servicio VMware.
 Valor de retorno: *Cadena*.

Parámetros:

- **url** - la URL del servicio VMware.

vmware.cl.perfcounter[url,id,path,<instance>]

 Las métricas del contador de rendimiento del clúster de VMware.
 Valor de retorno: *Entero*.

Parámetros:

- **url**: la URL del servicio VMware;
- **id**: el ID del clúster de VMware. El `id` se puede obtener de `vmware.cluster.discovery[]` como `{#CLUSTER.ID}`.
- **path**: la ruta del contador de rendimiento[1];
- **instance**: la instancia del contador de rendimiento.

vmware.cluster.alarms.get[url,id]

 Los datos de alarmas del clúster de VMware.
 Valor devuelto: JSON object.

Parámetros:

- **url** - la URL del servicio de VMware;
- **id** - el ID del clúster de VMware.

vmware.cluster.discoveryurl

 El descubrimiento de clústeres de VMware.
 Valor de retorno: objeto JSON.

Parámetros:

- **url** - la URL del servicio VMware.

vmware.cluster.property[url,id,prop]

 La propiedad del clúster de VMware.
 Valor de retorno: *Cadena*.

Parámetros:

- **url**: la URL del servicio VMware;
- **id**: el ID del clúster de VMware;
- **prop**: la ruta de la propiedad, que es el nombre de una propiedad del objeto VM según se define en el SDK de VMware.

Ejemplos:

```
vmware.vm.property[{$VMWARE.URL},{$VMWARE.VM.UUID},overallStatus]
vmware.vm.property[{$VMWARE.URL},{$VMWARE.VM.UUID},runtime.powerState]
```

vmware.cluster.status[url,name]

 El estado del clúster de VMware.
 Valor devuelto: *0* - gris; *1* - verde; *2* - amarillo; *3* - rojo.

Parámetros:

- **url** - la URL del servicio VMware;
- **name** - el nombre del clúster de VMware.

vmware.cluster.tags.get[url,id]

 El array de etiquetas del clúster de VMware.
 Valor devuelto: JSON object.

Parámetros:

- **url** - la URL del servicio de VMware;
- **id** - el ID del clúster de VMware.

Este item funciona con vSphere 6.5 y versiones posteriores.

vmware.datastore.alarms.get[url,uuid]

 Los datos de alarmas del datastore de VMware.
 Valor devuelto: JSON object.

Parámetros:

- **url** - la URL del servicio de VMware;
- **uuid** - el identificador único global del datastore de VMware.

vmware.datastore.discovery[url,<filter_uuid>]

 El descubrimiento de almacenes de datos de VMware.
 Valor de retorno: objeto JSON.

Parámetros:

- **url**: la URL del servicio VMware;
- **filter_uuid**: un identificador único de HV o VM (por defecto: vacío). Si se especifica, solo se descubrirán los almacenes de datos vinculados.

vmware.datastore.hv.list[url,datastore]

 La lista de hipervisores del almacén de datos.
 Valor de retorno: *Cadena*.

Parámetros:

- **url**: la URL del servicio VMware;
- **datastore**: el UUID o nombre del almacén de datos.

Ejemplo de salida:

```
esx7-01-host.zabbix.sandbox
esx7-02-host.zabbix.sandbox
```

vmware.datastore.perfcounter[url,uuid,path,<instance>]

 El valor del contador de rendimiento del almacén de datos de VMware.
 Valor devuelto: *Entero* [2].

Parámetros:

- **url**: la URL del servicio VMware;
- **uuid**: el identificador global único del almacén de datos de VMware;
- **path**: la ruta del contador de rendimiento[1];
- **instance**: la instancia del contador de rendimiento. Utilice una instancia vacía para valores agregados (por defecto). `instance` se puede obtener de `vmware.datastore.discovery[]` como parte del array {#DATASTORE.EXTENT}.

vmware.datastore.property[url,uuid,prop]

 La propiedad del almacén de datos de VMware.
 Valor de retorno: *Cadena*.

Parámetros:

- **url**: la URL del servicio VMware;
- **uuid**: el identificador global único del almacén de datos de VMware;
- **prop**: la ruta de la propiedad.

vmware.datastore.read[url,datastore,<mode>]

 La cantidad de tiempo para una operación de lectura desde el almacén de datos (milisegundos).
 Valor de retorno: *Integer* [2].

Parámetros:

- **url** - la URL del servicio VMware;
- **datastore** - el UUID o nombre del almacén de datos;
- **mode** - *latency* (valor promedio, por defecto) o *maxlatency* (valor máximo).

vmware.datastore.size[url,datastore,<mode>]

 El espacio del almacén de datos de VMware en bytes o en porcentaje del total.
 Valor devuelto: *Entero* - para bytes; *Flotante* - para porcentaje.

Parámetros:

- **url** - la URL del servicio VMware;
- **datastore** - el UUID o nombre del almacén de datos;
- **mode** - valores posibles: *total* (por defecto), *free*, *pfree* (porcentaje libre), *uncommitted*.

vmware.datastore.tags.get[url,uuid]

 El array de etiquetas del datastore de VMware.
 Valor devuelto: objeto JSON.

Parámetros:

- **url** - la URL del servicio de VMware;
- **uuid** - el identificador único global del datastore de VMware.

Este item funciona con vSphere 6.5 y versiones posteriores.

vmware.datastore.write[url,datastore,<mode>]

 La cantidad de tiempo para una operación de escritura en el almacén de datos (milisegundos).
 Valor de retorno: *Entero* [2].

Parámetros:

- **url** - la URL del servicio VMware;
- **datastore** - el UUID o nombre del almacén de datos;
- **mode** - *latency* (valor promedio, por defecto) o *maxlatency* (valor máximo).

vmware.dc.alarms.get[url,id]

 Los datos de alarmas del centro de datos de VMware.
 Valor devuelto: objeto JSON.

Parámetros:

- **url** - la URL del servicio de VMware;
- **id** - el ID del centro de datos de VMware.

vmware.dc.discoveryurl

 El descubrimiento de centros de datos de VMware.
 Valor de retorno: objeto JSON.

Parámetros:

- **url**: la URL del servicio VMware.

vmware.dc.tags.get[url,id]

 El array de etiquetas del centro de datos de VMware.
 Valor devuelto: objeto JSON.

Parámetros:

- **url** - la URL del servicio de VMware;
- **id** - el ID del centro de datos de VMware.

Este item funciona con vSphere 6.5 y versiones posteriores.

vmware.dvswitch.discoveryurl

 El descubrimiento de VMware vSphere Distributed Switches.
 Valor de retorno: objeto JSON.

Parámetros:

- **url** - la URL del servicio VMware.

vmware.dvswitch.fetchports.get[url,uuid,<filter>,<mode>]

 Los datos de los puertos de VMware vSphere Distributed Switch.
 Valor devuelto: objeto JSON.

Parámetros:

- **url** - la URL del servicio de VMware;
- **uuid** - el identificador único global de VMware DVSwitch.
- **filter** - una sola cadena con criterios separados por comas para seleccionar puertos;
- **mode** - *state* (todo el XML sin nodos XML "config", valor predeterminado) o *full*.

El parámetro **filter** admite los criterios disponibles en el objeto de datos de VMware DistributedVirtualSwitchPortCriteria.

Ejemplo:

```
vmware.dvswitch.fetchports.get[{$VMWARE.URL},{$VMWARE.DVS.UUID},"connected:true,active:true,uplinkPort:fal
```

vmware.hv.alarms.get[url,uuid]

 Los datos de alarmas del hipervisor VMware.
 Valor devuelto: objeto JSON.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador único global del hipervisor VMware.

vmware.hv.cluster.name[url,uuid]

 El nombre del clúster de hipervisores de VMware.
 Valor de retorno: *Cadena*.

Parámetros:

- **url** - la URL del servicio de VMware;
- **uuid** - el identificador único global del hipervisor de VMware.

vmware.hv.connectionstate[url,uuid]

 El estado de conexión del hipervisor de VMware.
 Valor de retorno: *Cadena*: *connected*, *disconnected* o *notResponding*.

Parámetros:

- **url** - la URL del servicio de VMware;
- **uuid** - el identificador global único del hipervisor de VMware.

vmware.hv.cpu.usage[url,uuid]

 El uso del procesador del hipervisor de VMware (Hz).
 Valor devuelto: *Entero*.

Parámetros:

- **url** - la URL del servicio de VMware;
- **uuid** - el identificador global único del hipervisor de VMware.

vmware.hv.cpu.usage.perf[url,uuid]

 El uso del procesador del hipervisor de VMware como porcentaje durante el intervalo.
 Valor de retorno: *Float*.

Parámetros:

- **url** - la URL del servicio de VMware;
- **uuid** - el identificador global único del hipervisor de VMware.

vmware.hv.cpu.utilization[url,uuid]

 El uso del procesador del hipervisor de VMware como porcentaje durante el intervalo, depende de la gestión de energía o HT.
 Valor devuelto: *Float*.

Parámetros:

- **url** - la URL del servicio VMware;

- **uuid** - el identificador global único del hipervisor de VMware.

vmware.hv.datacenter.name[url,uuid]

 El nombre del centro de datos del hipervisor de VMware.
 Valor de retorno: *Cadena*.

Parámetros:

- **url**: la URL del servicio de VMware;
- **uuid**: el identificador global único del hipervisor de VMware.

vmware.hv.datastore.discovery[url,uuid]

 El descubrimiento de almacenes de datos del hipervisor VMware.
 Valor de retorno: objeto JSON.

Parámetros:

- **url**: la URL del servicio VMware;
- **uuid**: el identificador global único del hipervisor VMware.

vmware.hv.datastore.list[url,uuid]

 La lista de almacenes de datos del hipervisor VMware.
 Valor de retorno: *Cadena*.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único del hipervisor VMware.

Ejemplo de salida:

```
SSD-RAID1-VAULT1
SSD-RAID1-VAULT2
SSD-RAID10
```

vmware.hv.datastore.multipath[url,uuid,<datastore>,<partitionid>]

 El número de rutas de almacenamiento disponibles.
 Valor de retorno: *Entero*.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único del hipervisor VMware;
- **datastore** - el UUID o nombre del almacenamiento;
- **partitionid** - el ID interno del dispositivo físico de `vmware.hv.datastore.discovery`.

vmware.hv.datastore.read[url,uuid,datastore,<mode>]

 La cantidad promedio de tiempo para una operación de lectura desde el almacén de datos (milisegundos).
 Valor de retorno: *Integer* [2].

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único del hipervisor VMware;
- **datastore** - el UUID o nombre del almacén de datos;
- **mode** - *latency* (por defecto).

vmware.hv.datastore.size[url,uuid,datastore,<mode>]

 El espacio del almacén de datos de VMware en bytes o en porcentaje del total.
 Valor de retorno: *Entero* - para bytes; *Flotante* - para porcentaje.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único del hipervisor VMware;
- **datastore** - el UUID o nombre del almacén de datos;
- **mode** - valores posibles: *total* (por defecto), *free*, *pfree* (porcentaje libre), *uncommitted*.

vmware.hv.datastore.write[url,uuid,datastore,<mode>]

 La cantidad promedio de tiempo para una operación de escritura en el almacén de datos (milisegundos).
 Valor de retorno: *Entero* [2].

Parámetros:

- **url** - la URL del servicio VMware;

- **uuid** - el identificador global único del hipervisor VMware;
- **datastore** - el UUID o nombre del almacén de datos;
- **mode** - *latency* (por defecto).

vmware.hv.discoveryurl

 El descubrimiento de hipervisores VMware.
 Valor de retorno: objeto JSON.

Parámetros:

- **url** - la URL del servicio VMware.

vmware.hv.diskinfo.get[url,uuid]

 Los datos de disco del hipervisor VMware.
 Valor devuelto: objeto JSON.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador único global del hipervisor VMware.

vmware.hv.fullname[url,uuid]

 El nombre completo del producto, incluida la información de la versión.
 Valor devuelto: *Cadena*.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador único global del hipervisor VMware.

vmware.hv.hw.cpu.freq[url,uuid]

 La frecuencia del procesador del hipervisor VMware (Hz).
 Valor de retorno: *Entero*.

Parámetros:

- **url**: la URL del servicio VMware;
- **uuid**: el identificador global único del hipervisor VMware.

vmware.hv.hw.cpu.model[url,uuid]

 El modelo de procesador del hipervisor VMware.
 Valor devuelto: *Cadena*.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único del hipervisor VMware.

vmware.hv.hw.cpu.num[url,uuid]

 El número de núcleos de procesador en el hipervisor de VMware.
 Valor de retorno: *Entero*.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único del hipervisor de VMware.

vmware.hv.hw.cpu.threads[url,uuid]

 El número de hilos de procesador en el hipervisor VMware.
 Valor devuelto: *Integer*.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único del hipervisor VMware.

vmware.hv.hw.memory[url,uuid]

 El tamaño total de memoria del hipervisor VMware (bytes).
 Valor devuelto: *Entero*.

Parámetros:

- **url**: la URL del servicio VMware;
- **uuid**: el identificador global único del hipervisor VMware.

vmware.hv.hw.model[url,uuid]

 El modelo del hipervisor de VMware.
 Valor de retorno: *Cadena*.

Parámetros:

- **url** - la URL del servicio de VMware;
- **uuid** - el identificador global único del hipervisor de VMware.

vmware.hv.hw.sensors.get[url,uuid]

 El valor de los sensores de hardware del hipervisor VMware.
 Valor devuelto: objeto JSON.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador único global del hipervisor VMware.

vmware.hv.hw.serialnumber[url,uuid]

 El número de serie del hipervisor de VMware.
 Valor devuelto: *Cadena*.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único del hipervisor de VMware.

Esta métrica funciona con la API de vSphere 6.7 y posteriores.

vmware.hv.hw.uuid[url,uuid]

 El UUID de la BIOS del hipervisor de VMware.
 Valor de retorno: *Cadena*.

Parámetros:

- **url** - la URL del servicio de VMware;
- **uuid** - el identificador global único del hipervisor de VMware.

Esta métrica funciona con la API de vSphere 6.7 y versiones posteriores.

vmware.hv.hw.vendor[url,uuid]

 El nombre del proveedor del hipervisor de VMware.
 Valor de retorno: *Cadena*.

Parámetros:

- **url**: la URL del servicio de VMware;
- **uuid**: el identificador global único del hipervisor de VMware.

Esta métrica funciona con la API vSphere 6.7 y versiones posteriores.

vmware.hv.maintenance[url,uuid]

 El estado de mantenimiento del hipervisor de VMware.
 Valor de retorno: *0* - no está en mantenimiento; *1* - está en mantenimiento.

Parámetros:

- **url** - la URL del servicio de VMware;
- **uuid** - el identificador global único del hipervisor de VMware.

vmware.hv.memory.size.ballooned[url,uuid]

 Tamaño de memoria inflada del hipervisor VMware (bytes).
 Valor de retorno: *Entero*.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único del hipervisor VMware.

vmware.hv.memory.used[url,uuid]

 El tamaño de memoria usada por el hipervisor de VMware (bytes).
 Valor de retorno: *Entero*.

Parámetros:

- **url** - la URL del servicio de VMware;
- **uuid** - el identificador global único del hipervisor de VMware.

vmware.hv.net.if.discovery[url,uuid]

 El descubrimiento de interfaces de red del hipervisor VMware.
 Valor de retorno: *objeto JSON*.

Parámetros:

- **url**: la URL del servicio VMware;

- **uuid**: el identificador global único del hipervisor VMware.

vmware.hv.network.in[url,uuid,<mode>]

 Las estadísticas de entrada de red del hipervisor VMware (bytes por segundo).
 Valor de retorno: *Entero* [2].

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único del hipervisor VMware;
- **mode** - *bps* (por defecto), *packets*, *dropped*, *errors*, *broadcast*.

vmware.hv.network.linkspeed[url,uuid,ifname]

 La velocidad de la interfaz de red del hipervisor VMware.
 Valor de retorno: *Entero*. Devuelve *0* si la interfaz de red está inactiva, de lo contrario, el valor de velocidad de la interfaz.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único del hipervisor VMware;
- **ifname** - el nombre de la interfaz.

vmware.hv.network.out[url,uuid,<mode>]

 Las estadísticas de salida de red del hipervisor VMware (bytes por segundo).
 Valor de retorno: *Entero* [2].

Parámetros:

- **url**: la URL del servicio VMware;
- **uuid**: el identificador global único del hipervisor VMware;
- **mode**: *bps* (por defecto), *packets*, *dropped*, *errors*, *broadcast*.

vmware.hv.perfcounter[url,uuid,path,<instance>]

 El valor del contador de rendimiento del hipervisor de VMware.
 Valor de retorno: *Entero* [2].

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único del hipervisor de VMware;
- **path** - la ruta del contador de rendimiento [1];
- **instance** - la instancia del contador de rendimiento. Utilice una instancia vacía para valores agregados (por defecto).

vmware.hv.property[url,uuid,prop]

 La propiedad del hipervisor de VMware.
 Valor de retorno: *Cadena*.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único del hipervisor de VMware;
- **prop** - la ruta de la propiedad.

vmware.hv.power[url,uuid,<max>]

 El uso de energía del hipervisor VMware (W).
 Valor de retorno: *Entero*.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador único global del hipervisor VMware;
- **max** - el uso máximo de energía permitido.

vmware.hv.sensor.health.state[url,uuid]

 El sensor de estado de salud del hipervisor de VMware.
 Valor de retorno: *Entero*: *0* - gris; *1* - verde; *2* - amarillo; *3* - rojo.

Parámetros:

- **url** - la URL del servicio de VMware;
- **uuid** - el identificador global único del hipervisor de VMware.

Tenga en cuenta que la métrica podría no funcionar en VMware vSphere 6.5 y versiones posteriores, porque VMware ha desaprobado el sensor *VMware Rollup Health State*.

vmware.hv.sensors.get[url,uuid]

 Los sensores de estado del proveedor de hardware del hipervisor VMware.
 Valor devuelto: objeto JSON.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador único global del hipervisor VMware.

vmware.hv.status[url,uuid]

 El estado del hipervisor de VMware.
 Valor de retorno: *Entero*: *0* - gris; *1* - verde; *2* - amarillo; *3* - rojo.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único del hipervisor de VMware.

Esta métrica utiliza la propiedad de estado general del sistema del equipo.

vmware.hv.tags.get[url,uuid]

 El array de etiquetas del hipervisor VMware.
 Valor devuelto: objeto JSON.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador único global del hipervisor VMware.

Este item funciona con vSphere 6.5 y versiones posteriores.

vmware.hv.uptime[url,uuid]

 El tiempo de actividad del hipervisor de VMware (segundos).
 Valor devuelto: *Entero*.

Parámetros:

- **url**: la URL del servicio de VMware;
- **uuid**: el identificador global único del hipervisor de VMware.

Esta métrica utiliza la propiedad de estado general del sistema del equipo.

vmware.hv.version[url,uuid]

 La versión del hipervisor de VMware.
 Valor devuelto: *Cadena*.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador único global del hipervisor de VMware.

vmware.hv.vm.num[url,uuid]

 El número de máquinas virtuales en el hipervisor VMware.
 Valor de retorno: *Entero*.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único del hipervisor VMware.

vmware.rp.cpu.usage[url,rpid]

 El uso de CPU en hertzios durante el intervalo en el Pool de Recursos de VMware.
 Valor de retorno: *Entero*.

Parámetros:

- **url** - la URL del servicio VMware;
- **rpid** - el ID del pool de recursos de VMware.

vmware.rp.memory[url,rpid,<mode>]

 Las métricas de memoria del pool de recursos de VMware.
 Valor de retorno: *Entero*.

Parámetros:

- **url** - la URL del servicio VMware;
- **rpid** - el ID del pool de recursos de VMware;
- **mode** - valores posibles:
consumed (por defecto): la cantidad de memoria física del equipo consumida para respaldar las páginas de memoria física del invitado
 ballooned: la cantidad de memoria física del invitado recuperada de la máquina virtual por el controlador balloon en el invitado
overhead: la memoria física del equipo consumida por las estructuras de datos de ESXi para ejecutar las máquinas virtuales

vmware.alarms.geturl

 Los datos de alarmas del centro virtual de VMware.
 Valor devuelto: Objeto JSON.

Parámetros:

- **url** - la URL del servicio de VMware.

vmware.vm.alarms.get[url,uuid]

 Los datos de alarmas de la máquina virtual de VMware.
 Valor devuelto: objeto JSON.

Parámetros:

- **url** - la URL del servicio de VMware;
- **uuid** - el identificador único global de la máquina virtual de VMware.

vmware.vm.attribute[url,uuid,name]

 El valor del atributo personalizado de la máquina virtual de VMware.
 Valor devuelto: *Cadena*.

Parámetros:

- **url** - la URL del servicio de VMware;
- **uuid** - el identificador global único de la máquina virtual de VMware;
- **name** - el nombre del atributo personalizado.

vmware.vm.cluster.name[url,uuid]

 El nombre de la máquina virtual de VMware.
 Valor devuelto: *String*.

Parámetros:

- **url** - la URL del servicio de VMware;
- **uuid** - el identificador único global de la máquina virtual de VMware.

vmware.vm.consolidationneeded[url,uuid]

 El disco de la máquina virtual de VMware requiere consolidación.
 Valor de retorno: *Cadena*: *true* - se necesita consolidación; *false* - no se necesita consolidación.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único de la máquina virtual de VMware.

vmware.vm.cpu.latency[url,uuid]

 El porcentaje de tiempo que la máquina virtual no puede ejecutarse porque está compitiendo por el acceso a la(s) CPU(s) física(s).
 Valor devuelto: *Float*.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único de la máquina virtual VMware.

vmware.vm.cpu.num[url,uuid]

 El número de procesadores en la máquina virtual de VMware.
 Valor de retorno: *Entero*.

Parámetros:

- **url**: la URL del servicio VMware;
- **uuid**: el identificador global único de la máquina virtual de VMware.

vmware.vm.cpu.readiness[url,uuid,<instance>]

 El porcentaje de tiempo que la máquina virtual estuvo lista, pero no pudo ser programada para ejecutarse en la CPU física.
 Valor de retorno: *Float*.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único de la máquina virtual VMware;
- **instance** - la instancia de CPU.

vmware.vm.cpu.ready[url,uuid]

 El tiempo (en milisegundos) que la máquina virtual estuvo lista, pero no pudo ser programada para ejecutarse en la CPU física. El tiempo de CPU ready depende del número de máquinas virtuales en el equipo y de sus cargas de CPU (%).
 Valor de retorno: *Entero* [2].

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único de la máquina virtual VMware.

vmware.vm.cpu.swapwait[url,uuid,<instance>]

 El porcentaje de tiempo de CPU dedicado a esperar por swap-in.
 Valor devuelto: *Float*.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único de la máquina virtual VMware;
- **instance** - la instancia de CPU.

vmware.vm.cpu.usage[url,uuid]

 El uso del procesador de la máquina virtual de VMware (Hz).
 Valor devuelto: *Entero*.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único de la máquina virtual de VMware.

vmware.vm.cpu.usage.perf[url,uuid]

 El uso del procesador de la máquina virtual de VMware como porcentaje durante el intervalo.
 Valor devuelto: *Float*.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único de la máquina virtual de VMware.

vmware.vm.datacenter.name[url,uuid]

 El nombre del centro de datos de la máquina virtual de VMware.
 Valor devuelto: *Cadena*.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único de la máquina virtual de VMware.

vmware.vm.discoveryurl

 El descubrimiento de máquinas virtuales VMware.
 Valor de retorno: objeto JSON.

Parámetros:

- **url**: la URL del servicio VMware.

vmware.vm.guest.memory.size.swapped[url,uuid]

 La cantidad de memoria física de invitado que se ha intercambiado al espacio de intercambio (KB).
 Valor de retorno: *Entero*.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único de la máquina virtual VMware.

vmware.vm.guest.osuptime[url,uuid]

 El tiempo total transcurrido desde el último arranque del sistema operativo (en segundos).
 Valor devuelto: *Entero*.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único de la máquina virtual VMware.

vmware.vm.hv.maintenance[url,uuid]

 El estado de mantenimiento del hipervisor de la máquina virtual de VMware.
 Valor de retorno: *0* - no está en mantenimiento; *1* - está en mantenimiento.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único de la máquina virtual de VMware.

vmware.vm.hv.name[url,uuid]

 El nombre del hipervisor de la máquina virtual de VMware.
 Valor de retorno: *Cadena*.

Parámetros:

- **url** - la URL del servicio de VMware;
- **uuid** - el identificador global único de la máquina virtual de VMware.

vmware.vm.memory.size[url,uuid]

 El tamaño total de memoria (bytes) de la máquina virtual de VMware.
 Valor de retorno: *Entero*.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único de la máquina virtual de VMware.

vmware.vm.memory.size.ballooned[url,uuid]

 El tamaño de memoria inflada (ballooned) de la máquina virtual de VMware (bytes).
 Valor devuelto: *Entero*.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único de la máquina virtual de VMware.

vmware.vm.memory.size.compressed[url,uuid]

 El tamaño de la memoria comprimida de la máquina virtual VMware (bytes).
 Valor de retorno: *Entero*.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único de la máquina virtual VMware.

vmware.vm.memory.size.consumed[url,uuid]

 La cantidad de memoria física del equipo consumida para respaldar las páginas de memoria física del invitado (KB).
 Valor de retorno: *Entero*.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único de la máquina virtual VMware.

vmware.vm.memory.size.private[url,uuid]

 El tamaño de la memoria privada de la máquina virtual de VMware (bytes).
 Valor de retorno: *Entero*.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único de la máquina virtual de VMware.

vmware.vm.memory.size.shared[url,uuid]

 El tamaño de la memoria compartida de la máquina virtual de VMware (bytes).
 Valor de retorno: *Entero*.

Parámetros:

- **url** - la URL del servicio de VMware;
- **uuid** - el identificador global único de la máquina virtual de VMware.

vmware.vm.memory.size.swapped[url,uuid]

 Tamaño de la memoria intercambiada de la máquina virtual VMware (bytes).
 Valor de retorno: *Entero*.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único de la máquina virtual VMware.

vmware.vm.memory.size.usage.guest[url,uuid]

 El uso de memoria de invitado de la máquina virtual de VMware (bytes).
 Valor devuelto: *Entero*.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único de la máquina virtual de VMware.

vmware.vm.memory.size.usage.host[url,uuid]

 El uso de memoria del equipo de la máquina virtual de VMware (bytes).
 Valor de retorno: *Entero*.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único de la máquina virtual de VMware.

vmware.vm.memory.usage[url,uuid]

 El porcentaje de memoria física del equipo que se ha consumido.
 Valor de retorno: *Float*.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único de la máquina virtual VMware.

vmware.vm.net.if.discovery[url,uuid]

 El descubrimiento de interfaces de red de máquinas virtuales VMware.
 Valor de retorno: objeto JSON.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único de la máquina virtual VMware.

vmware.vm.net.if.in[url,uuid,instance,<mode>]

 Estadísticas de entrada de la interfaz de red de la máquina virtual VMware (bytes/paquetes por segundo).
 Valor de retorno: *Entero* [2].

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único de la máquina virtual VMware;
- **instance** - la instancia de la interfaz de red;
- **mode** - *bps* (por defecto) o *pps* - bytes o paquetes por segundo.

vmware.vm.net.if.out[url,uuid,instance,<mode>]

 Las estadísticas de salida de la interfaz de red de la máquina virtual VMware (bytes/paquetes por segundo).
 Valor de retorno: *Entero* [2].

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único de la máquina virtual VMware;
- **instance** - la instancia de la interfaz de red;
- **mode** - *bps* (por defecto) o *pps* - bytes o paquetes por segundo.

vmware.vm.net.if.usage[url,uuid,<instance>]

 La utilización de red de la máquina virtual de VMware (tasas combinadas de transmisión y recepción) durante el intervalo (KBps).
 Valor de retorno: *Entero*.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único de la máquina virtual de VMware;
- **instance** - la instancia de la interfaz de red.

vmware.vm.perfcounter[url,uuid,path,<instance>]

 El valor del contador de rendimiento de la máquina virtual VMware.
 Valor de retorno: *Entero* [2].

Parámetros:

- **url** - la URL del servicio VMware;

- **uuid** - el identificador global único de la máquina virtual VMware;
- **path** - la ruta del contador de rendimiento [1];
- **instance** - la instancia del contador de rendimiento. Utilice una instancia vacía para valores agregados (por defecto).

vmware.vm.powerstate[url,uuid]

 El estado de encendido de la máquina virtual de VMware.
 Valor de retorno: *0* - apagada; *1* - encendida; *2* - suspendida.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único de la máquina virtual de VMware.

vmware.vm.property[url,uuid,prop]

 La propiedad de la máquina virtual de VMware.
 Valor de retorno: *Cadena*.

Parámetros:

- **url**: la URL del servicio VMware;
- **uuid**: el identificador global único de la máquina virtual de VMware;
- **prop**: la ruta de la propiedad.

vmware.vm.snapshot.get[url,uuid]

 El estado de instantánea de la máquina virtual de VMware.
 Valor devuelto: objeto JSON.

Parámetros:

- **url** - la URL del servicio de VMware;
- **uuid** - el identificador único global de la máquina virtual de VMware.

vmware.vm.state[url,uuid]

 El estado de la máquina virtual de VMware.
 Valor devuelto: *Cadena*: *notRunning*, *resetting*, *running*, *shuttingDown*, *standby* o *unknown*.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único de la máquina virtual de VMware.

vmware.vm.storage.committed[url,uuid]

 El espacio de almacenamiento comprometido de la máquina virtual de VMware (bytes).
 Valor devuelto: *Entero*.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único de la máquina virtual de VMware.

vmware.vm.storage.readoio[url,uuid,instance]

 El número promedio de solicitudes de lectura pendientes al disco virtual durante el intervalo de recopilación.
 Valor de retorno: *Entero*.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único de la máquina virtual VMware;
- **instance** - la instancia del dispositivo de disco.

vmware.vm.storage.totalreadlatency[url,uuid,instance]

 El tiempo promedio que tarda una lectura desde el disco virtual (milisegundos).
 Valor de retorno: *Entero*.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único de la máquina virtual VMware;
- **instance** - la instancia del dispositivo de disco.

vmware.vm.storage.totalwritelatency[url,uuid,instance]

 El tiempo promedio que tarda una escritura en el disco virtual (milisegundos).
 Valor de retorno: *Entero*.

Parámetros:

- **url** - la URL del servicio VMware;

- **uuid** - el identificador global único de la máquina virtual VMware;
- **instance** - la instancia del dispositivo de disco.

vmware.vm.storage.uncommitted[url,uuid]

 El espacio de almacenamiento no comprometido de la máquina virtual VMware (bytes).
 Valor de retorno: *Entero*.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único de la máquina virtual VMware.

vmware.vm.storage.unshared[url,uuid]

 El espacio de almacenamiento no compartido de la máquina virtual de VMware (bytes).
 Valor devuelto: *Entero*.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único de la máquina virtual de VMware.

vmware.vm.storage.writeoio[url,uuid,instance]

 El número promedio de solicitudes de escritura pendientes en el disco virtual durante el intervalo de recopilación.
 Valor de retorno: *Entero*.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único de la máquina virtual VMware;
- **instance** - la instancia del dispositivo de disco.

vmware.vm.tags.get[url,uuid]

 El array de etiquetas de la máquina virtual de VMware.
 Valor devuelto: JSON object.

Parámetros:

- **url** - la URL del servicio de VMware;
- **uuid** - el identificador único global de la máquina virtual de VMware.

Este item funciona con vSphere 6.5 y versiones posteriores.

vmware.vm.tools[url,uuid,<mode>]

 El estado de las herramientas de invitado de la máquina virtual de VMware.
 Valor de retorno: *Cadena*.
 En modo *status*: *guestToolsExecutingScripts* - VMware Tools se está iniciando; *guestToolsNotRunning* - VMware Tools no se está ejecutando; *guestToolsRunning* - VMware Tools se está ejecutando.
 En modo *version*: versión.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único de la máquina virtual de VMware;
- **mode** - *version*, *status* (por defecto).

vmware.vm.uptime[url,uuid]

 El tiempo de actividad de la máquina virtual VMware (segundos).
 Valor devuelto: *Entero*.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único de la máquina virtual VMware.

vmware.vm.vfs.dev.discovery[url,uuid]

 El descubrimiento de dispositivos de disco de máquinas virtuales VMware.
 Valor de retorno: objeto JSON.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único de la máquina virtual VMware.

vmware.vm.vfs.dev.read[url,uuid,instance,<mode>]

 Las estadísticas de lectura del dispositivo de disco de la máquina virtual de VMware (bytes/operaciones por segundo).
 Valor de retorno: *Entero* [2].

Parámetros:

- **url**: la URL del servicio VMware;
- **uuid**: el identificador global único de la máquina virtual de VMware;
- **instance**: la instancia del dispositivo de disco;
- **mode**: *bps* (por defecto) o *ops* - bytes u operaciones por segundo.

vmware.vm.vfs.dev.write[url,uuid,instance,<mode>]

 Las estadísticas de escritura del dispositivo de disco de la máquina virtual de VMware (bytes/operaciones por segundo).
 Valor de retorno: *Entero* [2].

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único de la máquina virtual de VMware;
- **instance** - la instancia del dispositivo de disco;
- **mode** - *bps* (por defecto) o *ops* - bytes u operaciones por segundo.

vmware.vm.vfs.fs.discovery[url,uuid]

 El descubrimiento de sistemas de archivos de máquinas virtuales VMware.
 Valor de retorno: objeto JSON.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único de la máquina virtual VMware.

VMware Tools debe estar instalado en la máquina virtual invitada para que esta métrica funcione.

vmware.vm.vfs.fs.size[url,uuid,fsname,<mode>]

 Las estadísticas del sistema de archivos de la máquina virtual VMware (bytes/porcentajes).
 Valor de retorno: *Entero* - para bytes; *Flotante* - para porcentaje.

Parámetros:

- **url** - la URL del servicio VMware;
- **uuid** - el identificador global único de la máquina virtual VMware;
- **fsname** - el nombre del sistema de archivos;
- **mode** - *total*, *free*, *used*, *pfree* o *pused*.

VMware Tools debe estar instalado en la máquina virtual invitada para que esta métrica funcione.

Notas al pie

[1] Consulte Creación de nombres personalizados de contadores de rendimiento para VMware.

[2] El valor de estos items se obtiene de los contadores de rendimiento de VMware y el parámetro VMwarePerfFrequency se utiliza para actualizar sus datos en la caché de VMware de Zabbix:

- vmware.cl.perfcounter
- vmware.hv.datastore.read
- vmware.hv.datastore.write
- vmware.hv.network.in
- vmware.hv.network.out
- vmware.hv.perfcounter
- vmware.vm.cpu.ready
- vmware.vm.net.if.in
- vmware.vm.net.if.out
- vmware.vm.perfcounter
- vmware.vm.vfs.dev.read
- vmware.vm.vfs.dev.write

Más información

Consulte Supervisión de máquinas virtuales para obtener información detallada sobre cómo configurar Zabbix para supervisar entornos VMware.

Campos de clave de descubrimiento de máquinas virtuales

La siguiente tabla enumera los campos devueltos por las claves de descubrimiento relacionadas con máquinas virtuales.

Item key		
Description	**Field**	**Retrieved content**
vmware.cluster.discovery		
Realiza el descubrimiento de clústeres.	{#CLUSTER.ID}	Identificador del clúster.
	{#CLUSTER.NAME}	Nombre del clúster.

	”resource_pool”	Un array que contiene datos del resource pool, incluido el ID del grupo de recursos, el array de etiquetas, la ruta del resource pool y el número de máquinas virtuales.
		
		Estructura del array:
		`[{`
		`"rpid":"resource group id",`
		`"tags":[{}],`
		`"rpath":"resource group path",`
		`"vm_count":0`
		`}]`

	”tags”	Para la estructura del array `"tags"`, consulte el campo ”tags”.
		Un array que contiene etiquetas con nombre de etiqueta, descripción y categoría.
		
		Estructura del array:
		`[{`
		`"tag":"tag name",`
		`"tag_description":"tag description",`
		`"category":"tag category"`
		`}]`

vmware.datastore.discovery		
Realiza el descubrimiento de datastores.	{#DATASTORE}	Nombre del datastore.

	{#DATASTORE.EXTENT}	Un array que contiene el ID de partición del extent del datastore y el nombre de la instancia.
		
		Estructura del array:
		`[{`
		`"partitionid":1,`
		`"instance":"name"`
		`}]`

	{#DATASTORE.TYPE}	Tipo de datastore. Ejemplos de valores: `VMFS`, `NFS`, `vsan`, etc.
	{#DATASTORE.UUID}	Identificador del datastore.

	”tags”	Un array que contiene etiquetas con nombre de etiqueta, descripción y categoría.
		
		Estructura del array:
		`[{`
		`"tag":"tag name",`
		`"tag_description":"tag description",`
		`"category":"tag category"`
		`}]`

vmware.dc.discovery		
Realiza el descubrimiento de centros de datos.	{#DATACENTER}	Nombre del centro de datos.
	{#DATACENTERID}	Identificador del centro de datos.

Item key		

	”tags”	Un array que contiene etiquetas con nombre de etiqueta, descripción y categoría.
		
		Estructura del array:
		`[{`
		`"tag":"tag name",`
		`"tag_description":"tag description",`
		`"category":"tag category"`
		`}]`

vmware.dvswitch.discovery		
Realiza el descubrimiento de switches distribuidos de vSphere.	{#DVS.NAME}	Nombre del switch.
	{#DVS.UUID}	Identificador del switch.
vmware.hv.discovery		
Realiza el descubrimiento de hipervisores.	{#HV.UUID}	Identificador único del hipervisor.
	{#HV.ID}	Identificador del hipervisor (nombre del objeto administrado HostSystem).
	{#HV.NAME}	Nombre del hipervisor.
	{#HV.NETNAME}	Nombre de host de red del hipervisor.
	{#HV.IP}	Dirección IP del hipervisor, puede estar vacía. En caso de una configuración HA con varias interfaces de red, se observa la siguiente prioridad de selección para la interfaz: - se prefiere la IP que comparte la subred IP con la IP de vCenter; - se prefiere la IP de la subred IP con puerta de enlace predeterminada; - se prefiere la IP de la interfaz con el ID más bajo.
	{#CLUSTER.NAME}	Nombre del clúster, puede estar vacío.
	{#DATACENTER.NAME}	Nombre del centro de datos.
	{#PARENT.NAME}	Nombre del contenedor que almacena el hipervisor.
	{#PARENT.TYPE}	Tipo de contenedor en el que se almacena el hipervisor. Los valores pueden ser `Datacenter`, `Folder`, `ClusterComputeResource`, `VMware`, donde ”VMware” representa un tipo de contenedor desconocido.

	”resource_pool”	Un array que contiene datos del resource pool, incluido el ID del grupo de recursos, el array de etiquetas, la ruta del resource pool y el número de máquinas virtuales.
		
		Estructura del array:
		`[{`
		`"rpid":"resource group id",`
		`"tags":[{}],`
		`"rpath":"resource group path",`
		`"vm_count":0`
		`}]`
		
		Para la estructura del array `"tags"`, consulte el campo ”tags”.

	”tags”	Un array que contiene etiquetas con nombre de etiqueta, descripción y categoría.
		
		Estructura del array:
		`[{`
		`"tag":"tag name",`
		`"tag_description":"tag description",`
		`"category":"tag category"`
		`}]`

vmware.hv.datastore.discovery		

Item key		
Realiza el descubrimiento de datastores del hipervisor. Tenga en cuenta que varios hipervisores pueden usar el mismo datastore.	{#DATASTORE}	Nombre del datastore.
	{#DATASTORE.TYPE}	Tipo de datastore. Ejemplos de valores: `VMFS`, `NFS`, `vsan`, etc.
	{#DATASTORE.UUID}	Identificador del datastore.
	{#MULTIPATH.COUNT}	Número registrado de rutas del datastore.
	{#MULTIPATH.PARTITION.COUNT}	Número de particiones de disco disponibles.

	"datastore_extent"	Un array que contiene el nombre de la instancia del extent del datastore y el ID de partición.
		
		Estructura del array:
		`[{`
		`"partitionid":1,`
		`"instance":"name"`
		`}]`

	"tags"	Un array que contiene etiquetas con nombre de etiqueta, descripción y categoría.
		
		Estructura del array:
		`[{`
		`"tag":"tag name",`
		`"tag_description":"tag description",`
		`"category":"tag category"`
		`}]`

vmware.hv.net.if.discovery		
Realiza el descubrimiento de interfaces de red del hipervisor.	{#IFNAME}	Nombre de la interfaz.
	{#IFDRIVER}	Controlador de la interfaz.
	{#IFDUPLEX}	Configuración dúplex de la interfaz.
	{#IFSPEED}	Velocidad de la interfaz.
	{#IFMAC}	Dirección MAC de la interfaz.
vmware.vm.discovery		
Realiza el descubrimiento de máquinas virtuales.	{#VM.UUID}	Identificador único de la máquina virtual.
	{#VM.ID}	Identificador de la máquina virtual (nombre del objeto administrado VirtualMachine).
	{#VM.NAME}	Nombre de la máquina virtual.
	{#HV.NAME}	Nombre del hipervisor.
	{#HV.UUID}	Identificador único del hipervisor.
	{#HV.ID}	Identificador del hipervisor (nombre del objeto administrado HostSystem).
	{#CLUSTER.NAME}	Nombre del clúster, puede estar vacío.
	{#DATACENTER.NAME}	Nombre del centro de datos.
	{#DATASTORE.NAME}	Nombre del datastore.
	{#DATASTORE.UUID}	Identificador del datastore.
	{#VM.IP}	Dirección IP de la máquina virtual, puede estar vacía.
	{#VM.DNS}	Nombre DNS de la máquina virtual, puede estar vacío.
	{#VM.GUESTFAMILY}	Familia del sistema operativo invitado de la máquina virtual, puede estar vacía.
	{#VM.GUESTFULLNAME}	Nombre completo del sistema operativo invitado de la máquina virtual, puede estar vacío.

Item key		
	{#VM.FOLDER}	La cadena de carpetas padre de la máquina virtual, puede usarse como valor para grupos anidados; los nombres de las carpetas se combinan con "/". Puede estar vacío.
	{#VM.TOOLS.STATUS}	Estado de VMware Tools de la máquina virtual.
	{#VM.POWERSTATE}	Estado de energía de la máquina virtual de VMware (`poweredOff`, `poweredOn` o `suspended`).
	{#VM.RPOOL.ID}	Identificador del resource pool.
	{#VM.RPOOL.PATH}	Ruta completa del resource pool excluyendo el nombre raíz "Resources". Los nombres de las carpetas se combinan con "/".
	{#VM.SNAPSHOT.COUNT}	Número de instantáneas de la VM.

	"tags"	Un array que contiene etiquetas con nombre de etiqueta, descripción y categoría.
		
		Estructura del array:
		`[{`
		`"tag":"tag name",`
		`"tag_description":"tag description",`
		`"category":"tag category"`
		`}]`

	"vm_customattribute"	Un array de atributos personalizados de la máquina virtual (si están definidos).
		
		Estructura del array:
		`[{`
		`"name":"custom field name",`
		`"value":"custom field value"`
		`}]`

	"net_if"	Un array de interfaces de red de la máquina virtual.
		
		Estructura del array:
		`[{`
		`"ifname": "interface name",`
		`"ifdesc": "interface description",`
		`"ifmac": "00:00:00:00:00:00",`
		`"ifconnected": true,`
		`"iftype": "interface type",`
		`"ifbackingdevice": "interface backing device",`
		`"ifdvswitch_uuid": "interface switch uuid",`
		`"ifdvswitch_portgroup": "interface switch port group",`
		`"ifdvswitch_port": "interface switch port",`
		`"ifip": ["interface ip addresses"]`
		`}]`
		
		Para la descripción de los datos devueltos, consulte la clave de item "vmware.vm.net.if.discovery".

vmware.vm.net.if.discovery		
Realiza el descubrimiento de interfaces de red de la máquina virtual.	{#IFNAME}	Nombre de la interfaz de red.
	{#IFDESC}	Descripción de la interfaz.
	{#IFMAC}	Dirección MAC de la interfaz.
	{#IFCONNECTED}	Estado de conexión de la interfaz (`false` - desconectada; `true` - conectada).
	{#IFTYPE}	Tipo de interfaz.
	{#IFBACKINGDEVICE}	Nombre del dispositivo de respaldo.
	{#IFDVSWITCH.UUID}	Identificador único de vSphere Distributed Switch.
	{#IFDVSWITCH.PORTGROUP}	Grupo de puertos distribuido.
	{#IFDVSWITCH.PORT}	Puerto de vSphere Distributed Switch.
	"ifip"	Un array de direcciones de la interfaz.

Item key		
vmware.vm.vfs.dev.discovery		
Realiza el descubrimiento de dispositivos de disco de la máquina virtual.	{#DISKNAME}	Nombre del dispositivo de disco.
vmware.vm.vfs.fs.discovery		
Realiza el descubrimiento del sistema de archivos de la máquina virtual.	{#FSNAME}	Nombre del sistema de archivos.

Ejemplos JSON para items de VMware

Descripción general Esta sección proporciona información adicional sobre los objetos JSON devueltos por varios items de VMware.

vmware.*.alarms.get Los items **vmware.alarms.get[]**, **vmware.cluster.alarms.get[]**, **vmware.datastore.alarms.get[]**, **vmware.dc.alarms.get[]**, **vmware.hv.alarms.get[]**, **vmware.vm.alarms.get[]** devuelven objetos JSON con la siguiente estructura (los valores se proporcionan como ejemplo):

```
{
    "alarms": [
        {
            "name": "Host connection and power state",
            "system_name": "alarm.HostConnectionStateAlarm",
            "description": "Default alarm to monitor host connection and power state",
            "enabled": true,
            "key": "alarm-1.host-2013",
            "time": "2022-06-27T05:27:38.759976Z",
            "overall_status": "red",
            "acknowledged": false,
            "entity": {
                "id": "host-2013",
                "uuid": "4201b707-f737-7cf8-881f-49215d1aad2a",
                "type": "HostSystem",
                "name": "esx-riga-01.prod.local"
            }
        },
        {
            "name": "Host memory usage",
            "system_name": "alarm.HostMemoryUsageAlarm",
            "description": "Default alarm to monitor host memory usage",
            "enabled": true,
            "key": "alarm-4.host-1004",
            "time": "2022-05-16T13:32:42.47863Z",
            "overall_status": "yellow",
            "acknowledged": false,
            "entity": {
                "id": "host-1004",
                "uuid": "5315c818-g848-8dg9-992g-50326e2bbe3a",
                "type": "HostSystem",
                "name": "esx-riga-02.prod.local",
            }
        },
        {
            // other alarms
        }
```

```
    ]
}
```

vmware.*.tags.get Las métricas **vmware.cluster.tags.get[]**, **vmware.datastore.tags.get[]**, **vmware.dc.tags.get[]**, **vmware.hv.tags.get[]**, **vmware.vm.tags.get[]** devuelven objetos JSON con la siguiente estructura (los valores se proporcionan como ejemplo):

```
{
    "tags": [
        {
            "name": "Windows",
        "description": "tag for cat OS type",
        "category": "OS type"
    },
    {
            "name": "SQL Server",
        "description": "tag for cat application name",
        "category": "application name"
        },
    {
            // otras etiquetas
        }
    ]
}
```

vmware.hv.diskinfo.get La métrica **vmware.hv.diskinfo.get[]** devuelve objetos JSON con la siguiente estructura (los valores se proporcionan como ejemplo):

```
[
  {
    "instance": "mpx.vmhba32:C0:T0:L0",
    "hv_uuid": "8002299e-d7b9-8728-d224-76004bbb6100",
    "datastore_uuid": "",
    "operational_state": [
      "ok"
    ],
    "lun_type": "disk",
    "queue_depth": 1,
    "model": "USB DISK",
    "vendor": "SMI Corp",
    "revision": "1100",
    "serial_number": "CCYYMMDDHHmmSS9S62CK",
    "vsan": {}
  },
  {
    // otras instancias
  }
]
```

vmware.dvswitch.fetchports.get La métrica **vmware.dvswitch.fetchports.get[]** devuelve objetos JSON con la siguiente estructura (los valores se proporcionan como ejemplo):

```
{
    "FetchDVPortsResponse":
    {
        "returnval": [
            {
                "key": "0",
                "dvsUuid": "50 36 6a 24 25 c0 10 9e-05 4a f6 ea 4e 3d 09 88",
                "portgroupKey": "dvportgroup-2023",
                "proxyHost":
                {
                    "@type": "HostSystem",
```

```
                    "#text": "host-2021"
                },
                "connectee":
                {
                    "connectedEntity":
                    {
                        "@type": "HostSystem",
                        "#text": "host-2021"
                    },
                    "nicKey": "vmnic0",
                    "type": "pnic"
                },
                "conflict": "false",
                "state":
                {
                    "runtimeInfo":
                    {
                        "linkUp": "true",
                        "blocked": "false",
                        "vlanIds":
                        {
                            "start": "0",
                            "end": "4094"
                        },
                        "trunkingMode": "true",
                        "linkPeer": "vmnic0",
                        "macAddress": "00:00:00:00:00:00",
                        "statusDetail": null,
                        "vmDirectPathGen2Active": "false",
                        "vmDirectPathGen2InactiveReasonOther": "portNptIncompatibleConnectee"
                    },
                    "stats":
                    {
                        "packetsInMulticast": "2385470",
                        "packetsOutMulticast": "45",
                        "bytesInMulticast": "309250248",
                        "bytesOutMulticast": "5890",
                        "packetsInUnicast": "155601537",
                        "packetsOutUnicast": "113008658",
                        "bytesInUnicast": "121609489384",
                        "bytesOutUnicast": "47240279759",
                        "packetsInBroadcast": "1040420",
                        "packetsOutBroadcast": "7051",
                        "bytesInBroadcast": "77339771",
                        "bytesOutBroadcast": "430392",
                        "packetsInDropped": "0",
                        "packetsOutDropped": "0",
                        "packetsInException": "0",
                        "packetsOutException": "0"
                    }
                },
                "connectionCookie": "1702765133",
                "lastStatusChange": "2022-03-25T14:01:11Z",
                "hostLocalPort": "false"
            },
            {
                //otras claves
            }
        ]
    }
}
```

vmware.hv.hw.sensors.get La métrica **vmware.hv.hw.sensors.get[]** devuelve objetos JSON con la siguiente estructura (los valores se proporcionan como ejemplo):


```
{
    "val":
        {
            "@type": "HostHardwareStatusInfo",
            "storageStatusInfo": [
                {
                    "name": "Intel Corporation HD Graphics 630 #2",
                "status":
                        {
                            "label": "Unknown",
                    "summary": "Cannot report on the current status of the physical element",
                    "key": "Unknown"
                }
            },
            {
                    "name": "Intel Corporation 200 Series/Z370 Chipset Family USB 3.0 xHCI Controller #20"
                "status":
                        {
                     "label": "Unknown",
                     "summary": "Cannot report on the current status of the physical element",
                     "key": "Unknown"
                 }
            },
            {
                    // otros sensores hv hw
                }
        ]
        }
}
```


vmware.hv.sensors.get La métrica **vmware.hv.sensors.get[]** devuelve objetos JSON con la siguiente estructura (los valores se proporcionan como ejemplo):


```
{
    "val":
        {
            "@type": "ArrayOfHostNumericSensorInfo", "HostNumericSensorInfo": [
                {
                    "@type": "HostNumericSensorInfo",
                    "name": "System Board 1 PwrMeter Output --- Normal",
                    "healthState":
                        {
                            "label": "Green",
                            "summary": "Sensor is operating under normal conditions",
                    "key": "green"
                },
                    "currentReading": "10500",
                "unitModifier": "-2",
                "baseUnits": "Watts",
                "sensorType": "other"
            },
                {
                    "@type": "HostNumericSensorInfo",
                    "name": "Power Supply 1 PS 1 Output --- Normal",
                    "healthState":
                        {
                            "label": "Green",
                    "summary": "Sensor is operating under normal conditions",
                    "key": "green"
                },
                    "currentReading": "10000",
```

```
            "unitModifier": "-2",
            "baseUnits": "Watts",
            "sensorType": "power"
        },
            {
                    // otros sensores hv
                }
        ]
        }
}
```

vmware.vm.snapshot.get Si existen instantáneas, la métrica **vmware.snapshot.get[]** devuelve un objeto JSON con la siguiente estructura (los valores se proporcionan como ejemplo):

```
{
  "snapshot": [
    {
      "name": "VM Snapshot 4%2f1%2f2022, 9:16:39 AM",
      "description": "Descr 1",
      "createtime": "2022-04-01T06:16:51.761Z",
      "size": 5755795171,
      "uniquesize": 5755795171
    },
    {
      "name": "VM Snapshot 4%2f1%2f2022, 9:18:21 AM",
      "description": "Descr 2",
      "createtime": "2022-04-01T06:18:29.164999Z",
      "size": 118650595,
      "uniquesize": 118650595
    },
    {
      "name": "VM Snapshot 4%2f1%2f2022, 9:37:29 AM",
      "description": "Descr 3",
      "createtime": "2022-04-01T06:37:53.534999Z",
      "size": 62935016,
      "uniquesize": 62935016
    }
  ],
  "count": 3,
  "latestdate": "2022-04-01T06:37:53.534999Z",
  "latestage": 22729203,
  "oldestdate": "2022-04-01T06:16:51.761Z",
  "oldestage": 22730465,
  "size": 5937380782,
  "uniquesize": 5937380782
}
```

Si no existe ninguna instantánea, la métrica **vmware.snapshot.get[]** devuelve un objeto JSON con valores vacíos:

```
{
  "snapshot": [],
  "count": 0,
  "latestdate": null,
  "latestage": 0,
  "oldestdate": null,
  "oldestage": 0,
  "size": 0,
  "uniquesize": 0
}
```

Ejemplo de configuración de monitorización de VMware

Descripción general

El siguiente ejemplo describe cómo configurar Zabbix para monitorizar máquinas virtuales VMware. Esto implica:

- crear un equipo que represente su entorno VMware;
- crear una regla de descubrimiento de bajo nivel que descubra máquinas virtuales en su entorno VMware;
- crear un prototipo de equipo, en base al cual Zabbix generará equipos reales para las máquinas virtuales descubiertas por la regla de descubrimiento de bajo nivel.

Requisitos previos

> **Note:**
> Este ejemplo no cubre la configuración de VMware. Se supone que VMware ya está configurado.

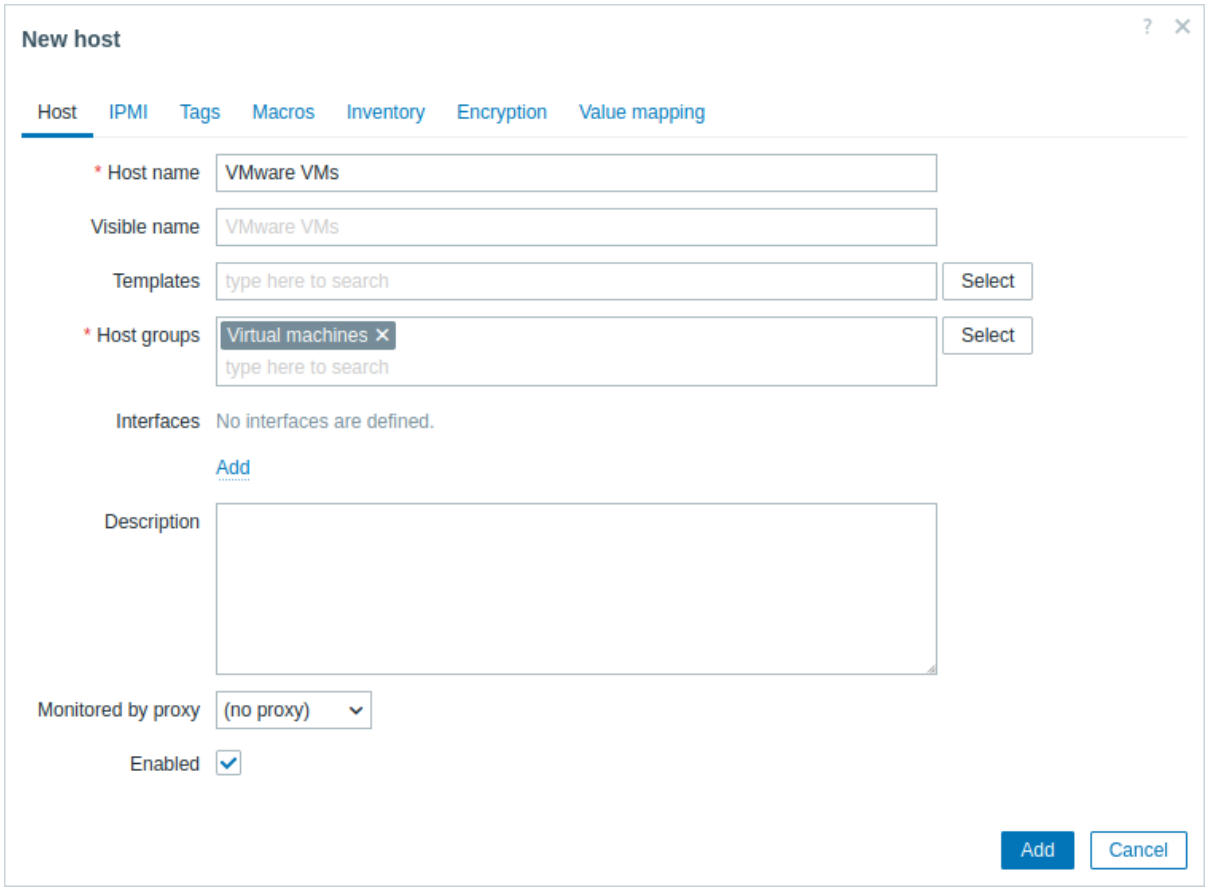
Antes de continuar, establezca el parámetro `StartVMwareCollectors` en el archivo de configuración de Zabbix server en 2 o más (el valor predeterminado es 0).

Crear un host

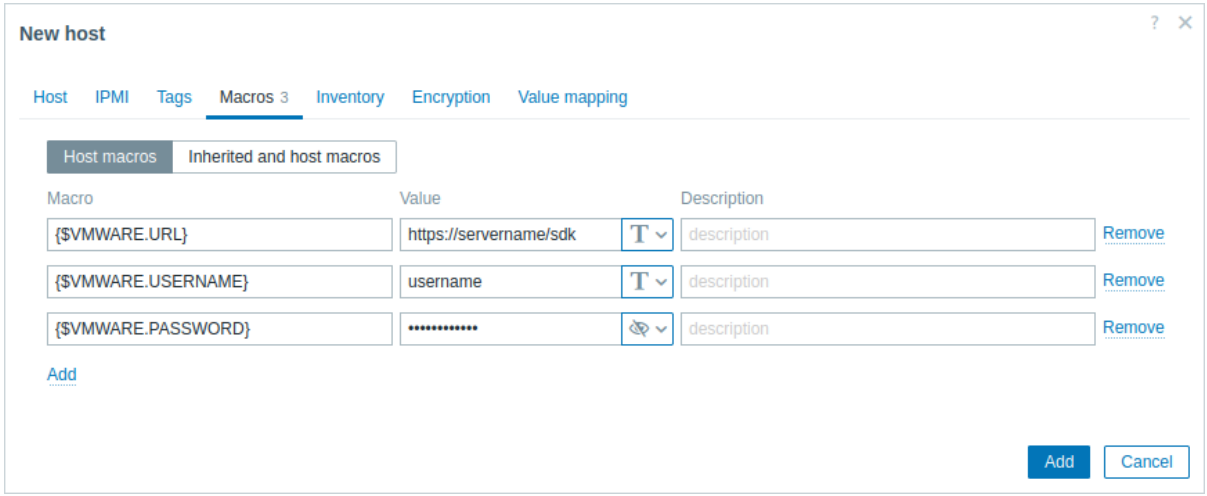
1. Vaya a *Data collection* > *Hosts*.

2. Cree un host:

- En el campo *Host name*, introduzca un nombre de host (por ejemplo, "VMware VMs").
- En el campo *Host groups*, escriba o seleccione un grupo de hosts (por ejemplo, "Virtual machines").



- En la pestaña *Macros*, establezca las siguientes macros del host:
  - {$VMWARE.URL} - URL del SDK del servicio VMware (hipervisor ESXi) (https://servername/sdk)
  - {$VMWARE.USERNAME} - nombre de usuario del servicio VMware
  - {$VMWARE.PASSWORD} - contraseña del usuario {$VMWARE.USERNAME} del servicio VMware



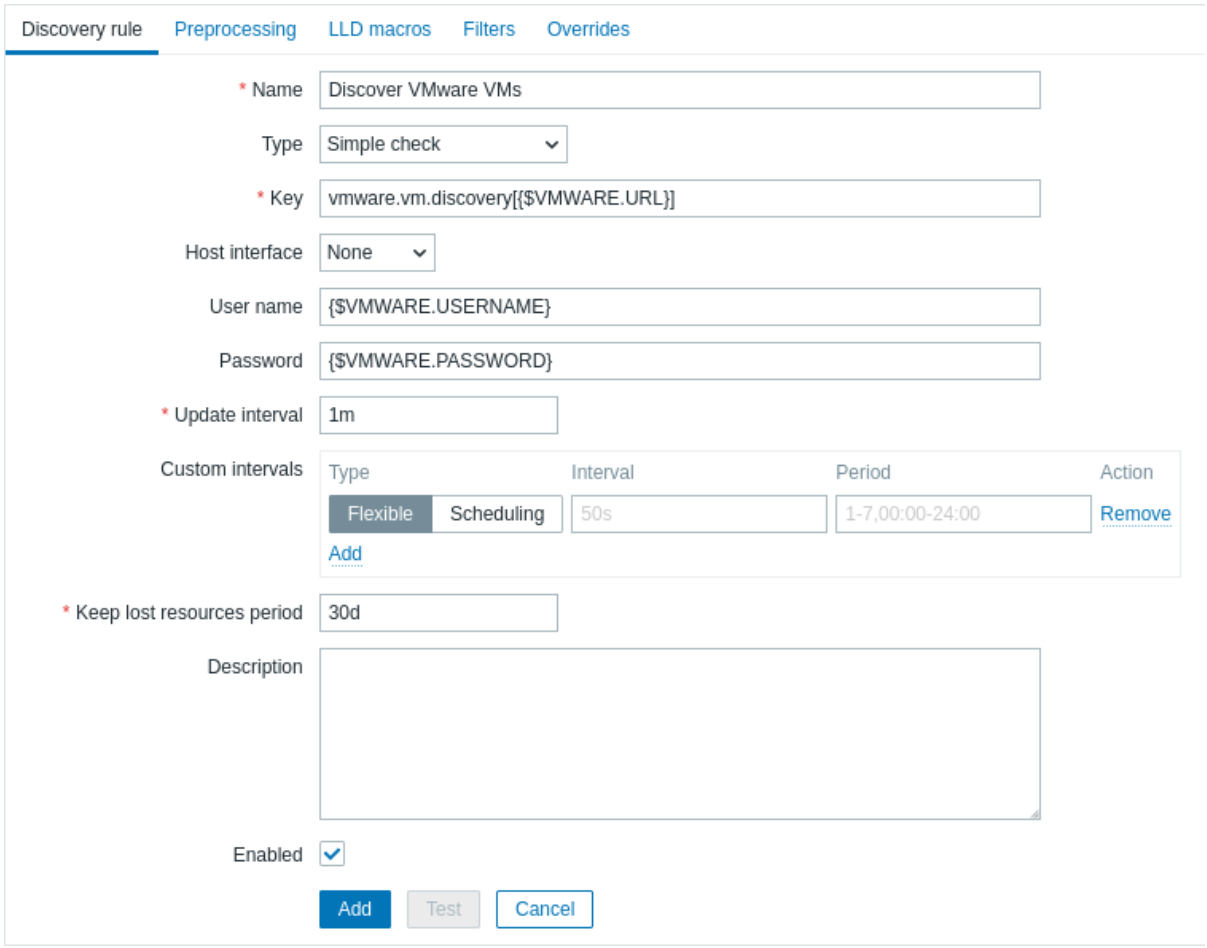
3. Haga clic en el botón *Add* para crear el host. Este host representará su entorno VMware.

Crear una regla de descubrimiento de bajo nivel

1. Haga clic en *Discovery* para el host creado y vaya a la lista de reglas de descubrimiento de bajo nivel de ese host.

2. Cree una regla de descubrimiento de bajo nivel:

- En el campo *Name*, introduzca un nombre para la regla de descubrimiento de bajo nivel (por ejemplo, "Discover VMware VMs").
- En el campo *Type*, seleccione "Simple check".
- En el campo *Key*, introduzca la clave de item integrada para descubrir máquinas virtuales de VMware: `vmware.vm.discovery[{$VMWA`
- En los campos *User name* y *Password*, introduzca las macros correspondientes configuradas previamente en el host.



3. Haga clic en el botón *Add* para crear la regla de descubrimiento de bajo nivel. Esta regla de descubrimiento detectará las máquinas virtuales en su entorno VMware.

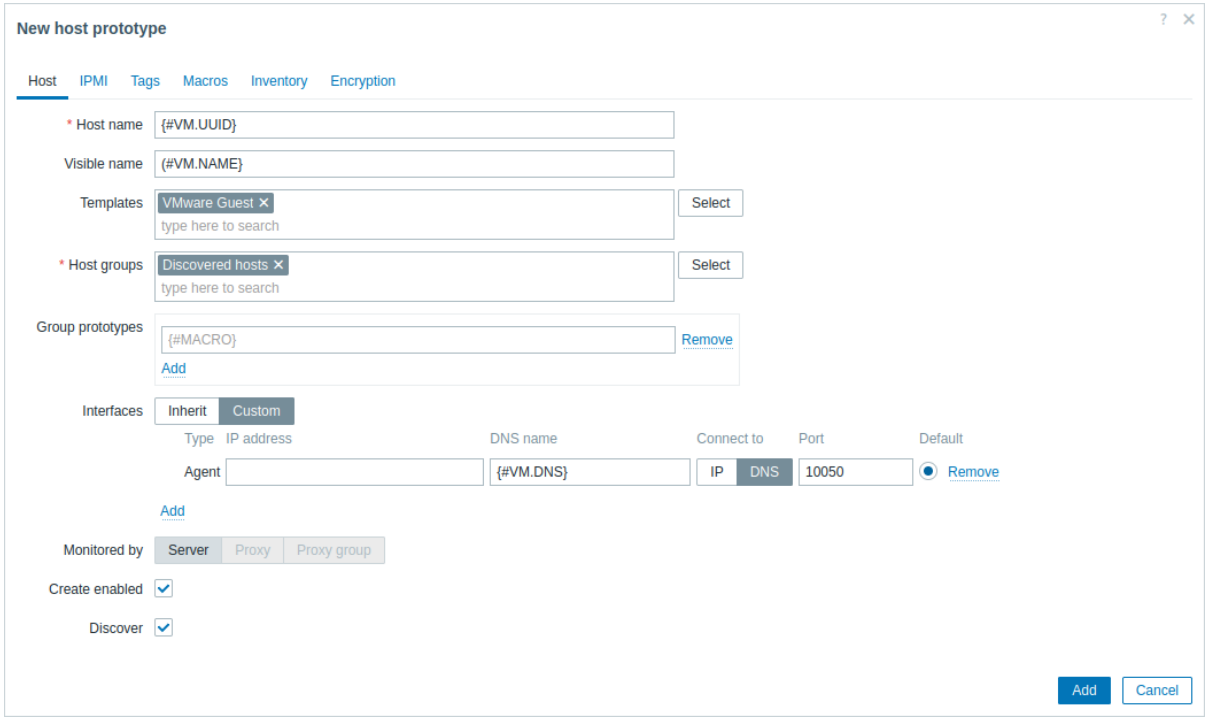
Crear una plantilla de equipo

1. En la lista de reglas de descubrimiento de bajo nivel, haga clic en *Plantillas de equipo* para la regla de descubrimiento de bajo

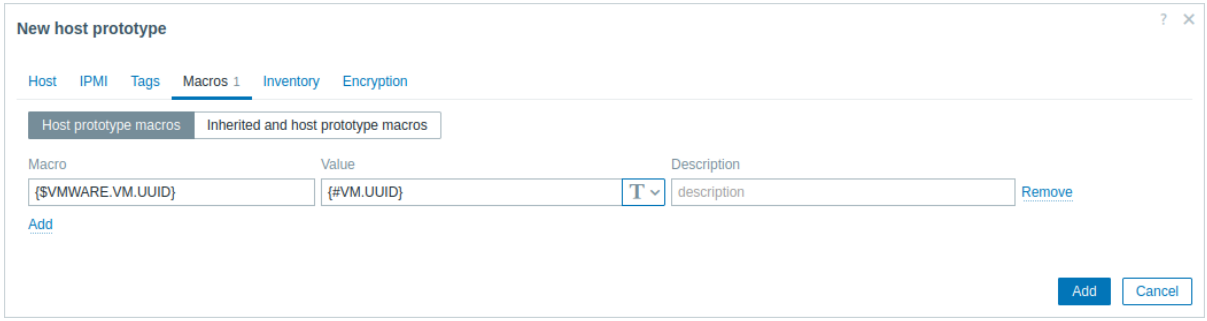
nivel creada previamente.

2. Cree una plantilla de equipo. Dado que las plantillas de equipo son modelos para crear equipos a través de reglas de descubrimiento de bajo nivel, la mayoría de los campos contendrán macros de descubrimiento de bajo nivel. Esto garantiza que los equipos se creen con propiedades basadas en el contenido recuperado por la regla de descubrimiento de bajo nivel creada previamente.

- En el campo *Nombre del equipo*, introduzca la macro {#VM.UUID}.
- En el campo *Nombre visible*, introduzca la macro {#VM.NAME}.
- En el campo *Plantillas*, introduzca o seleccione la plantilla "VMware Guest". Esta plantilla contiene métricas de VMware y reglas de descubrimiento para monitorizar el estado de energía de una máquina virtual, uso de CPU, uso de memoria, dispositivos de red, etc.
- En el campo *Grupos de equipos*, introduzca o seleccione un grupo de equipos (por ejemplo, "Equipos descubiertos").
- En el campo *Interfaces*, añada una interfaz de equipo personalizada. Luego, introduzca la macro {#VM.DNS} en el campo *Nombre DNS*, o introduzca la macro {#VM.IP} en el campo *Dirección IP*. Alternativamente, si sus máquinas virtuales del entorno VMware tienen varias interfaces, continúe con la sección *Configuración avanzada de interfaz de equipo*. Configurar una interfaz de equipo personalizada es necesario para el correcto funcionamiento de la plantilla *VMware Guest*.



- En la pestaña *Macros*, establezca la macro {$VMWARE.VM.UUID} con el valor {#VM.UUID}. Esto es necesario para el correcto funcionamiento de la plantilla *VMware Guest* que utiliza esta macro como macro de usuario a nivel de equipo en los parámetros de la métrica (por ejemplo, *vmware.vm.net.if.discovery[{$VMWARE.URL}, {$VMWARE.VM.UUID}]*).

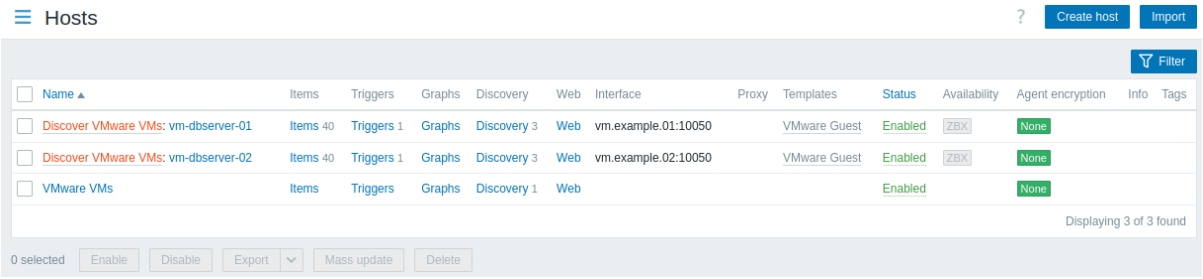


3. Haga clic en el botón *Añadir* para crear la plantilla de equipo. Esta plantilla de equipo se utilizará para crear equipos para las máquinas virtuales descubiertas por la regla de descubrimiento de bajo nivel creada previamente.

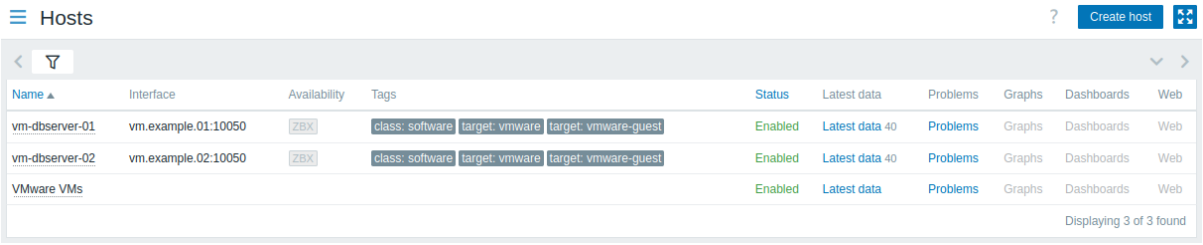
Ver hosts y métricas

Después de crear el host prototype, la regla de descubrimiento de bajo nivel creará hosts para las máquinas virtuales de VMware descubiertas, y Zabbix comenzará a monitorizarlas. Tenga en cuenta que el descubrimiento y la creación de hosts también pueden ejecutarse manualmente, si es necesario.

Para ver los hosts creados, vaya a la sección de menú *Data collection > Hosts*.



Para ver las métricas recopiladas, vaya a la sección de menú *Monitoring > Hosts* y haga clic en *Latest data* para uno de los hosts.



Configuración avanzada de la interfaz de equipo

La clave de métrica `vmware.vm.discovery[{$VMWARE.URL}]`, configurada en la sección *Crear una regla de descubrimiento de bajo nivel*, devuelve datos de interfaces de red en el campo "net_if":

```
"net_if": [
      {
        "ifname": "5000",
        "ifdesc": "Network adapter 1",
        "ifmac": "00:11:22:33:44:55",
        "ifconnected": true,
        "iftype": "VirtualVmxnet3",
        "ifbackingdevice": "VLAN(myLab)",
        "ifdvswitch_uuid": "",
        "ifdvswitch_portgroup": "",
        "ifdvswitch_port": "",
        "ifip": [
          "127.0.0.1",
          "::1"
        ]
      },
      {
        "ifname": "5001",
        "ifdesc": "Network adapter 2",
        "ifmac": "00:11:22:33:44:55",
        "ifconnected": false,
        "iftype": "VirtualVmxnet3",
        "ifbackingdevice": "VLAN(myLab2)",
        "ifdvswitch_uuid": "",
        "ifdvswitch_portgroup": "",
        "ifdvswitch_port": "",
        "ifip": []
      }
    ]
```

Estos datos pueden utilizarse para configurar una interfaz de equipo personalizada.

1. Al crear una regla de descubrimiento de bajo nivel, configure adicionalmente una macro de descubrimiento de bajo nivel. En la pestaña *Macros LLD*, cree una macro LLD personalizada con un valor JSONPath. Por ejemplo:

- `{#MYLAB.NET.IF}` - `$.net_if[?(@.ifbackingdevice=="VLAN(myLab)")].ifip[0].first()`

2. Al crear un prototipo de equipo, añada una interfaz de equipo personalizada e introduzca la macro LLD en el campo *Nombre DNS* o *Dirección IP*.

Supervisión web

Resumen Con Zabbix puede comprobar varios aspectos de disponibilidad de sitios web.

> **Attention:**
> Para realizar la supervisión web, el server de Zabbix debe estar inicialmente configurado con soporte para cURL (libcurl).

Para activar la supervisión web, debe definir escenarios web. Un escenario web consiste en una o varias solicitudes HTTP o "pasos". Los pasos se ejecutan periódicamente por el server de Zabbix en un orden predefinido. Si un host es supervisado por proxy, los pasos se ejecutan por el proxy.

Los escenarios web se asocian a hosts/templates de la misma manera que los items, triggers, etc. Esto significa que los escenarios web también pueden crearse a nivel de template y luego aplicarse a varios hosts de una sola vez.

La siguiente información se recopila en cualquier escenario web:

- velocidad media de descarga por segundo para todos los pasos de todo el escenario
- número del paso que falló
- último mensaje de error

La siguiente información se recopila en cualquier paso de un escenario web:

- velocidad de descarga por segundo
- tiempo de respuesta
- código de respuesta

Para más detalles, consulte items de supervisión web.

Los datos recopilados al ejecutar escenarios web se almacenan en la base de datos. Los datos se usan automáticamente para gráficos, triggers y notificaciones.

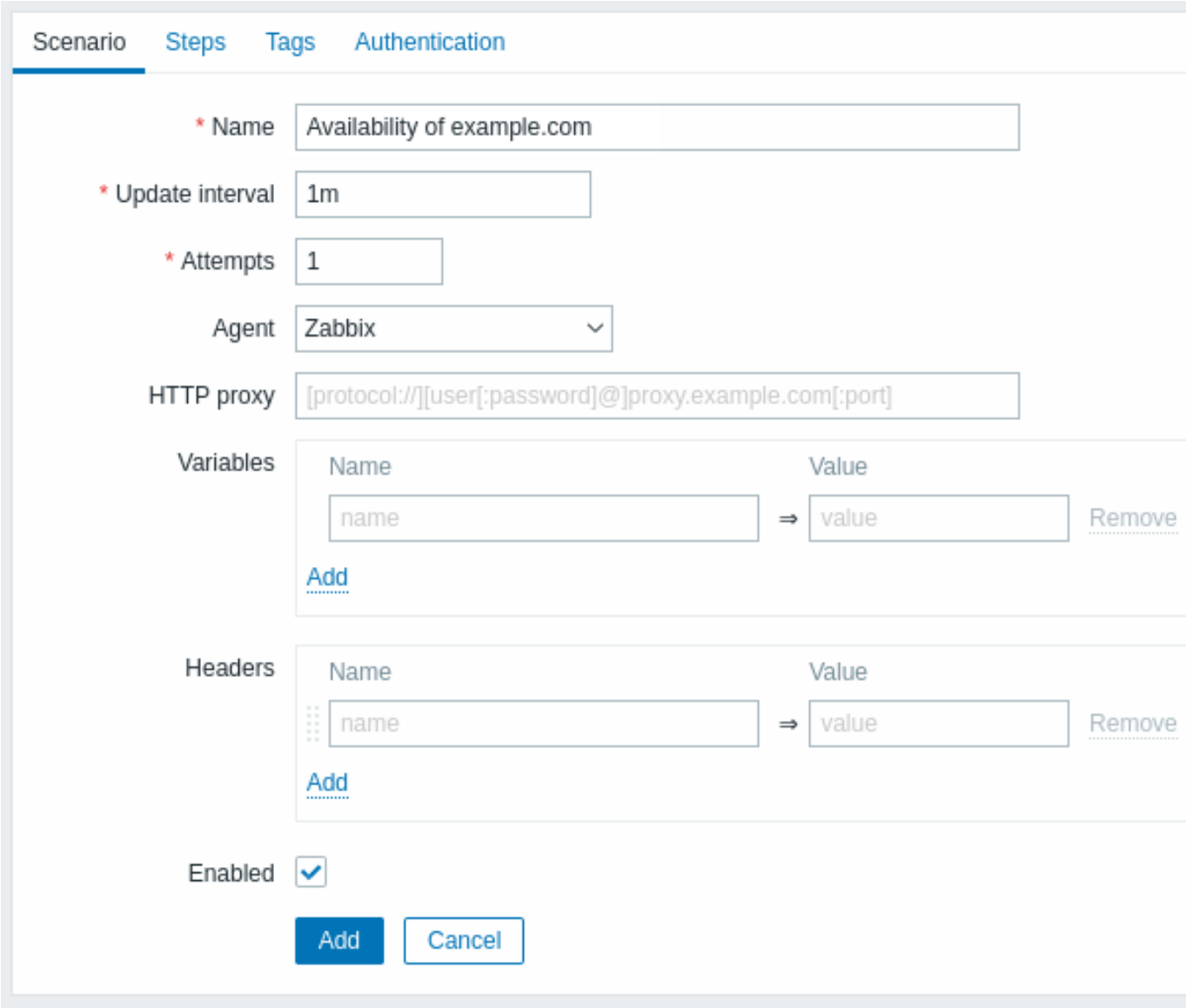
Zabbix también puede comprobar si una página HTML recuperada contiene una cadena predefinida. Puede ejecutar un inicio de sesión simulado y seguir una ruta de clics simulados del mouse en la página.

La supervisión web de Zabbix admite tanto HTTP como HTTPS. Al ejecutar un escenario web, Zabbix seguirá opcionalmente las redirecciones (consulte la opción *Seguir redirecciones* más abajo). El número máximo de redirecciones está codificado en 10 (usando la opción de cURL CURLOPT_MAXREDIRS). Todas las cookies se conservan durante la ejecución de un único escenario.

Configuración de un escenario web Para configurar un escenario web:

1. Vaya a: *Recopilación de datos* > *Hosts* (o *Templates*).
2. Haga clic en *Web* en la fila del host/template.
3. Haga clic en *Crear escenario web* a la derecha (o en el nombre del escenario para editar un escenario existente).
4. Introduzca los parámetros del escenario en el formulario.

La pestaña **Escenario** permite configurar los parámetros generales de un escenario web.



Todos los campos de entrada obligatorios están marcados con un asterisco rojo.

Parámetros del escenario:

Parámetro	Descripción
Nombre	Nombre único del escenario. Se admiten macros de usuario. *Nota*: si se utilizan macros de usuario, estas macros permanecerán sin resolver en los nombres de los items de monitorización web.
Intervalo de actualización	Frecuencia con la que se ejecutará el escenario. Se admiten sufijos de tiempo, por ejemplo, 30s, 1m, 2h, 1d. Se admiten macros de usuario. *Nota*: si se utiliza una macro de usuario y se cambia su valor (por ejemplo, 5m → 30s), la siguiente comprobación se ejecutará según el valor anterior (más adelante en el futuro con los valores del ejemplo). Los escenarios web nuevos se comprobarán en un plazo de 60 segundos desde su creación.
Intentos	Número de intentos para ejecutar los pasos del escenario web. En caso de problemas de red (tiempo de espera agotado, falta de conectividad, etc.), Zabbix puede repetir la ejecución de un paso varias veces. El valor establecido se aplicará por igual a cada paso del escenario. Se pueden especificar hasta 10 intentos; el valor predeterminado es 1. *Nota*: Zabbix no repetirá un paso debido a un código de respuesta incorrecto o a la falta de coincidencia de una cadena requerida.
Agent	Seleccione un agent cliente. Zabbix simulará ser el navegador seleccionado. Esto resulta útil cuando un sitio web devuelve contenido diferente para distintos navegadores. En este campo se pueden utilizar macros de usuario.

Parámetro	Descripción
Proxy HTTP	Puede especificar un proxy HTTP que se utilizará con el formato `[protocol://][username[:password]@]proxy.example.com[:port]`. Esto establece la opción cURL CURLOPT_PROXY. El prefijo opcional `protocol://` puede utilizarse para especificar protocolos de proxy alternativos (la compatibilidad con el prefijo de protocolo se añadió en cURL 7.21.7). Si no se especifica ningún protocolo, el proxy se tratará como un proxy HTTP. De forma predeterminada, se utilizará el puerto 1080. Si se especifica, el proxy sobrescribirá las variables de entorno relacionadas con el proxy, como http_proxy y HTTPS_PROXY. Si no se especifica, el proxy no sobrescribirá las variables de entorno relacionadas con el proxy. El valor introducido se transmite "tal cual", sin realizar comprobaciones de validez. También puede introducir una dirección de proxy SOCKS. Si especifica el protocolo incorrecto, la conexión fallará y el item dejará de ser compatible. *Nota*: con un proxy HTTP solo se admite la autenticación simple. En este campo se pueden utilizar macros de usuario.
Variables	Variables que pueden utilizarse en los pasos del escenario (URL y variables de publicación). Tienen el siguiente formato: **{macro1}**=value1 **{macro2}**=value2 **{macro3}**=regex:<regular expression> **{macro4}**=jsonpath:<jsonpath> **{macro5}**=xmlxpath:<xmlxpath> **{macro6}**={{macro}.function()} (consulte funciones de macros) Por ejemplo: {username}=Alexei {password}=kj3h5kJ34bd {hostid}=regex:hostid is ([0-9]+) {url}=jsonpath:$.host_url {status}=xmlxpath://host/response/status {newvar}={{myvar}.btoa()} Después, las macros pueden referenciarse en los pasos como {username}, {password}, {hostid}, etc. Zabbix las sustituirá automáticamente por sus valores reales. Tenga en cuenta que las variables con `regex:` necesitan un paso para obtener el valor de la expresión regular, por lo que el valor extraído solo puede aplicarse al paso siguiente. Si la parte del valor comienza con `regex:`, la parte posterior se trata como una expresión regular que busca en la página web y, si la encuentra, almacena la coincidencia en la variable. Debe haber al menos un subgrupo para poder extraer el valor coincidente. Se admiten macros de usuario y macros `{HOST.*}`. Las variables se codifican automáticamente en formato URL cuando se utilizan en campos de consulta o en datos de formulario para variables de publicación, pero deben codificarse manualmente en formato URL cuando se utilizan en una publicación sin formato o directamente en la URL.

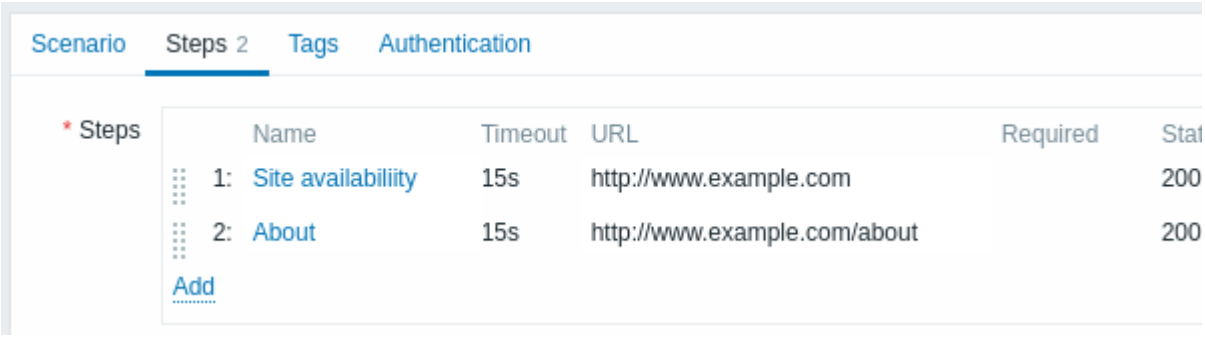
Parámetro	Descripción
Encabezados	Los encabezados HTTP se utilizan al realizar una solicitud. Se pueden utilizar encabezados predeterminados y personalizados. Los encabezados se asignarán mediante la configuración predeterminada según el tipo de Agent seleccionado en una lista desplegable a nivel del escenario y se aplicarán a todos los pasos, a menos que se definan de forma personalizada a nivel del paso. **Debe tenerse en cuenta que definir el encabezado a nivel del paso descarta automáticamente todos los encabezados definidos anteriormente, excepto un encabezado predeterminado asignado al seleccionar 'User-Agent' en una lista desplegable a nivel del escenario.** Sin embargo, incluso el encabezado predeterminado 'User-Agent' puede sobrescribirse especificándolo a nivel del paso. Para anular el encabezado a nivel del escenario, el encabezado debe nombrarse y asignarse sin ningún valor a nivel del paso. Los encabezados deben enumerarse utilizando la misma sintaxis con la que aparecerían en el protocolo HTTP, opcionalmente usando algunas funciones adicionales compatibles con la opción cURL CURLOPT_HTTPHEADER. Por ejemplo: Accept-Charset=utf-8 Accept-Language=en-US Content-Type=application/xml; charset=utf-8 Se admiten macros de usuario y macros `{HOST.*}`.
Habilitado	El escenario está activo si esta casilla está marcada; de lo contrario, está deshabilitado.

Tenga en cuenta que, al editar un escenario existente, hay dos botones adicionales disponibles en el formulario:

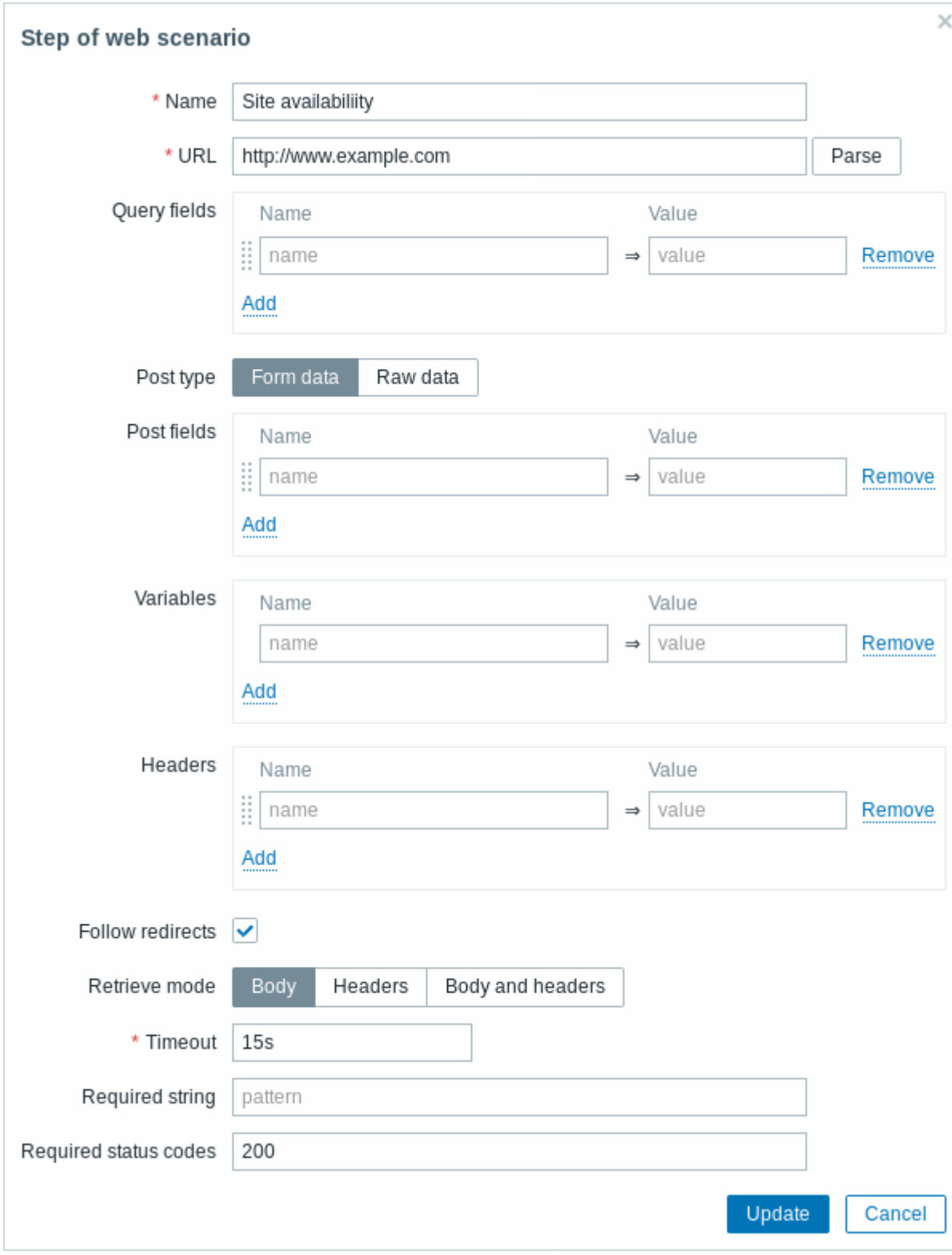
Clone	Crea otro escenario basándose en las propiedades del existente.
Clear history and trends	Elimina los datos históricos y de tendencias del escenario. Esto hará que el server ejecute el escenario inmediatamente después de eliminar los datos.

Note:
Si el campo *Proxy HTTP* se deja vacío, otra forma de utilizar un proxy HTTP consiste en establecer las variables de entorno relacionadas con el proxy.
Para comprobaciones HTTP, establezca la variable de entorno `http_proxy` para el usuario del server de Zabbix. Por ejemplo, `http_proxy=http://proxy_ip:proxy_port`.
Para comprobaciones HTTPS, establezca la variable de entorno `HTTPS_PROXY`. Por ejemplo, `HTTPS_PROXY=http://proxy_ip:proxy_port`. Hay más detalles disponibles ejecutando un comando de shell: `# man curl`.

La pestaña **Pasos** permite configurar los pasos del escenario web. Para añadir un paso de escenario web, haga clic en *Añadir* en el bloque *Pasos*.



Note:
Las macros de usuario secretas no deben utilizarse en las URL, ya que se resolverán como "******".



Pasos de configuración

Parámetros del paso:

Parámetro	Descripción
Nombre	Nombre único del paso. Se admiten macros de usuario. *Nota*: si se utilizan macros de usuario, estas macros quedarán sin resolver en los nombres de los items de monitorización web.

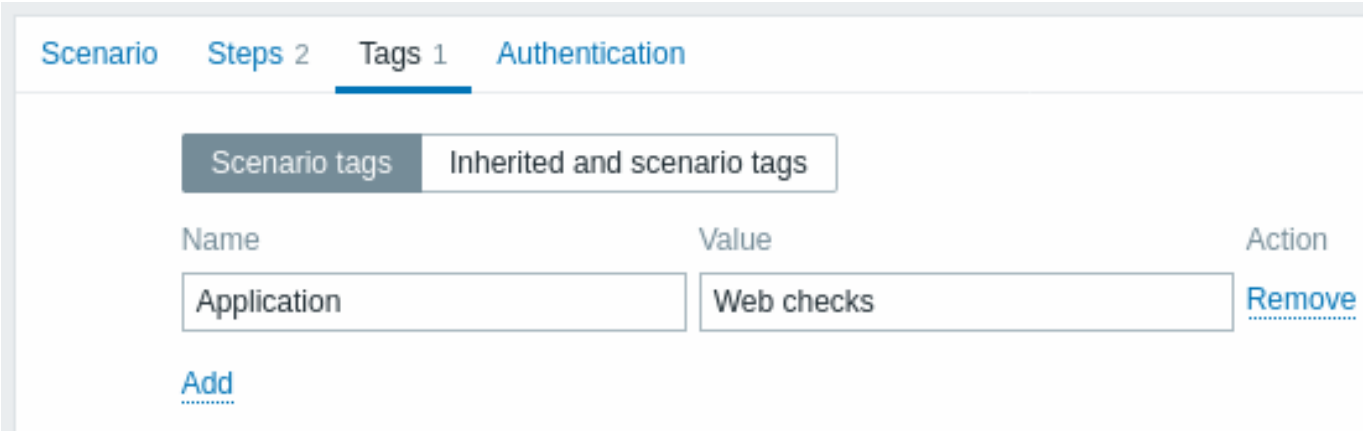
Parámetro	Descripción
URL	URL a la que conectarse y de la que recuperar datos. Por ejemplo: https://www.example.com http://www.example.com/download Los nombres de dominio se pueden especificar mediante caracteres Unicode. Se convierten automáticamente a punycode ASCII al ejecutar el paso del escenario web. El botón *Analizar* se puede utilizar para separar los campos de consulta opcionales (como `?key=value&limit=10`) de la URL y mover los atributos y valores a *Campos de consulta* para la codificación automática de la URL. Se pueden utilizar variables en la URL mediante la sintaxis {macro}. Las variables se pueden codificar manualmente en la URL mediante la sintaxis {{macro}.urlencode()}. Se admiten macros de usuario y macros `{HOST.*}`. Por motivos de seguridad, no incluya nombres de usuario ni contraseñas reales en las URL. Utilice macros de usuario para evitar exponer las credenciales. Limitado a 2048 caracteres.
Campos de consulta	Variables HTTP GET para la URL. Se especifican como pares de atributo y valor. Los valores se codifican automáticamente en la URL. Los valores de las variables del escenario, las macros de usuario o las macros `{HOST.*}` se resuelven y, a continuación, se codifican automáticamente en la URL. El uso de la sintaxis {{macro}.urlencode()} codificará dos veces la URL. Se admiten macros de usuario y macros `{HOST.*}`.
Post	Variables HTTP POST. En el modo **Datos de formulario**, se especifican como pares de atributo y valor. Los valores se codifican automáticamente en la URL. Los valores de las variables del escenario, las macros de usuario o las macros `{HOST.*}` se resuelven y, a continuación, se codifican automáticamente en la URL. En el modo **Datos sin procesar**, los atributos y valores se muestran en una sola línea y se concatenan con el símbolo **&**. Los valores sin procesar se pueden codificar o descodificar manualmente en la URL mediante la sintaxis {{macro}.urlencode()} o {{macro}.urldecode()}. Por ejemplo: id=2345&userid={user} Si {user} está definido como una variable del escenario web, se sustituirá por su valor al ejecutar el paso. Si desea codificar la variable en la URL, sustituya {user} por {{user}.urlencode()}. Se admiten macros de usuario y macros `{HOST.*}`.
Variables	Variables de nivel de paso que se pueden utilizar para las funciones GET y POST. Se especifican como pares de atributo y valor. Las variables de nivel de paso tienen prioridad sobre las variables de nivel de escenario o las variables del paso anterior. Sin embargo, el valor de una variable de nivel de paso solo afecta al paso siguiente (no al paso actual). Tienen el siguiente formato: **{macro}**=valor **{macro}**=regex:<expresión regular> Para obtener más información, consulte la descripción de las variables en el nivel de escenario. Las variables se codifican automáticamente en la URL cuando se utilizan en campos de consulta o en datos de formulario para variables POST, pero deben codificarse manualmente en la URL cuando se utilizan en datos POST sin procesar o directamente en la URL.
Encabezados	Encabezados HTTP personalizados que se enviarán al realizar una solicitud. Se especifican como pares de atributo y valor. Un encabezado definido en el nivel de paso se utilizará para ese paso concreto. **Debe tenerse en cuenta que definir el encabezado en el nivel de paso descarta automáticamente todos los encabezados definidos anteriormente, excepto un encabezado predeterminado asignado al seleccionar 'User-Agent' en una lista desplegable del nivel de escenario.** Sin embargo, incluso el encabezado predeterminado 'User-Agent' se puede sustituir especificándolo en el nivel de paso. Por ejemplo, asignar un nombre a un encabezado, pero no establecer ningún valor, anulará el encabezado predeterminado del nivel de escenario. Se admiten macros de usuario y macros `{HOST.*}`. Esto establece la opción de cURL `CURLOPT_HTTPHEADER`.
Seguir redirecciones	Marque la casilla para seguir las redirecciones HTTP. Esto establece la opción de cURL `CURLOPT_FOLLOWLOCATION`.

Parámetro	Descripción
Modo de recuperación	Seleccione el modo de recuperación: **Cuerpo** - recuperar solo el cuerpo de la respuesta HTTP **Encabezados** - recuperar solo los encabezados de la respuesta HTTP **Cuerpo y encabezados** - recuperar el cuerpo y los encabezados de la respuesta HTTP
Tiempo de espera	Zabbix no dedicará más del tiempo establecido al procesamiento de la URL (desde un segundo hasta un máximo de 1 hora). En realidad, este parámetro define el tiempo máximo para establecer la conexión con la URL y el tiempo máximo para realizar una solicitud HTTP. Por lo tanto, Zabbix no dedicará más de **2 x Tiempo de espera** segundos al paso. Se admiten sufijos de tiempo, por ejemplo, 30s, 1m, 1h. Se admiten macros de usuario.
Cadena requerida	Patrón de expresión regular requerido. El paso fallará si el contenido recuperado (HTML) no coincide con el patrón requerido. Si está vacío, no se realiza ninguna comprobación de la cadena requerida. Por ejemplo: Homepage of Zabbix Welcome.*admin *Nota*: este campo no admite referencias a expresiones regulares creadas en el frontend de Zabbix. Se admiten macros de usuario y macros `{HOST.*}`.
Códigos de estado requeridos	Lista de códigos de estado HTTP esperados. Si Zabbix obtiene un código que no está en la lista, el paso fallará. Si está vacía, no se realiza ninguna comprobación de los códigos de estado. Por ejemplo: 200,201,210-299 Se admiten macros de usuario.

Note:
Los cambios en los pasos del escenario web solo se guardarán cuando se guarde el escenario completo.

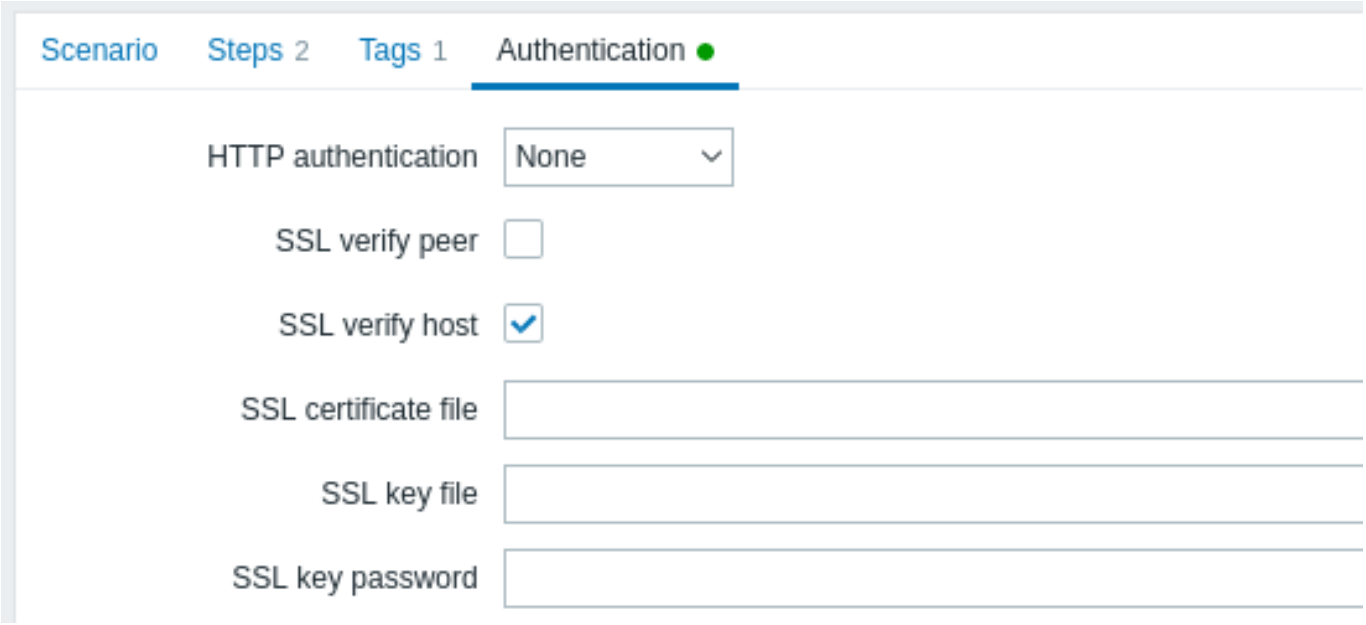
Consulte también un ejemplo práctico de cómo se pueden configurar los pasos de monitorización web.

Configuración de tags La pestaña **Tags** permite definir tags a nivel de escenario.



El uso de tags permite filtrar escenarios web y items de monitorización web.

Configuración de la autenticación La pestaña **Authentication** permite configurar las opciones de autenticación del escenario. Un punto verde junto al nombre de la pestaña indica que algún tipo de autenticación HTTP está habilitado.



Parámetros de autenticación:

Parameter	Description
HTTP authentication	Seleccione la opción de autenticación: **None**: no se utiliza autenticación; **Basic**: se utiliza autenticación básica; **NTLM**: se utiliza autenticación NTLM (Windows NT LAN Manager); **Kerberos**: se utiliza autenticación Kerberos (consulte también: Configuración de Kerberos con Zabbix); **Digest**: se utiliza autenticación Digest.
User	Introduzca el nombre de usuario (hasta 255 caracteres). Este campo está disponible si *HTTP authentication* está configurado como Basic, NTLM, Kerberos o Digest. Se admiten macros de usuario.
Password	Introduzca la contraseña del usuario (hasta 255 caracteres). Este campo está disponible si *HTTP authentication* está configurado como Basic, NTLM, Kerberos o Digest. Se admiten macros de usuario.
SSL verify peer	Marque la casilla para verificar el certificado SSL del web server. El certificado del server se verifica con los certificados de la autoridad certificadora (CA) de confianza de la ubicación de CA de todo el sistema. Puede sobrescribir la ubicación de los archivos de CA mediante el parámetro de configuración del server o proxy de Zabbix SSLCALocation. Esto establece la opción de cURL CURLOPT_SSL_VERIFYPEER.
SSL verify host	Marque la casilla para verificar que el campo *Common Name* o el campo *Subject Alternate Name* del certificado del web server coincida. Esto establece la opción de cURL CURLOPT_SSL_VERIFYHOST.
SSL certificate file	Nombre del archivo de certificado SSL utilizado para la autenticación del cliente. El archivo de certificado debe estar en formato PEM[1]. Si el archivo de certificado también contiene la clave privada, deje vacío el campo *SSL key file*. Si la clave está cifrada, especifique la contraseña en el campo *SSL key password*. El directorio que contiene este archivo se especifica mediante el parámetro de configuración del server o proxy de Zabbix SSLCertLocation. En este campo se pueden utilizar macros `HOST.*` y macros de usuario. Esto establece la opción de cURL CURLOPT_SSLCERT.
SSL key file	Nombre del archivo de clave privada SSL utilizado para la autenticación del cliente. El archivo de clave privada debe estar en formato PEM[1]. El directorio que contiene este archivo se especifica mediante el parámetro de configuración del server o proxy de Zabbix SSLKeyLocation. En este campo se pueden utilizar macros `HOST.*` y macros de usuario. Esto establece la opción de cURL CURLOPT_SSLKEY.
SSL key password	Contraseña del archivo de clave privada SSL. En este campo se pueden utilizar macros de usuario. Esto establece la opción de cURL CURLOPT_KEYPASSWD.

Attention:
[1] Zabbix admite únicamente archivos de certificado y de clave privada en formato PEM. Si los datos del certificado y de la clave privada están en un archivo con formato PKCS #12 (normalmente con la extensión *.p12 o *.pfx), puede generar el archivo PEM a partir de él mediante los siguientes comandos:

```
openssl pkcs12 -in ssl-cert.p12 -clcerts -nokeys -out ssl-cert.pem
openssl pkcs12 -in ssl-cert.p12 -nocerts -nodes  -out ssl-cert.key
```

Note:
El server de Zabbix detecta los cambios en los certificados sin necesidad de reiniciarse.

Note:
Si el certificado del cliente y la clave privada se encuentran en un solo archivo, especifíquelo en el campo ”SSL certificate file” y deje vacío el campo ”SSL key file”. El certificado y la clave deben seguir estando en formato PEM. Combinar el certificado y la clave es sencillo:

```
cat client.crt client.key > client.pem
```

Visualización Para ver los escenarios web configurados para un host:

1. Vaya a *Monitoring* > *Hosts*.
2. Localice el host en la lista y haga clic en el hipervínculo *Web* de la última columna.
3. Haga clic en el nombre del escenario para obtener información detallada.



También puede mostrar una descripción general de los escenarios web en *Dashboards* mediante el widget de monitorización web.

Los resultados recientes de la ejecución del escenario web están disponibles en la sección *Monitoring* > *Latest data*.

Monitoreo extendido A veces es necesario registrar el contenido de la página HTML recibida.
Esto es especialmente útil si falla algún paso de un escenario web.

El nivel de depuración 5 (trace) cumple ese propósito.
Este nivel se puede configurar en los archivos de configuración de server y proxy o mediante una opción de control en tiempo de ejecución (`-R log_level_increase="http poller,N"`, donde N es el número del proceso).
Los siguientes ejemplos muestran cómo se puede iniciar el monitoreo extendido siempre que ya esté configurado el nivel de depuración 4:

```
# Increase log level of all http pollers:
zabbix_server -R log_level_increase="http poller"

# Increase log level of second http poller:
zabbix_server -R log_level_increase="http poller,2"
```

Si no se requiere el monitoreo web extendido, se puede detener usando la opción `-R log_level_decrease`.

Elementos de monitorización web

Descripción general

Al crear escenarios web, se añaden automáticamente algunas nuevas métricas para su monitorización.

Todas las métricas heredan las etiquetas del escenario web.

Métricas de escenario

Tan pronto como se crea un escenario, Zabbix añade automáticamente las siguientes métricas para su monitorización.

Métrica	Descripción
Velocidad de descarga para el escenario <Escenario>	Esta métrica recopilará información sobre la velocidad de descarga (bytes por segundo) de todo el escenario, es decir, el promedio de todos los pasos. Clave de la métrica: web.test.in[Escenario,,bps] Tipo: *Numérico (coma flotante)*
Paso fallido del escenario <Escenario>	Esta métrica mostrará el número del paso que falló en el escenario. Si todos los pasos se ejecutan correctamente, se devuelve 0. Clave de la métrica: web.test.fail[Escenario] Tipo: *Numérico (sin signo)*
Último mensaje de error del escenario <Escenario>	Esta métrica devuelve el texto del último mensaje de error del escenario. Se almacena un nuevo valor solo si el escenario tiene un paso fallido. Si todos los pasos están bien, no se recopila un nuevo valor. Clave de la métrica: web.test.error[Escenario] Tipo: *Caracter*

Se utilizará el nombre real del escenario en lugar de "Escenario".

Note:
Si el nombre del escenario contiene macros de usuario, estas macros permanecerán sin resolver en los nombres de las métricas de monitorización web.

 Si el nombre del escenario comienza con una comilla doble o contiene una coma o un corchete, se citará correctamente en las claves de las métricas. En otros casos no se realizará ninguna cita adicional.

Note:
Las métricas de monitorización web se añaden con un historial de 30 días y un período de retención de tendencias de 90 días.

Estas métricas pueden utilizarse para crear disparadores y definir condiciones de notificación.

Ejemplo 1

Para crear un disparador de "Escenario web fallido", puede definir una expresión de disparador:

```
last(/equipo/web.test.fail[Escenario])<>0
```

Asegúrese de reemplazar 'Escenario' por el nombre real de su escenario.

Ejemplo 2

Para crear un disparador ”Escenario web fallido” con una descripción útil del problema en el nombre del disparador, puede definir un disparador con el nombre:

```
Escenario web "Escenario" fallido: {ITEM.VALUE}
```

y la expresión del disparador:

```
length(last(/equipo/web.test.error[Escenario]))>0 and last(/equipo/web.test.fail[Escenario])>0
```

Asegúrese de reemplazar ’Escenario’ con el nombre real de su escenario.

Ejemplo 3

Para crear un disparador ”La aplicación web es lenta”, puede definir una expresión de disparador:

```
last(/equipo/web.test.in[Escenario,,bps])<10000
```

Asegúrese de reemplazar ’Escenario’ con el nombre real de su escenario.

Métricas de pasos de escenario

Tan pronto como se crea un paso, Zabbix agrega automáticamente las siguientes métricas para su monitoreo.

Métrica	Descripción
Velocidad de descarga para el paso <Paso> del escenario <Escenario>	Esta métrica recopilará información sobre la velocidad de descarga (bytes por segundo) del paso. Clave de la métrica: web.test.in[Escenario,Paso,bps] Tipo: *Numérico(decimal)*
Tiempo de respuesta para el paso <Paso> del escenario <Escenario>	Esta métrica recopilará información sobre el tiempo de respuesta del paso en segundos. El tiempo de respuesta se cuenta desde el inicio de la solicitud hasta que se ha transferido toda la información. Clave de la métrica: web.test.time[Escenario,Paso,resp] Tipo: *Numérico(decimal)*
Código de respuesta para el paso <Paso> del escenario <Escenario>	Esta métrica recopilará los códigos de respuesta del paso. Clave de la métrica: web.test.rspcode[Escenario,Paso] Tipo: *Numérico(sin signo)*

Se utilizarán los nombres reales del escenario y del paso en lugar de ”Escenario” y ”Paso” respectivamente.

Note:
Las métricas de monitoreo web se agregan con un historial de 30 días y un período de retención de tendencias de 90 días.

Note:
Si el nombre del escenario comienza con una comilla doble o contiene una coma o corchete, se citará correctamente en las claves de las métricas. En otros casos no se realizará ninguna cita adicional.

Estas métricas pueden utilizarse para crear disparadores y definir condiciones de notificación. Por ejemplo, para crear un disparador ”El inicio de sesión en la GUI de Zabbix es demasiado lento”, puede definir una expresión de disparador:

```
last(/zabbix/web.test.time[ZABBIX GUI,Login,resp])>3
```

Escenario de la vida real

Resumen

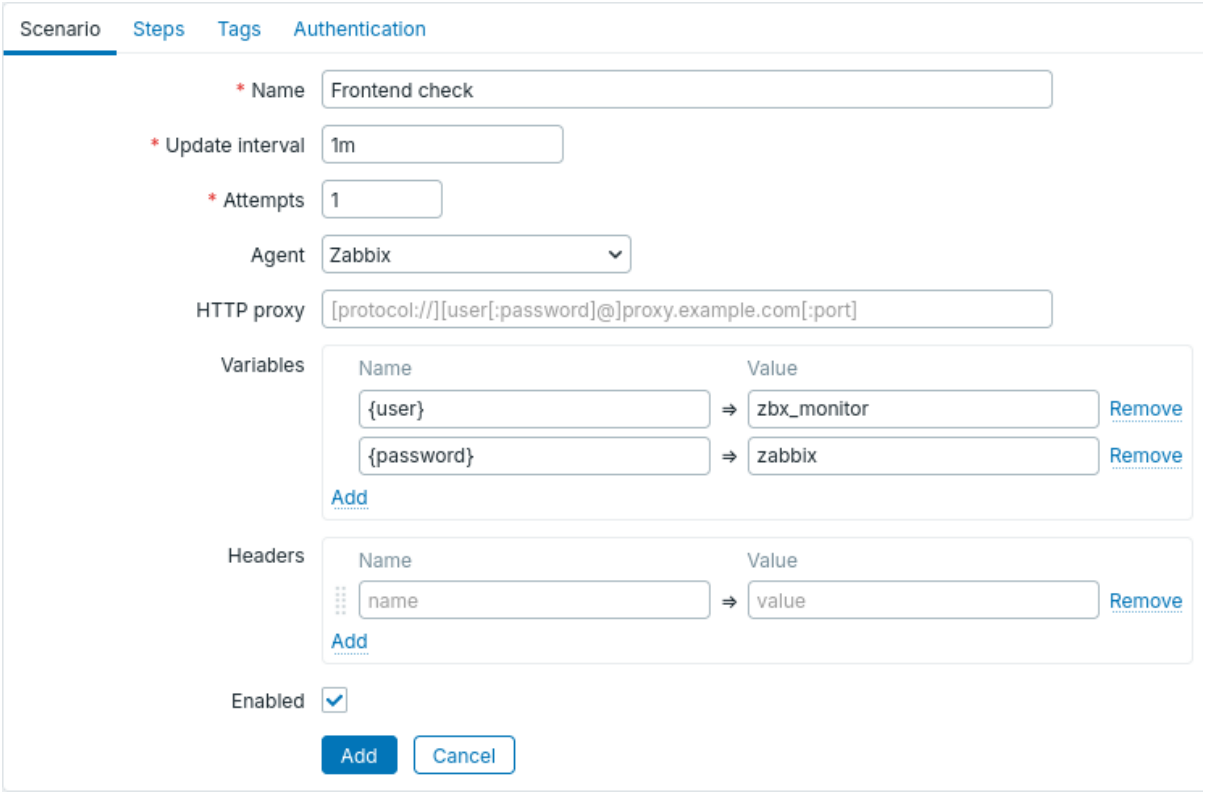
Esta sección presenta un ejemplo práctico paso a paso de cómo se puede utilizar la monitorización web.

La monitorización web de Zabbix se utilizará para supervisar el frontend de Zabbix. El objetivo es determinar si está disponible, si proporciona el contenido correcto y con qué rapidez funciona. Para lograrlo, se requieren varios pasos, entre ellos comprobar la disponibilidad de la primera página, iniciar sesión con un nombre de usuario y una contraseña, verificar que el inicio de sesión se haya realizado correctamente, cerrar sesión y confirmar el cierre de sesión.

Escenario

Añadir un nuevo escenario web

1. Vaya a *Data collection* > *Hosts*.
2. Elija un host y haga clic en *Web* en la fila de ese host.
3. Haga clic en *Create web scenario*.



4. En el formulario del nuevo escenario, en la pestaña *Scenario*, complete los siguientes campos:
    - *Name*: Frontend check.
    - *Update interval*: 1m.
    - *Attempts*: 1.
    - *Agent*: Zabbix.
 En la sección *Variables*, agregue los siguientes pares nombre-valor:
    - `{user}`: sus credenciales de usuario de Zabbix existentes;
    - `{password}`: sus credenciales de usuario de Zabbix existentes.

Note:
Por razones de seguridad, se recomienda crear un usuario independiente con permisos mínimos para usarlo con fines de monitoreo.

Opcionalmente, cambie a la pestaña *Tags* y agregue etiquetas del escenario web. Puede usar las etiquetas del escenario web para identificar rápidamente items y triggers relacionados o buscar entre los datos recopilados. Por ejemplo, las etiquetas adecuadas para este tutorial son `component: web-scenario` y/o `target: frontend`.

Configurar los pasos del escenario web

Cambie a la pestaña *Steps* y defina los pasos para el escenario.
 Después de configurar cada paso, haga clic en el botón *Add*.

Attention:
Tenga en cuenta que el frontend de Zabbix usa una redirección de JavaScript al iniciar sesión, por lo que el inicio de sesión debe realizarse primero, y las funciones que requieren sesión iniciada solo pueden comprobarse en los pasos posteriores. Además, el paso de inicio de sesión debe usar la URL completa del archivo **index.php**.

Campos comunes para todos los pasos

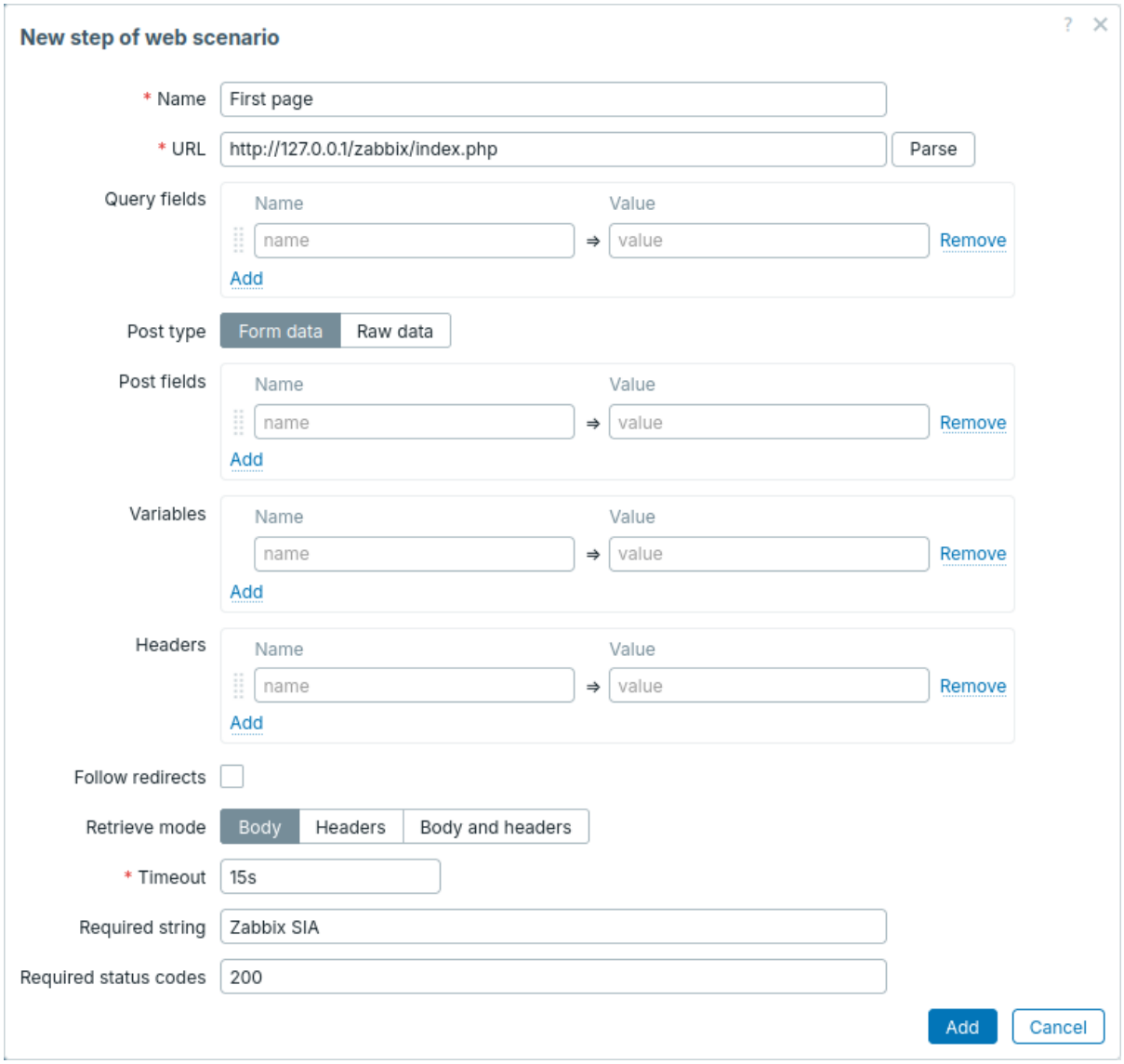
Además de los campos específicos de cada paso, complete los siguientes campos comunes para cada paso descrito a continuación:

- *URL*: la URL del frontend de Zabbix (`http://<server_ip_or_name>/zabbix/index.php` (para Apache) o `http://<server_ip_o` (para NGINX)). Para el paso 3 del escenario web, utilice una URL diferente, como se describe a continuación.
- *Timeout*: 15s.
- *Required status codes*: 200.

Paso 1 del escenario web

Compruebe que la primera página responde correctamente, devuelve el código de estado HTTP 200 y contiene el texto *Zabbix SIA*.
 Rellene los siguientes campos:

- *Name*: Primera página.
- *Timeout*: 15s.
- *Required string*: Zabbix SIA.
- Los campos comunes para todos los pasos.

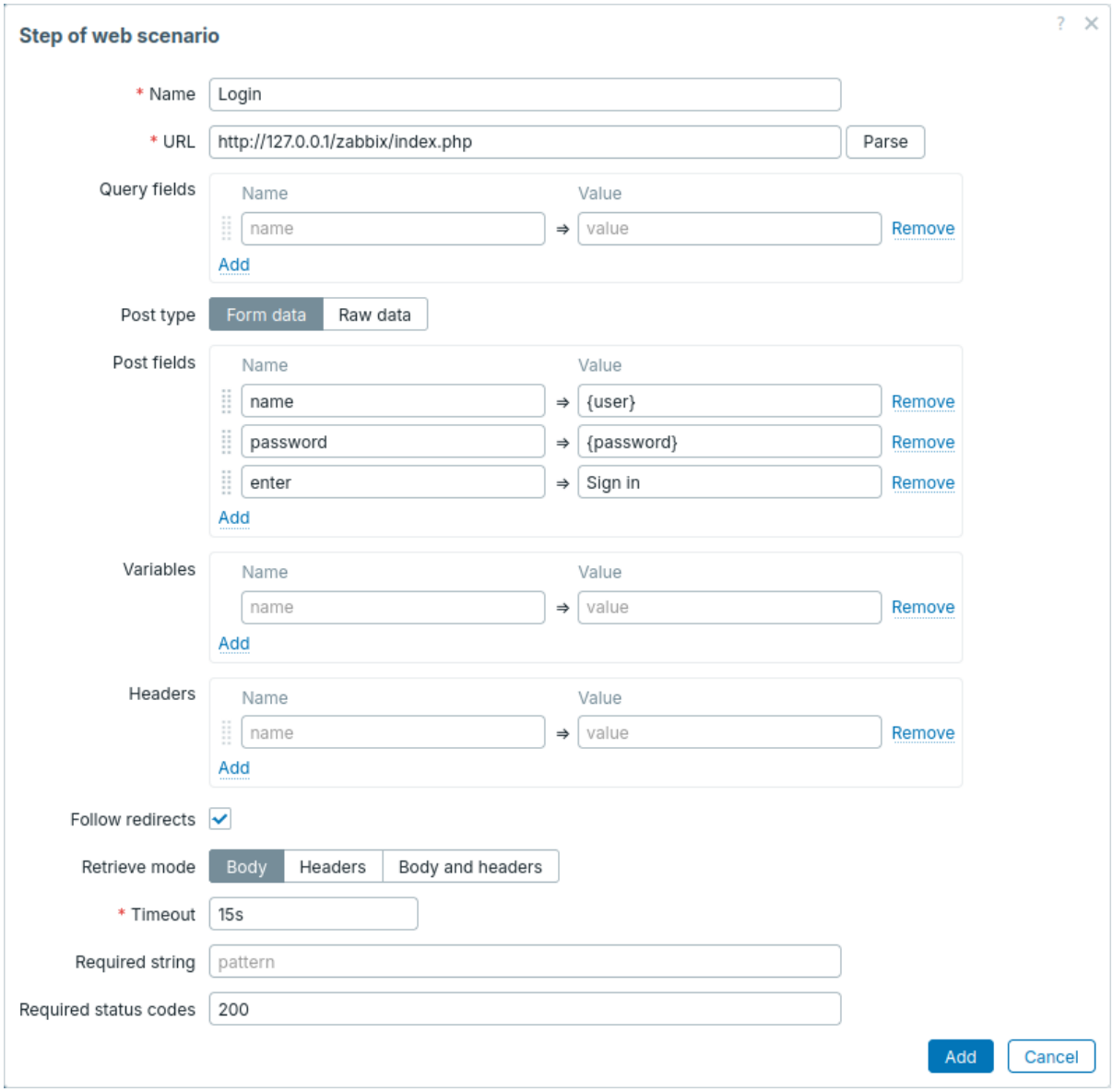


Paso 2 del escenario web

Inicie sesión en el frontend de Zabbix usando las macros (variables) definidas en la pestaña *Scenario*: *{user}* y *{password}*.
 Complete los siguientes campos:

- *Name*: Login.
- En la sección *Post fields*, agregue los siguientes pares nombre-valor:
  - name: `{user}`;
  - password: `{password}`;
  - enter: Sign in.
- Los campos comunes para todos los pasos.

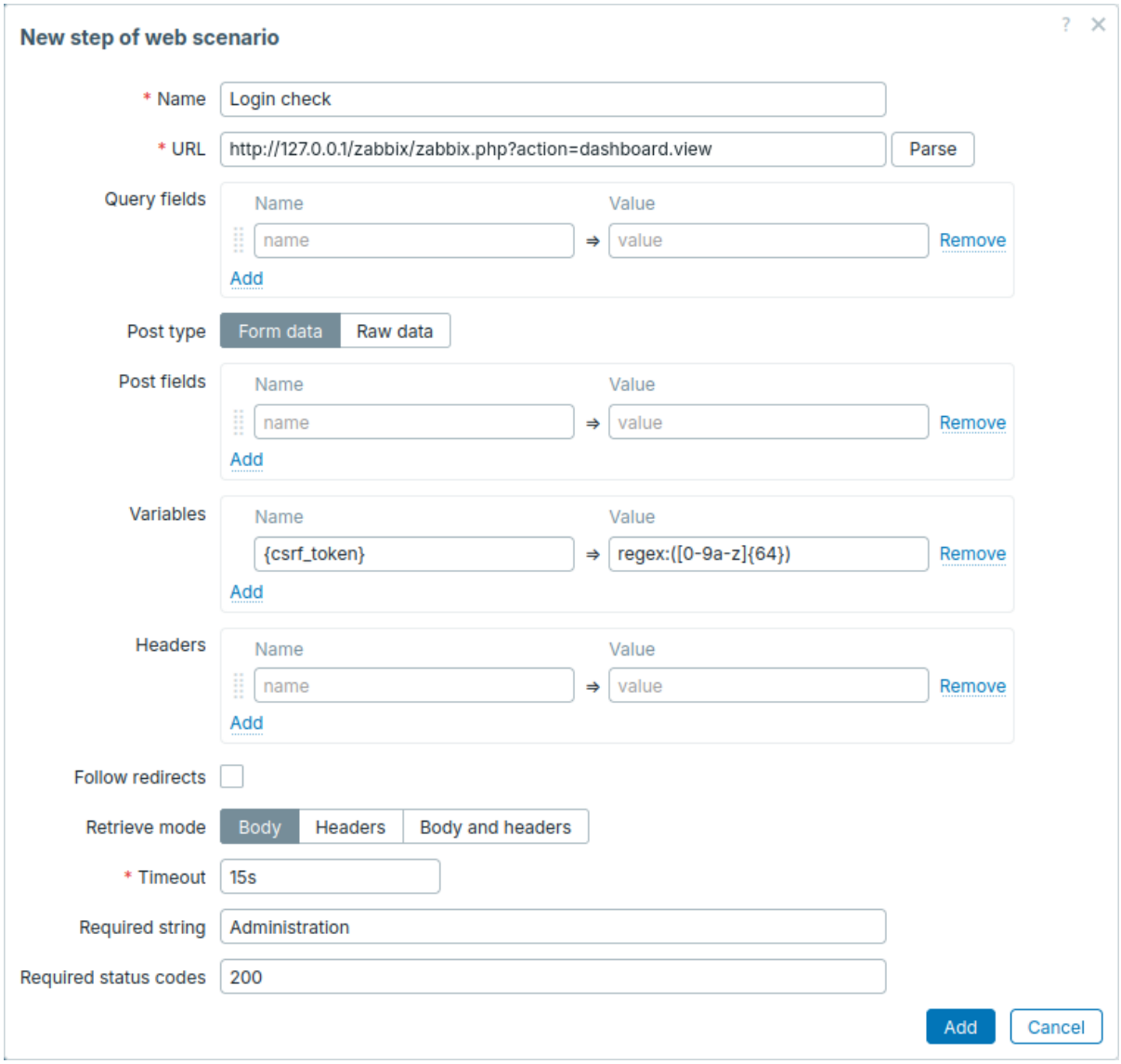
Seleccione la casilla *Follow redirects*.



Paso 3 del escenario web

Después de iniciar sesión, verifique el éxito comprobando una cadena visible solo cuando se ha iniciado sesión, por ejemplo, *Administration*.
 Complete los siguientes campos:

- *Name*: Login check.
- *URL*: la URL a la que se redirige al usuario después de iniciar sesión (por ejemplo, http://127.0.0.1/zabbix/zabbix.php?action=dashboard.v
- En la sección *Variables*, agregue el par nombre-valor `{csrf_token}`: `regex:([0-9a-z]{64})`. Esta variable capturará el valor del token CSRF asignado para reutilizarlo en el web scenario step 4.
- *Required string*: Administration.
- Los campos comunes para todos los pasos.



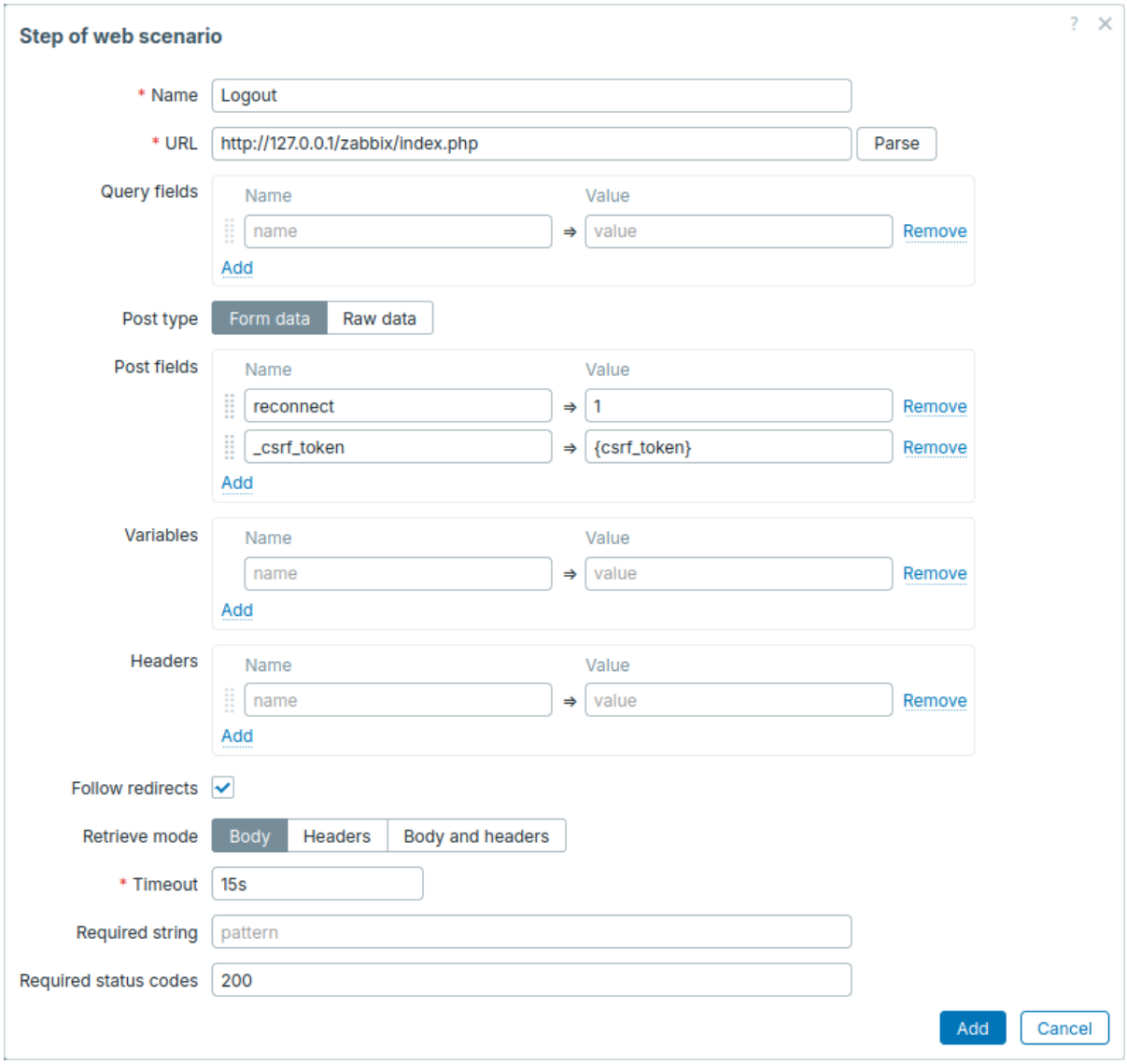
Paso 4 del escenario web

Una vez verificadas la accesibilidad y el inicio de sesión del frontend, añada un paso de cierre de sesión; de lo contrario, la base de datos de Zabbix se llenará de numerosos registros de sesiones abiertas.
 Complete los siguientes campos:

- *Name*: Logout.
- En la sección *Post fields*, añada los siguientes pares nombre-valor:
  - reconnect: `1`;
  - _csrf_token: `{csrf_token}`.
- Los campos comunes para todos los pasos.

Seleccione la casilla *Follow redirects*.

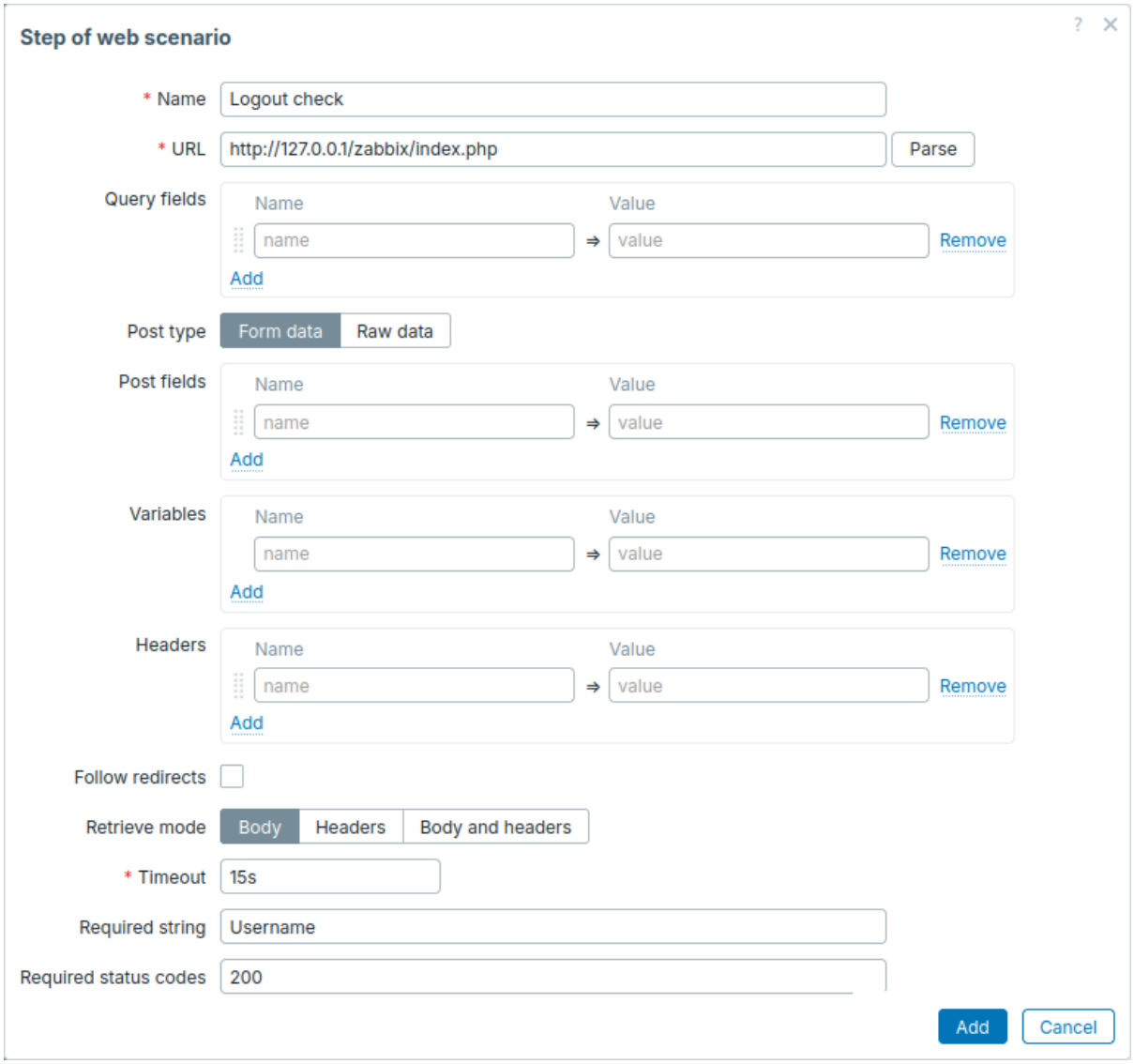
Este paso utiliza la variable `{csrf_token}` obtenida en el paso 3 del escenario web.



Paso 5 del escenario web

Para confirmar el cierre de sesión, compruebe la cadena *Username*.
 Rellene los siguientes campos:

- *Name*: comprobación de cierre de sesión.
- *Required string*: Username.
- Los campos comunes para todos los pasos.



Configuración completa de los pasos

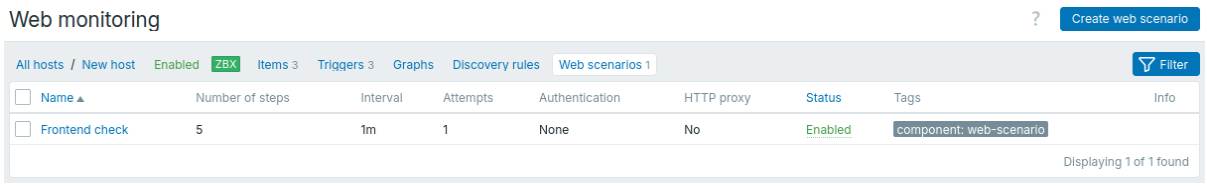
Una configuración completa de los pasos de un escenario web debería verse así:

Compruebe los resultados

Guarde el escenario de supervisión web terminado.

Cuando se crea el escenario web, se añaden automáticamente nuevos elementos de supervisión web para la supervisión.

El escenario se añadirá al host. Para ver la información del escenario web, vaya a *Monitoring* > *Hosts*, localice el host en la lista y haga clic en el hipervínculo *Web* de la última columna.



Haga clic en el nombre del escenario para ver estadísticas detalladas:



Supervisión de servicios

Descripción general La supervisión de servicios es una supervisión a nivel empresarial que puede utilizarse para obtener una visión general de todo el árbol de servicios de la infraestructura de TI, identificar los puntos débiles de la infraestructura, calcular el SLA de varios servicios de TI y consultar otra información a un nivel superior.
La supervisión de servicios se centra en la disponibilidad general de un servicio en lugar de en detalles de bajo nivel, como la falta de espacio en disco, una carga elevada del procesador, etc.
La supervisión de servicios también proporciona funcionalidad para encontrar la causa raíz de un problema si un servicio no funciona según lo esperado.

La supervisión de servicios proporciona una forma de crear una representación jerárquica de los datos supervisados.

Una estructura de servicios muy sencilla podría tener el siguiente aspecto:

```
Servicio
|
|-Estaciones de trabajo
| |
| |-Estación de trabajo1
| |
| |-Estación de trabajo2
|
|-Servidores
```

Cada nodo de la estructura tiene un atributo de estado.
El estado se calcula y se propaga a los niveles superiores según el algoritmo seleccionado.
El estado de los nodos individuales se ve afectado por el estado de los problemas asignados.
La asignación de problemas se realiza mediante el etiquetado.

Zabbix puede enviar notificaciones o ejecutar automáticamente un script en el Zabbix server en caso de que se detecte un cambio en el estado del servicio. Es posible definir reglas flexibles para determinar si un servicio principal debe pasar al «estado de problema» en función de los estados de los servicios secundarios.
Los datos de los problemas de los servicios pueden utilizarse para calcular el SLA y enviar informes de SLA basados en un conjunto flexible de condiciones.

La supervisión de servicios se configura en el menú Servicios, que consta de las siguientes secciones:

- Servicios

La sección Servicios permite crear una jerarquía de la infraestructura supervisada mediante la adición de servicios principales y, posteriormente, servicios secundarios a los servicios principales.

Además de configurar el árbol de servicios, esta sección proporciona una visión general de toda la infraestructura y permite identificar rápidamente los problemas que provocaron un cambio en el estado del servicio.

- SLA

En esta sección puede definir acuerdos de nivel de servicio y establecer objetivos de nivel de servicio para servicios específicos.

- Informe de SLA

En esta sección puede consultar informes de SLA.

Acciones de servicio

También puede configurar acciones de servicio.

Las acciones de servicio son opcionales y permiten:

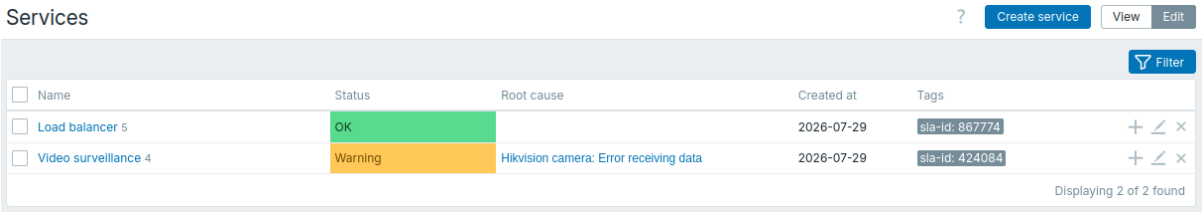
- enviar una notificación indicando que un servicio no está disponible
- ejecutar un comando remoto en el Zabbix server cuando cambia el estado del servicio
- enviar una notificación de recuperación cuando un servicio vuelve a estar disponible.

Véase también:

- Ejemplo de configuración de la supervisión de SLA
- Notas sobre la actualización de servicios desde versiones de Zabbix anteriores a la 6.0

Árbol de servicios

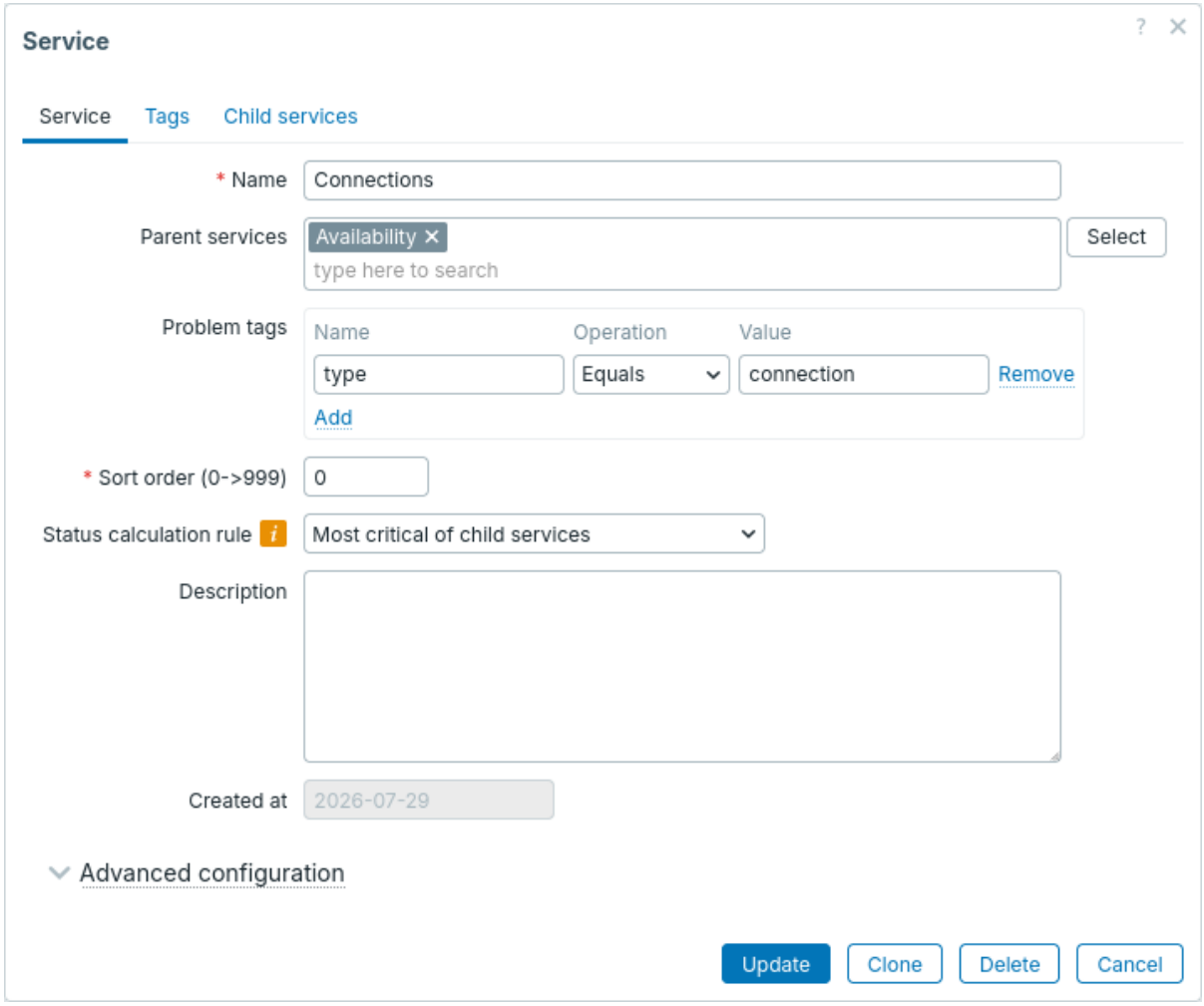
El árbol de servicios se configura en la sección de menú *Services* > *Services*. En la esquina superior derecha, cambie de view al modo de edición.



Para configurar un nuevo servicio, haga clic en el botón *Create service* en la esquina superior derecha.

Para agregar rápidamente un servicio secundario, también puede hacer clic en el icono de signo más junto al servicio principal. Esto abrirá el mismo formulario de configuración del servicio, pero el parámetro *Parent services* estará rellenado previamente.

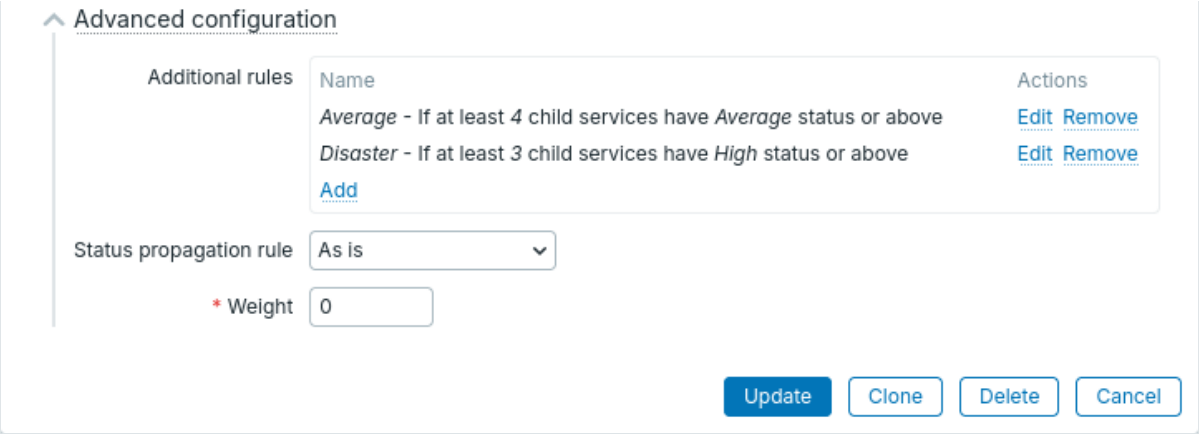
Configuración del service En la pestaña **Service**, especifique los parámetros requeridos del service:



Todos los campos de entrada obligatorios están marcados con un asterisco rojo.

Parameter	Description
Name	Nombre del service.
Parent services	Servicios padre a los que pertenece el service. Deje este campo vacío si está agregando el service de nivel más alto. Un service puede tener varios servicios padre. En este caso, se mostrará en el árbol de services debajo de cada uno de los servicios padre.
Problem tags	Especifique etiquetas para asignar los datos de problemas al service. Se pueden establecer varias condiciones. La coincidencia del nombre de la etiqueta siempre distingue entre mayúsculas y minúsculas. Hay dos operadores disponibles para cada condición: **Equals** - incluye los nombres y valores de etiqueta especificados (distingue entre mayúsculas y minúsculas) **Contains** - incluye los nombres de etiqueta especificados cuyos valores de etiqueta contienen la cadena introducida (coincidencia de subcadena, distingue entre mayúsculas y minúsculas)
Sort order	Orden de clasificación para la visualización; el valor más bajo aparece primero.
Status calculation rule	Regla para calcular el estado del service: **Most critical if all children have problems** - el problema más crítico en los servicios secundarios se usa para colorear el estado del service, si todos los servicios secundarios tienen problemas **Most critical of child services** - el problema más crítico en los servicios secundarios se usa para colorear el estado del service **Set status to OK** - no calcular el estado del service Se pueden configurar reglas adicionales de cálculo de estado en las opciones de advanced configuration.
Description	Descripción del service.
Created at	La hora en que se creó el service; se muestra al editar un service existente.
Advanced configuration	Haga clic en el encabezado *Advanced configuration* para mostrar las opciones de advanced configuration.

Configuración avanzada



Parameter	Description
Additional rules	Haga clic en *Add* para configurar reglas adicionales de cálculo de estado.
Set status to	Establezca el estado del servicio en *OK* (predeterminado), *Not classified*, *Information*, *Warning*, *Average*, *High* o *Disaster* en caso de que coincida una condición.
Condition	Seleccione la condición para los servicios secundarios directos: **si al menos (N) servicios secundarios tienen el estado (Status) o superior** **si al menos (N%) de los servicios secundarios tienen el estado (Status) o superior** **si menos de (N) servicios secundarios tienen el estado (Status) o inferior** **si menos de (N%) de los servicios secundarios tienen el estado (Status) o inferior** **si el peso de los servicios secundarios con el estado (Status) o superior es al menos (W)** **si el peso de los servicios secundarios con el estado (Status) o superior es al menos (N%)** **si el peso de los servicios secundarios con el estado (Status) o inferior es menor que (W)** **si el peso de los servicios secundarios con el estado (Status) o inferior es menor que (N%)** Si se especifican varias condiciones y la situación coincide con más de una, se establecerá la severidad más alta.
N (W)	Establezca el valor de N o W (1-100000), o N% (1-100) en la condición.
Status	Seleccione el valor de *Status* en la condición: *OK* (predeterminado), *Not classified*, *Information*, *Warning*, *Average*, *High* o *Disaster*.
Status propagation rule	Regla para propagar el estado del servicio al servicio padre: **As is** - el estado se propaga sin cambios **Increase by** - puede aumentar el estado propagado de 1 a 5 niveles de severidad **Decrease by** - puede disminuir el estado propagado de 1 a 5 niveles de severidad **Ignore this service** - el estado no se propaga al servicio padre en absoluto **Fixed status** - el estado se propaga de forma estática, es decir, siempre con el mismo valor
Weight	Peso del servicio (entero en el rango de 0 (predeterminado) a 1000000).

Note:
Las reglas adicionales de cálculo de estado solo pueden utilizarse para aumentar el nivel de severidad por encima del nivel calculado según el parámetro principal *Status calculation rule*. Si, según las reglas adicionales, el estado debe ser Warning, pero según *Status calculation rule* el estado es Disaster, el servicio tendrá el estado Disaster.

La pestaña *Tags* contiene tags de nivel de servicio. Los tags de nivel de servicio se utilizan para identificar un servicio. Los tags de este tipo no se utilizan para asignar problemas al servicio (para ello, utilice los *tags de problema* de la primera pestaña).

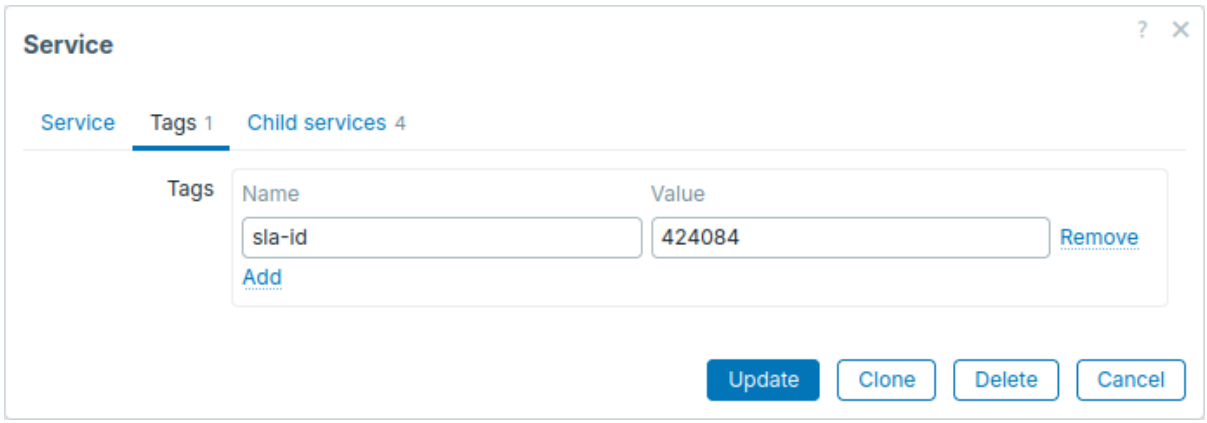
La pestaña *Child services* permite especificar servicios dependientes. Haga clic en *Add* para añadir un servicio de la lista de servicios existentes. Si desea añadir un nuevo servicio secundario, guarde primero este servicio y, a continuación, haga clic en el icono de signo más situado junto al servicio que acaba de crear.

Etiquetas Hay dos tipos diferentes de etiquetas en los servicios:

- Etiquetas de servicio
- Etiquetas de problema

Etiquetas de servicio

Las etiquetas de servicio se usan para asociar servicios con acciones de servicio y SLAs. Estas etiquetas se especifican en la pestaña de configuración del servicio *Tags*. Para la asignación de SLAs, se usa la lógica *OR*: un servicio se asignará a un SLA si tiene al menos una etiqueta coincidente. En las acciones de servicio, las reglas de asignación son configurables y pueden usar lógica *AND*, *OR* o *AND/OR*.

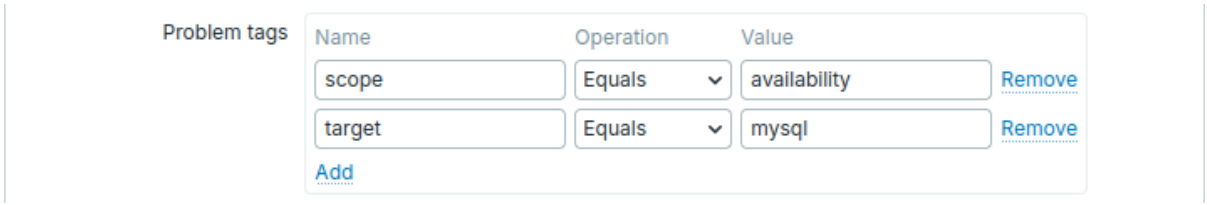


Etiquetas de problema

Las etiquetas de problema se utilizan para asociar problemas y servicios. Estas etiquetas se especifican en la pestaña de configuración del servicio principal.

Solo los servicios secundarios del nivel más bajo de la jerarquía pueden tener etiquetas de problema definidas y correlacionarse directamente con los problemas. Si las etiquetas de problema coinciden, el estado del servicio cambiará al mismo estado que tenga el problema. En caso de varios problemas, un servicio tendrá el estado del problema más grave. El estado de un servicio padre se calcula entonces en función de los estados de los servicios secundarios, de acuerdo con las reglas de cálculo de estado.

Si se especifican varias etiquetas, se usa la lógica *AND*: un problema debe tener todas las etiquetas especificadas en la configuración del servicio para asignarse al servicio.

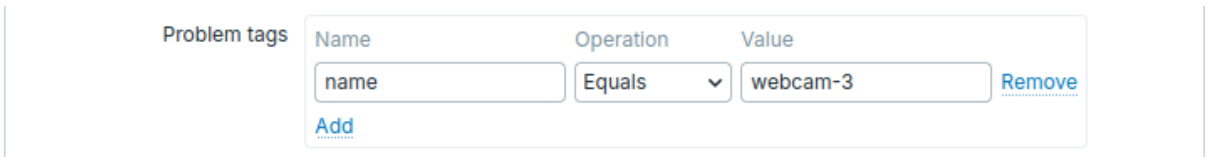


Note:
Un problema en Zabbix hereda etiquetas de toda la cadena de templates, hosts, items, escenarios web y triggers. Cualquiera de estas etiquetas puede utilizarse para asociar problemas con servicios.

Ejemplo:

El problema *Web camera 3 is down* tiene las etiquetas `type:video-surveillance`, `floor:1` y `name:webcam-3` y el estado *Warning*

El servicio **Web camera 3** tiene especificada solo la etiqueta de problema `name:webcam-3`



El estado del servicio cambiará de *OK* a *Warning* cuando se detecte este problema.

Si el servicio **Web camera 3** tuviera las etiquetas de problema `name:webcam-3` y `floor:2`, su estado no cambiaría cuando se detectara el problema, porque las condiciones solo se cumplen parcialmente.

Note:
Los botones descritos a continuación solo son visibles cuando la sección *Services* está en mod

Modificación de servicios existentes

Para editar un servicio existente, haga clic en el icono de lápiz junto al servicio.

Para clonar un servicio existente, haga clic en el icono de lápiz para abrir su configuración y luego haga clic en el botón *Clone*. Cuando se clona un servicio, se conservan sus enlaces al padre, mientras que los enlaces a los hijos no.

Para eliminar un servicio, haga clic en el icono x junto a él. Cuando elimine un servicio padre, sus servicios hijo no se eliminarán y subirán un nivel en el árbol de servicios (los hijos de 1.er nivel pasarán a tener el mismo nivel que el servicio padre eliminado).

Dos botones debajo de la lista de servicios ofrecen algunas opciones de edición masiva:

- *Mass update* - actualización masiva de las propiedades del servicio
- *Delete* - eliminar los servicios

Para usar estas opciones, marque las casillas de verificación delante de los servicios correspondientes y luego haga clic en el botón requerido.

SLA

Descripción general Una vez creados los servicios, puede comenzar a supervisar si su rendimiento cumple el acuerdo de nivel de servicio (SLA).

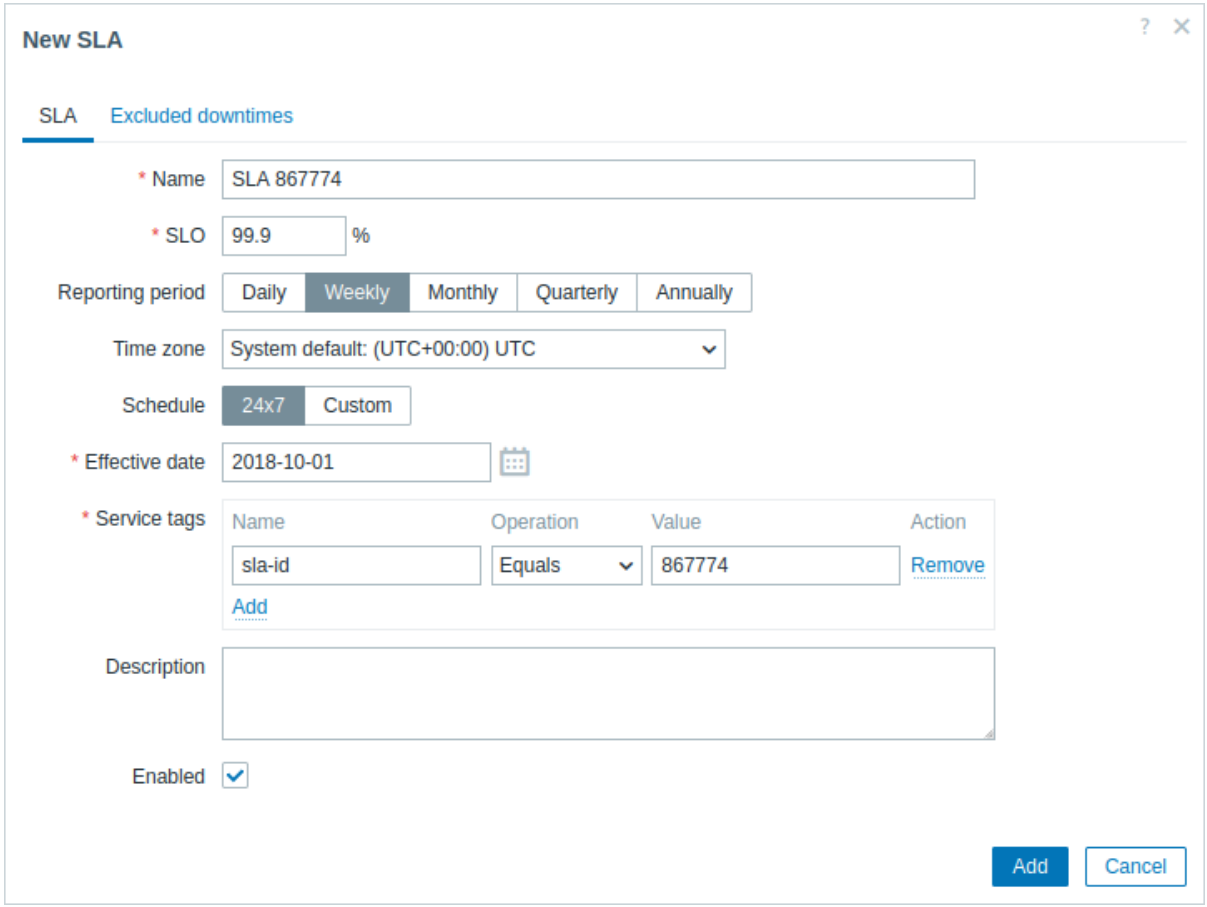
La sección de menú *Servicios > SLA* permite configurar SLA para varios servicios. Un SLA en Zabbix define el objetivo de nivel de servicio (SLO), el horario de disponibilidad esperado y los tiempos de inactividad planificados.

Los SLA y los servicios se relacionan mediante etiquetas de servicio. El mismo SLA puede aplicarse a varios servicios; el rendimiento se medirá por separado para cada servicio coincidente. Un solo servicio puede tener varios SLA asignados; los datos de cada SLA se mostrarán por separado.

En los informes de SLA, Zabbix proporciona datos del indicador de nivel de servicio (SLI), que mide la disponibilidad real del servicio. Para determinar si un servicio cumple los objetivos del SLA, se compara el SLO (disponibilidad esperada en %) con el SLI (disponibilidad real en %).

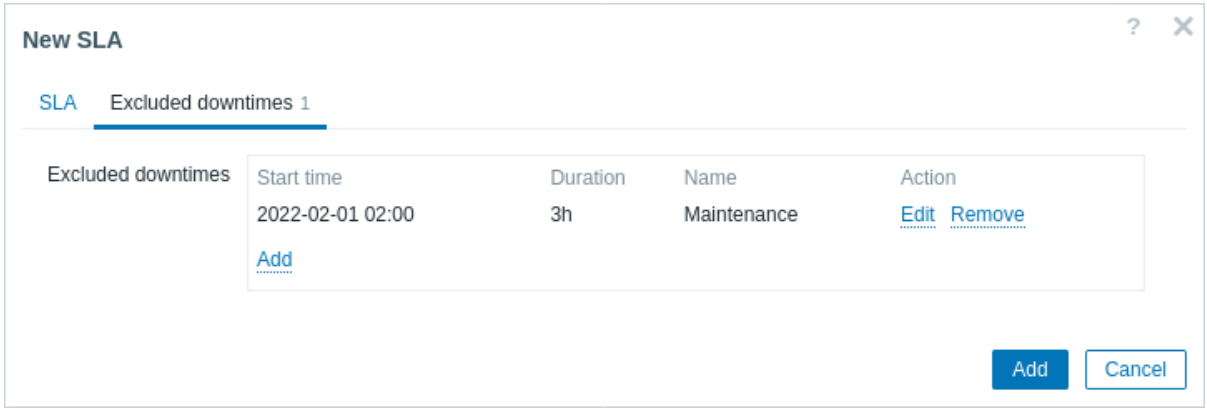
Configuración Para crear un SLA nuevo, haga clic en el botón *Crear SLA*.

La pestaña *SLA* permite especificar los parámetros generales del SLA.



Parámetro	Descripción
Nombre	Introduzca el nombre del SLA.
SLO	Introduzca el objetivo de nivel de servicio (SLO) como porcentaje.
Período de generación de informes	Seleccione el período de generación de informes del SLI utilizado para calcular los datos del SLI en el informe del SLA: **Diario**: cada día, de 00:00:00 a 23:59:59. **Semanal**: cada semana, de domingo a las 00:00:00 a sábado a las 23:59:59. **Mensual**: cada mes, desde el primer día a las 00:00:00 hasta el último día a las 23:59:59. **Trimestral**: cada trimestre natural: T1 (ene-mar), T2 (abr-jun), T3 (jul-sep), T4 (oct-dic). **Anual**: cada año natural, desde el 1 de enero a las 00:00:00 hasta el 31 de diciembre a las 23:59:59.
Zona horaria	Seleccione la zona horaria del SLA.
Programación	Seleccione la programación del SLA: 24x7 o personalizada.
Fecha de inicio	Seleccione la fecha de inicio del cálculo del SLA.
Etiquetas del servicio	Añada etiquetas de servicio para identificar los servicios a los que se debe aplicar este SLA. **Nombre**: nombre de la etiqueta de servicio; debe coincidir exactamente y distingue entre mayúsculas y minúsculas. **Operación**: seleccione *Igual a* si el valor de la etiqueta debe coincidir exactamente (distingue entre mayúsculas y minúsculas) o *Contiene* si debe coincidir una parte del valor de la etiqueta (no distingue entre mayúsculas y minúsculas). **Valor**: valor de la etiqueta de servicio que se buscará según la operación seleccionada. El SLA se aplica a un servicio si coincide al menos una etiqueta de servicio.
Descripción	Añada una descripción para el SLA.
Habilitado	Marque la casilla para habilitar el cálculo del SLA.

La pestaña *Tiempos de inactividad excluidos* permite especificar los tiempos de inactividad que se excluirán del cálculo del SLA.



Haga clic en *Añadir* para configurar los tiempos de inactividad excluidos y, a continuación, introduzca el nombre del período, la fecha de inicio y la duración.

Informes de SLA El rendimiento de un servicio en comparación con un SLA se puede ver en el informe de SLA. Los informes de SLA se pueden ver:

- desde la sección *SLA* haciendo clic en el hipervínculo del informe de SLA;
- desde la sección *Services* haciendo clic en el nombre del SLA en la pestaña de información;
- en el widget *SLA report* del Dashboard.

Una vez configurado un SLA, la pestaña *Info* en la sección de servicios también mostrará cierta información sobre el rendimiento del servicio.

Ejemplo de configuración

Descripción general Esta sección describe una configuración sencilla para monitorizar un clúster de alta disponibilidad de Zabbix como un servicio.

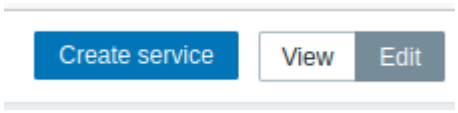
Requisitos previos Antes de configurar la monitorización de servicios, debe tener los equipos configurados:

- *Nodo HA 1* con al menos un disparador y una etiqueta (preferiblemente establecida a nivel de disparador) `component:ha-node-1`
- *Nodo HA 2* con al menos un disparador y una etiqueta (preferiblemente establecida a nivel de disparador) `component:ha-node-2`

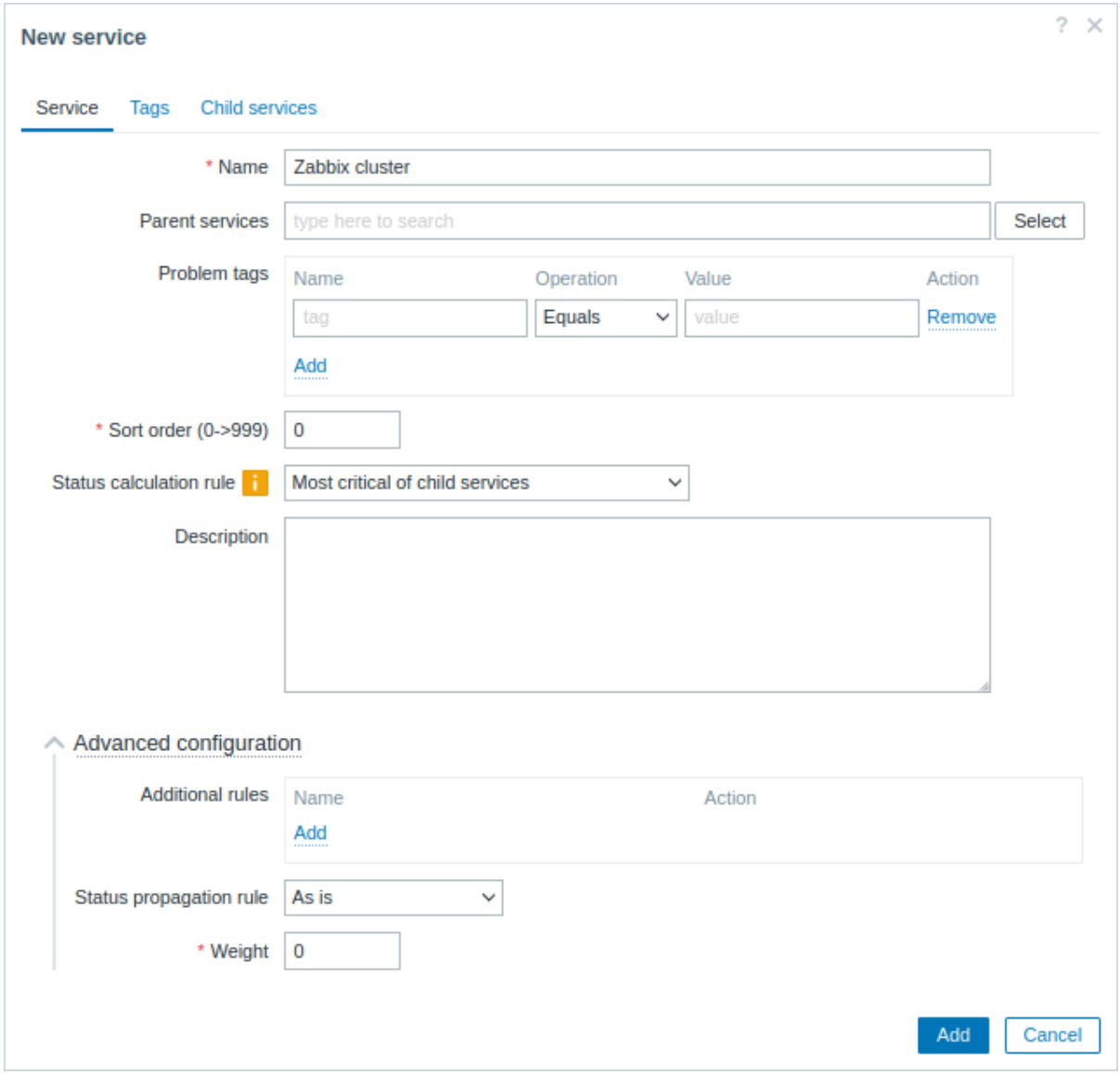
Árbol de servicios El siguiente paso es construir el árbol de servicios. En este ejemplo, la infraestructura es muy básica y consta de tres servicios: *Zabbix cluster* (padre) y dos servicios secundarios *Zabbix server node 1* y *Zabbix server node 2*.

```
Zabbix cluster
|
|- Zabbix server node 1
|- Zabbix server node 2
```

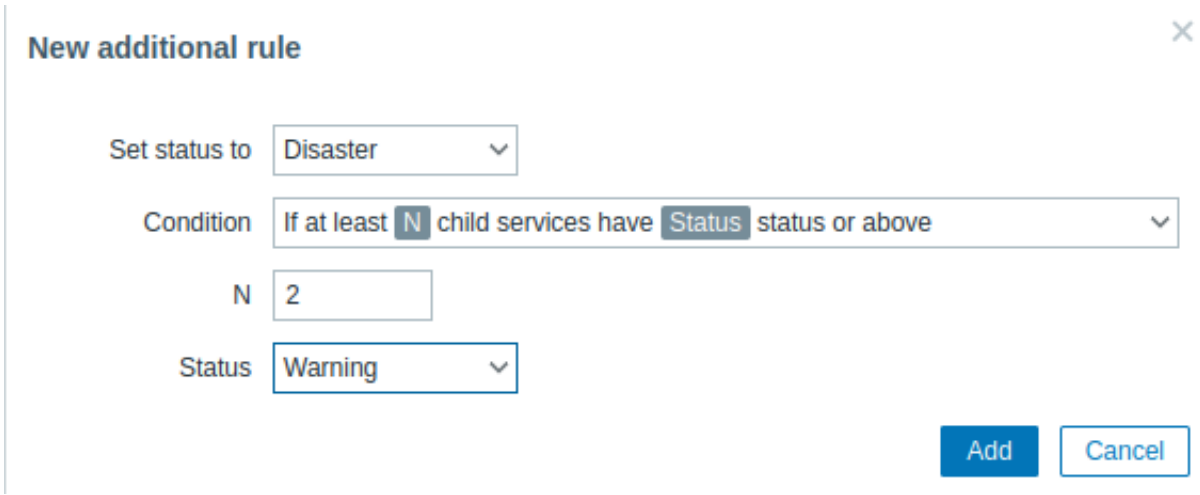
En la página de Services, active el modo *Edit* y haga clic en *Create service*:



En la ventana de configuración del servicio, introduzca el nombre *Zabbix cluster* y haga clic en el encabezado *Advanced configuration* para mostrar las opciones de configuración avanzada.



Configure una regla adicional:

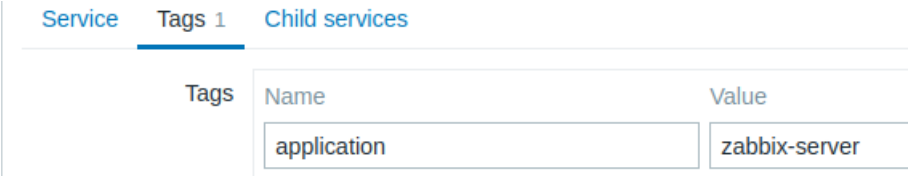


Zabbix cluster tendrá dos servicios secundarios, uno para cada uno de los nodos HA. Si ambos nodos HA presentan problemas con un estado de al menos *Warning*, el estado del servicio padre debe establecerse en *Disaster*. Para lograrlo, la regla adicional debe configurarse de la siguiente manera:

- Set status to: Disaster
- Condition: If at least `N` child services have `Status` status or above
- N: 2
- Status: Warning

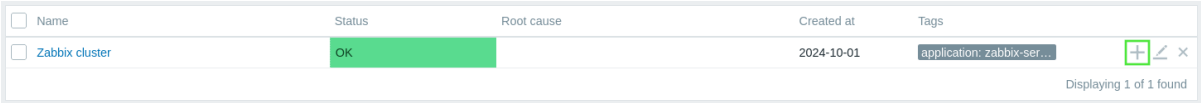
Cambie a la pestaña *Tags* y añada la etiqueta `application:zabbix-server`. Esta etiqueta se utilizará más adelante para las

acciones de servicio y los informes de SLA.



Guarde el nuevo servicio.

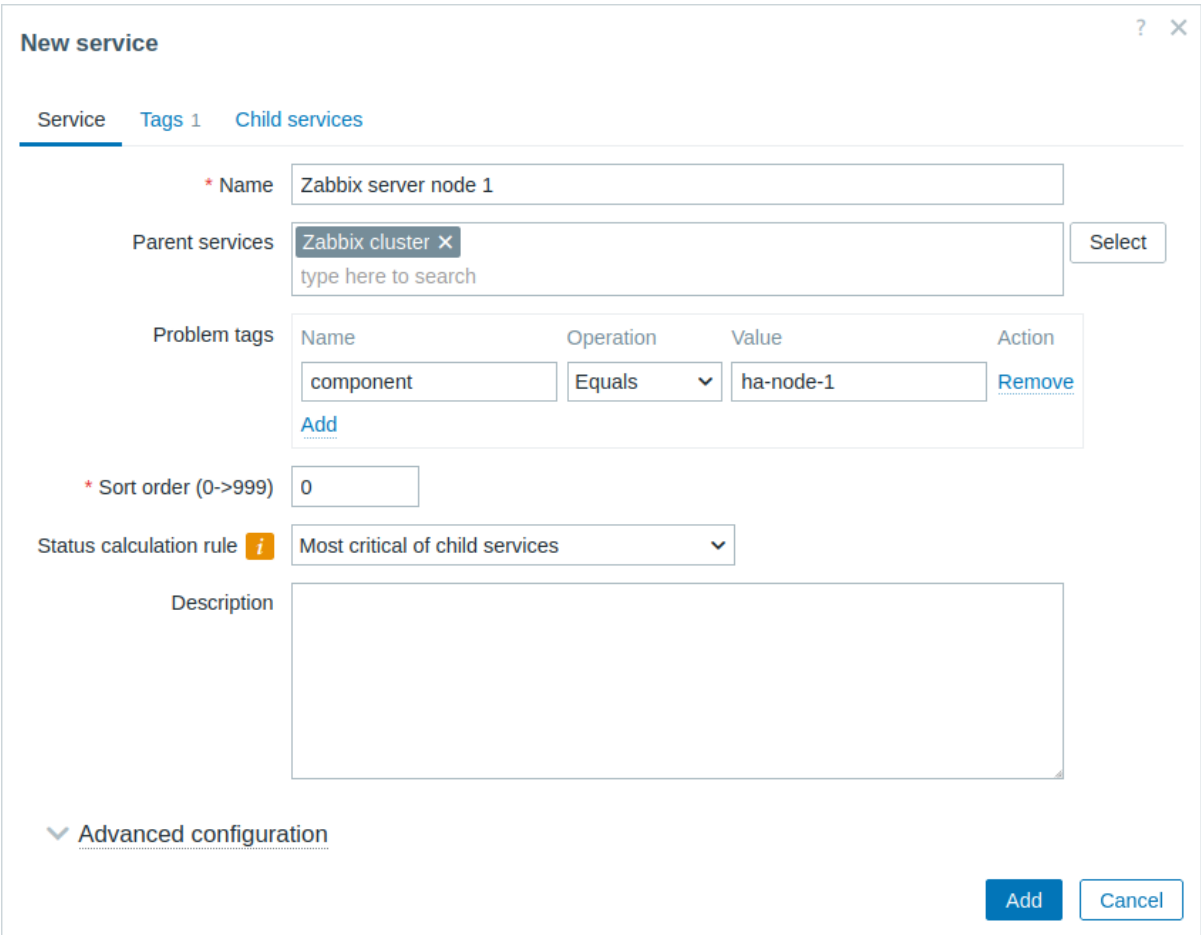
Para agregar un servicio secundario, haga clic en el icono de signo más junto al servicio del clúster de Zabbix (el icono solo es visible en modo *Edit*).



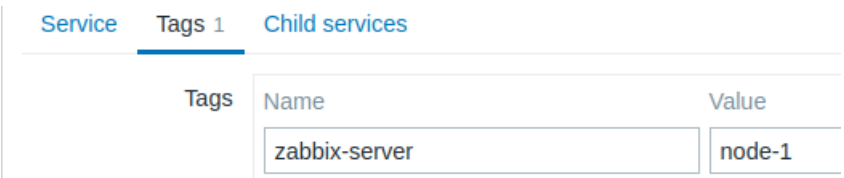
En la ventana de configuración del servicio, introduzca el nombre *Zabbix server node 1*. Tenga en cuenta que el parámetro Parent services ya está rellenado previamente con *Zabbix cluster*.

La disponibilidad de este servicio se ve afectada por problemas en el host *HA node 1*, marcados con la etiqueta de problema `component:ha-node-1`. En el parámetro Problem tags, introduzca:

- Name: component
- Operation: Equals
- Value: ha-node-1



Cambie a la pestaña *Tags* y añada una etiqueta de servicio: `zabbix-server:node-1`. Esta etiqueta se utilizará más adelante para las acciones de servicio y los informes de SLA.



Guarde el nuevo servicio.

Cree otro servicio hijo del clúster de Zabbix con el nombre ”Nodo 2 del servidor Zabbix”.

Establezca las etiquetas de problema como:

- Nombre: componente
- Operación: Igual
- Valor: ha-node-2

Cambie a la pestaña *Etiquetas* y agregue una etiqueta de servicio: `zabbix-server:node-2`.

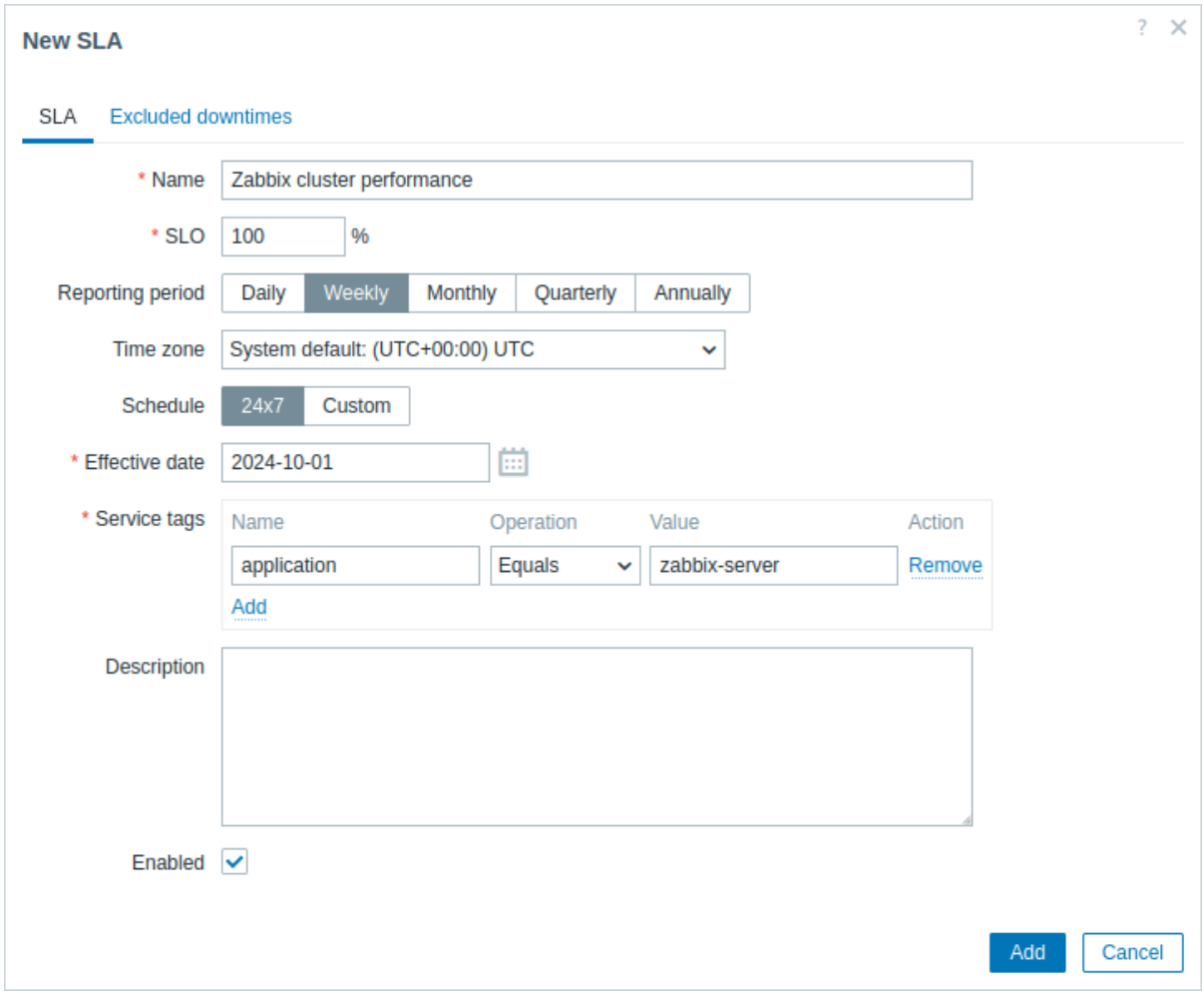
Guarde el nuevo servicio.

SLA En este ejemplo, el rendimiento esperado del clúster de Zabbix es del 100 %, excluyendo un período de mantenimiento semestral de una hora.

Primero, debe añadir un nuevo acuerdo de nivel de servicio:

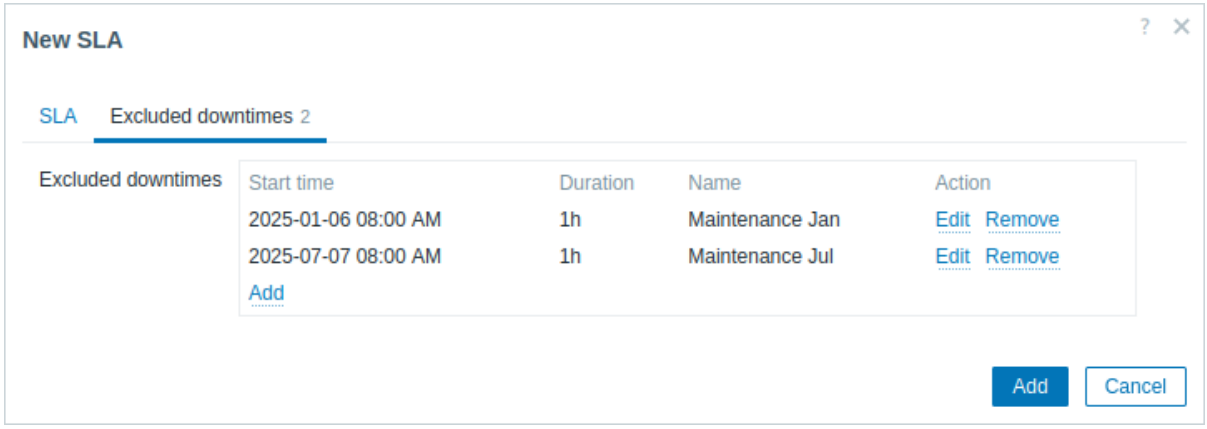
1. Vaya a *Services* > *SLA*.
2. Haga clic en *Create SLA*.
3. Introduzca el nombre *Zabbix cluster performance* y establezca el SLO en el 100 %. El servicio Zabbix cluster tiene la etiqueta de servicio `application:zabbix-server`.
4. Para utilizar este SLA para medir el rendimiento del clúster de Zabbix, especifique lo siguiente en el parámetro *Service tags*:

- Name: application
- Operation: Equals
- Value: zabbix-server



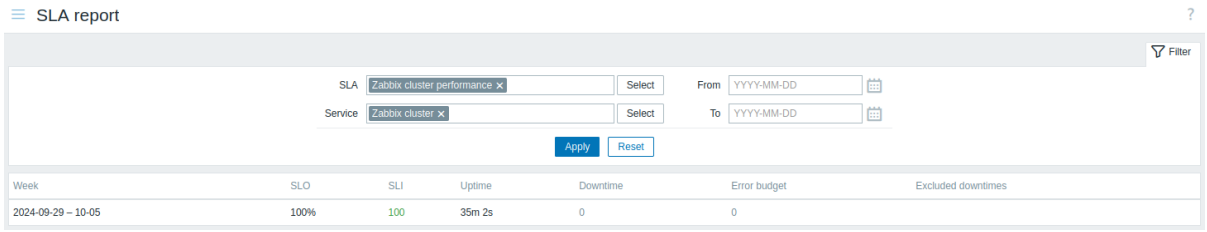
En una configuración real, también puede actualizar el período de generación de informes deseado, la zona horaria y la fecha de inicio, o cambiar el horario de 24/7 a uno personalizado. Para este ejemplo, la configuración predeterminada es suficiente.

5. Cambie a la pestaña *Excluded downtimes* y añada los períodos de inactividad correspondientes a los períodos de mantenimiento programados para excluirlos del cálculo del SLA.
6. En la sección *Excluded downtimes*, haga clic en el enlace *Add*, introduzca el nombre del período de inactividad, la hora de inicio planificada y la duración.

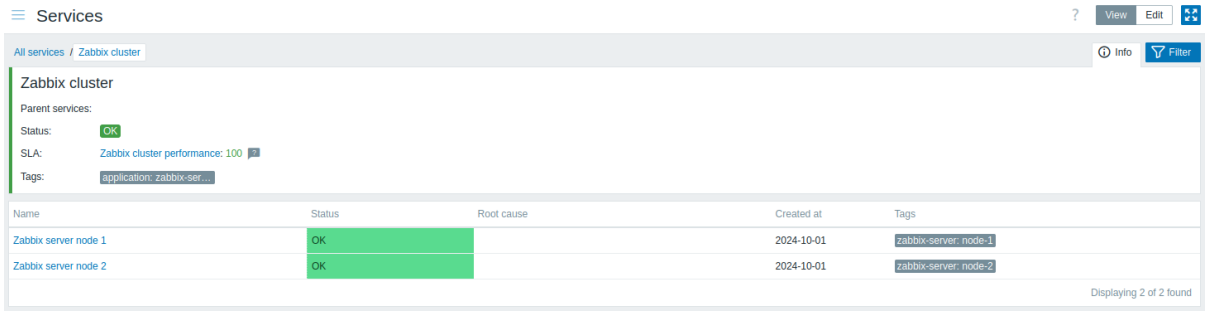


7. Haga clic en *Add* para guardar el nuevo SLA.

Cambie a la sección de informes de SLA para consultar el informe de SLA del clúster de Zabbix.



La información del SLA también se puede consultar en la sección *Services*.



Actualización de la monitorización de servicios

Resumen En Zabbix 6.0, la funcionalidad de monitorización de servicios se ha rediseñado de forma significativa (consulte Novedades de Zabbix 6.0.0 para ver la lista de cambios).

Esta página describe cómo se modifican los servicios y los SLA, definidos en versiones anteriores de Zabbix, durante una actualización a Zabbix 6.0 o posterior.

Servicios En versiones anteriores de Zabbix, los servicios tenían dos tipos de dependencias: blandas y duras. Después de una actualización, todas las dependencias pasarán a ser iguales.

Si un servicio ”Child service” se ha vinculado previamente a ”Parent service 1” mediante una dependencia dura y además a ”Parent service 2” mediante una dependencia blanda, después de una actualización el ”Child service” tendrá dos servicios padre: ”Parent service 1” y ”Parent service 2”.

La asignación entre problemas y servicios basada en triggers ha sido reemplazada por una asignación basada en etiquetas. En Zabbix 6.0 y versiones posteriores, el formulario de configuración del servicio tiene un nuevo parámetro *Problem tags*, que permite especificar uno o varios pares de nombre y valor de etiqueta para la coincidencia de problemas. Los triggers que se hayan vinculado a un servicio obtendrán una nueva etiqueta `ServiceLink`: `<trigger ID>:<trigger name>` (el valor de la etiqueta se truncará a 32 caracteres). Los servicios vinculados obtendrán `ServiceLink` problem tag con el mismo valor.

Reglas de cálculo del estado

El ”algoritmo de cálculo del estado” se actualizará usando las siguientes reglas:

- No calcular → Establecer el estado en OK

- Problem, si al menos un elemento secundario tiene un problema → El más crítico de los servicios secundarios
- Problem, si todos los elementos secundarios tienen problemas → El más crítico si todos los elementos secundarios tienen problemas

SLAs Anteriormente, los objetivos de SLA debían definirse por separado para cada service. Desde Zabbix 6.0, SLA se ha convertido en una entidad independiente, que contiene información sobre el horario del service, el objetivo de nivel de service esperado (SLO) y los períodos de inactividad que se excluirán del cálculo. Una vez configurado, un SLA puede asignarse a varios services mediante service tags.

Durante una actualización:

- Los SLA idénticos definidos para cada service se agruparán y se creará un SLA por cada grupo.
- Cada service afectado obtendrá una etiqueta especial `SLA:<ID>` y la misma etiqueta se especificará en el parámetro *Service tags* del SLA correspondiente.
- El tiempo de creación del service, una nueva métrica en los informes de SLA, se establecerá en 01/01/2000 00:00 para los services existentes.

Cifrado y seguridad

Encryption Zabbix admite comunicaciones cifradas entre los componentes de Zabbix mediante el protocolo Transport Layer Security (TLS) v.1.2 y 1.3 (según la biblioteca criptográfica). Se admite el cifrado basado en certificados y el cifrado basado en claves precompartidas.

El cifrado se puede configurar para las conexiones:

- Entre Zabbix server, Zabbix proxy, Zabbix agent, Zabbix web service, zabbix_sender y las utilidades zabbix_get
- Con la base de datos de Zabbix desde Zabbix frontend y server/proxy
- Entre Zabbix frontend y Zabbix server

El cifrado es opcional y se puede configurar para componentes individuales:

- Algunos proxies y agents pueden configurarse para usar cifrado basado en certificados con el server, mientras que otros pueden usar cifrado basado en claves precompartidas, y otros pueden seguir usando comunicaciones sin cifrar (como antes).
- El server (proxy) puede usar distintas configuraciones de cifrado para diferentes hosts.

Los programas daemon de Zabbix usan un único puerto de escucha para conexiones entrantes cifradas y no cifradas. Añadir cifrado no requiere abrir nuevos puertos en los firewalls.

Limitaciones

- Las claves privadas se almacenan en texto plano en archivos legibles por los componentes de Zabbix durante el inicio.
- Las claves precompartidas se introducen en Zabbix frontend y se almacenan en la base de datos de Zabbix en texto plano.
- El cifrado integrado no protege las comunicaciones entre el servidor web que ejecuta Zabbix frontend y el navegador web del usuario.
- Actualmente, cada conexión cifrada se abre con un handshake TLS completo; no se implementan caché de sesión ni tickets.
- Añadir cifrado aumenta el tiempo de las comprobaciones de item y de las acciones, según la latencia de la red:
  - Por ejemplo, si el retardo de los paquetes es de 100 ms, abrir una conexión TCP y enviar una solicitud sin cifrar tarda alrededor de 200 ms. Con cifrado, se añaden unos 1000 ms para establecer la conexión TLS.
  - Puede ser necesario aumentar los timeouts; de lo contrario, algunos items y acciones que ejecutan scripts remotos en agent pueden funcionar con conexiones sin cifrar, pero fallar por timeout con conexiones cifradas.
- El cifrado no es compatible con network discovery. Las comprobaciones de Zabbix agent realizadas por network discovery no estarán cifradas y, si Zabbix agent está configurado para rechazar conexiones sin cifrar, dichas comprobaciones no tendrán éxito.

Compilación de Zabbix con compatibilidad con el cifrado Para admitir el cifrado, Zabbix debe compilarse y vincularse con una de las bibliotecas criptográficas compatibles:

- GnuTLS: desde la versión 3.1.18
- OpenSSL: versiones 1.0.1, 1.0.2, 1.1.0, 1.1.1, 3.0.x - 3.5.x
- LibreSSL: probado con las versiones 2.7.4 y 2.8.2:
  - LibreSSL 2.6.x no es compatible
  - LibreSSL es compatible como reemplazo de OpenSSL; no se utilizan las nuevas funciones de API específicas de LibreSSL `tls_*()`. Los componentes de Zabbix compilados con LibreSSL no podrán utilizar PSK; solo se pueden utilizar certificados.

Note:
Puede obtener más información sobre la configuración de SSL para el frontend de Zabbix consultando estas prácticas recomendadas.

La biblioteca se selecciona especificando la opción correspondiente en el script `configure`:

- `--with-gnutls[=DIR]`
- `--with-openssl[=DIR]` (también se utiliza para LibreSSL)

Por ejemplo, para configurar las fuentes de server y agent con *OpenSSL*, puede utilizar algo como:

```
./configure --enable-server --enable-agent --with-mysql --enable-ipv6 --with-net-snmp --with-libcurl --wit
```

Diferentes componentes de Zabbix pueden compilarse con distintas bibliotecas criptográficas (por ejemplo, un server con *OpenSSL* y un agent con *GnuTLS*).

Attention:
Si planea utilizar claves precompartidas (PSK), considere utilizar las bibliotecas *GnuTLS* u *OpenSSL 1.1.0* (o posteriores) en los componentes de Zabbix que utilicen PSK. Las bibliotecas *GnuTLS* y *OpenSSL 1.1.0* admiten conjuntos de cifrado PSK con Perfect Forward Secrecy. Las versiones anteriores de la biblioteca *OpenSSL* (1.0.1 y 1.0.2c) también admiten PSK, pero los conjuntos de cifrado PSK disponibles no proporcionan Perfect Forward Secrecy.

Gestión del cifrado de las conexiones Las conexiones en Zabbix pueden utilizar:

- sin cifrado (predeterminado)
- cifrado basado en certificados RSA
- cifrado basado en PSK

Hay dos parámetros importantes que se utilizan para especificar el cifrado entre los componentes de Zabbix:

- TLSConnect: especifica qué cifrado se utilizará para las conexiones salientes (sin cifrado, PSK o certificado)
- TLSAccept: especifica qué tipos de conexiones se permiten para las conexiones entrantes (sin cifrado, PSK o certificado). Se pueden especificar uno o más valores.

`TLSConnect` se utiliza en los archivos de configuración del proxy de Zabbix (en modo activo, especifica únicamente las conexiones al server) y del agent de Zabbix (para las comprobaciones activas). En el frontend de Zabbix, el equivalente de `TLSConnect` es el campo *Conexiones al host* en *Recopilación de datos* > *Hosts* > <algún host> > pestaña *Cifrado* y el campo *Conexiones al proxy* en *Administración* > *Proxies* > <algún proxy> > pestaña *Cifrado*. Si falla el tipo de cifrado configurado para la conexión, no se probarán otros tipos de cifrado.

`TLSAccept` se utiliza en los archivos de configuración del proxy de Zabbix (en modo pasivo, especifica únicamente las conexiones desde el server) y del agent de Zabbix (para las comprobaciones pasivas). En el frontend de Zabbix, el equivalente de `TLSAccept` es el campo *Conexiones desde el host* en *Recopilación de datos* > *Hosts* > <algún host> > pestaña *Cifrado* y el campo *Conexiones desde el proxy* en *Administración* > *Proxies* > <algún proxy> > pestaña *Cifrado*.

Normalmente se configura un solo tipo de cifrado para las conexiones entrantes. Sin embargo, puede ser necesario cambiar el tipo de cifrado, por ejemplo, de sin cifrado a basado en certificados, con un tiempo de inactividad mínimo y posibilidad de reversión. Para ello:

- Establezca `TLSAccept=unencrypted,cert` en el archivo de configuración del agent y reinicie el agent de Zabbix.
- Pruebe la conexión con `zabbix_get` al agent utilizando el certificado. Si funciona, puede volver a configurar el cifrado de ese agent en el frontend de Zabbix, en *Recopilación de datos* > *Hosts* > <algún host> > pestaña *Cifrado*, estableciendo *Conexiones al host* en *Certificado*.
- Cuando se actualice la caché de configuración del server (y se actualice la configuración del proxy si el host está siendo monitorizado por un proxy), las conexiones a ese agent estarán cifradas.
- Si todo funciona según lo esperado, puede establecer `TLSAccept=cert` en el archivo de configuración del agent y reiniciar el agent de Zabbix. Ahora el agent aceptará únicamente conexiones cifradas basadas en certificados. Las conexiones sin cifrado y basadas en PSK serán rechazadas.

De forma similar, funciona en el server y el proxy. Si en la configuración del host del frontend de Zabbix *Conexiones desde el host* está establecido en "Certificado", solo se aceptarán conexiones cifradas basadas en certificados desde el agent (comprobaciones activas) y `zabbix\_sender` (items trapper).

Lo más probable es que configure las conexiones entrantes y salientes para utilizar el mismo tipo de cifrado o ningún cifrado. Sin embargo, técnicamente es posible configurarlas de forma asimétrica, por ejemplo, con cifrado basado en certificados para las conexiones entrantes y basado en PSK para las salientes.

La configuración del cifrado de cada host se muestra en el frontend de Zabbix, en *Recopilación de datos* > *Hosts*, en la columna *Cifrado del agent*. Por ejemplo:

Ejemplo	Conexiones al host	Conexiones permitidas desde el host	Conexiones rechazadas desde el host
NONE	Sin cifrado	Sin cifrado	Cifradas, basadas en certificados y basadas en PSK
CERT NONE PSK CERT	Cifradas, basadas en certificados	Cifradas, basadas en certificados	Sin cifrado y cifradas basadas en PSK
PSK NONE PSK CERT	Cifradas, basadas en PSK	Cifradas, basadas en PSK	Sin cifrado y cifradas basadas en certificados
PSK NONE PSK CERT	Cifradas, basadas en PSK	Sin cifrado y cifradas basadas en PSK	Cifradas basadas en certificados
CERT NONE PSK CERT	Cifradas, basadas en certificados	Sin cifrado, PSK o cifradas basadas en certificados	-

Attention:
Las conexiones no están cifradas de forma predeterminada. El cifrado debe configurarse individualmente para cada host y proxy.

zabbix_get y zabbix_sender con cifrado Consulte las páginas del manual de zabbix_get y zabbix_sender para usarlos con cifrado.

Ciphersuites Las ciphersuites se configuran de forma interna durante el inicio de Zabbix de manera predeterminada.

También se admiten ciphersuites configuradas por el usuario para GnuTLS y OpenSSL. Los usuarios pueden configurar ciphersuites según sus políticas de seguridad. El uso de esta función es opcional (las ciphersuites predeterminadas integradas siguen funcionando).

Para las bibliotecas criptográficas compiladas con la configuración predeterminada, las reglas integradas de Zabbix suelen dar como resultado las siguientes ciphersuites (en orden de mayor a menor prioridad):

Library	Certificate ciphersuites	PSK ciphersuites
GnuTLS 3.1.18	TLS_ECDHE_RSA_AES_128_GCM_SHA256 TLS_ECDHE_RSA_AES_128_CBC_SHA256 TLS_ECDHE_RSA_AES_128_CBC_SHA1 TLS_RSA_AES_128_GCM_SHA256 TLS_RSA_AES_128_CBC_SHA256 TLS_RSA_AES_128_CBC_SHA1	TLS_ECDHE_PSK_AES_128_CBC_SHA256 TLS_ECDHE_PSK_AES_128_CBC_SHA1 TLS_PSK_AES_128_GCM_SHA256 TLS_PSK_AES_128_CBC_SHA256 TLS_PSK_AES_128_CBC_SHA1
OpenSSL 1.0.2c	ECDHE-RSA-AES128-GCM-SHA256 ECDHE-RSA-AES128-SHA256 ECDHE-RSA-AES128-SHA AES128-GCM-SHA256 AES128-SHA256 AES128-SHA	PSK-AES128-CBC-SHA
OpenSSL 1.1.0	ECDHE-RSA-AES128-GCM-SHA256 ECDHE-RSA-AES128-SHA256 ECDHE-RSA-AES128-SHA AES128-GCM-SHA256 AES128-CCM8 AES128-CCM AES128-SHA256 AES128-SHA	ECDHE-PSK-AES128-CBC-SHA256 ECDHE-PSK-AES128-CBC-SHA PSK-AES128-GCM-SHA256 PSK-AES128-CCM8 PSK-AES128-CCM PSK-AES128-CBC-SHA256 PSK-AES128-CBC-SHA

Library	Certificate ciphersuites	PSK ciphersuites
OpenSSL 1.1.1d	TLS_AES_256_GCM_SHA384 TLS_CHACHA20_POLY1305_SHA256 TLS_AES_128_GCM_SHA256 ECDHE-RSA-AES128-GCM-SHA256 ECDHE-RSA-AES128-SHA256 ECDHE-RSA-AES128-SHA AES128-GCM-SHA256 AES128-CCM8 AES128-CCM AES128-SHA256 AES128-SHA	TLS_CHACHA20_POLY1305_SHA256 TLS_AES_128_GCM_SHA256 ECDHE-PSK-AES128-CBC-SHA256 ECDHE-PSK-AES128-CBC-SHA PSK-AES128-GCM-SHA256 PSK-AES128-CCM8 PSK-AES128-CCM PSK-AES128-CBC-SHA256 PSK-AES128-CBC-SHA

Cifrados configurados por el usuario Los criterios de selección de ciphersuite integrados pueden sobrescribirse con ciphersuites configuradas por el usuario.

> **Attention:**
> Las ciphersuites configuradas por el usuario es una función destinada a usuarios avanzados que entienden las ciphersuites TLS, su seguridad y las consecuencias de los errores, y que se sienten cómodos con la solución de problemas de TLS.

Los criterios de selección de ciphersuite integrados pueden sobrescribirse mediante los siguientes parámetros:

Override scope	Parameter	Value	Description
Selección de ciphersuite para certificados	TLSCipherCert13	Cadenas de cifrado de OpenSSL 1.1.1 válidas cipher strings para el protocolo TLS 1.3 (sus valores se pasan a la función de OpenSSL `SSL_CTX_set_ciphersuites()`).	Criterios de selección de ciphersuite basados en certificados para TLS 1.3 Solo OpenSSL 1.1.1 o posterior.
	TLSCipherCert	Cadenas de cifrado de OpenSSL válidas cipher strings para TLS 1.2 o cadenas de prioridad de GnuTLS válidas priority strings. Sus valores se pasan a las funciones `SSL_CTX_set_cipher_list()` o `gnutls_priority_init()`, respectivamente.	Criterios de selección de ciphersuite basados en certificados para TLS 1.2/1.3 (GnuTLS), TLS 1.2 (OpenSSL)
Selección de ciphersuite para PSK	TLSCipherPSK13	Cadenas de cifrado de OpenSSL 1.1.1 válidas cipher strings para el protocolo TLS 1.3 (sus valores se pasan a la función de OpenSSL `SSL_CTX_set_ciphersuites()`).	Criterios de selección de ciphersuite basados en PSK para TLS 1.3 Solo OpenSSL 1.1.1 o posterior.
	TLSCipherPSK	Cadenas de cifrado de OpenSSL válidas cipher strings para TLS 1.2 o cadenas de prioridad de GnuTLS válidas priority strings. Sus valores se pasan a las funciones `SSL_CTX_set_cipher_list()` o `gnutls_priority_init()`, respectivamente.	Criterios de selección de ciphersuite basados en PSK para TLS 1.2/1.3 (GnuTLS), TLS 1.2 (OpenSSL)
Lista combinada de ciphersuite para certificado y PSK	TLSCipherAll13	Cadenas de cifrado de OpenSSL 1.1.1 válidas cipher strings para el protocolo TLS 1.3 (sus valores se pasan a la función de OpenSSL `SSL_CTX_set_ciphersuites()`).	Criterios de selección de ciphersuite para TLS 1.3 Solo OpenSSL 1.1.1 o posterior.

Override scope	Parameter	Value	Description
	TLSCipherAll	Cadenas de cifrado de OpenSSL válidas cipher strings para TLS 1.2 o cadenas de prioridad de GnuTLS válidas priority strings. Sus valores se pasan a las funciones `SSL_CTX_set_cipher_list()` o `gnutls_priority_init()`, respectivamente.	Criterios de selección de ciphersuite para TLS 1.2/1.3 (GnuTLS), TLS 1.2 (OpenSSL)

Para sobrescribir la selección de ciphersuite en las utilidades zabbix_get y zabbix_sender, use los parámetros de línea de comandos:

- `--tls-cipher13`
- `--tls-cipher`

Los nuevos parámetros son opcionales. Si no se especifica un parámetro, se usa el valor predeterminado interno. Si un parámetro está definido, no puede estar vacío.

Si la configuración de un valor TLSCipher* en la biblioteca criptográfica falla, entonces el server, proxy o agent no se iniciará y se registrará un error.

Es importante entender cuándo es aplicable cada parámetro.

Conexiones salientes

El caso más simple es el de las conexiones salientes:

- Para conexiones salientes con certificado: use TLSCipherCert13 o TLSCipherCert
- Para conexiones salientes con PSK: use TLSCipherPSK13 o TLSCipherPSK
- En el caso de las utilidades zabbix_get y zabbix_sender, se pueden usar los parámetros de línea de comandos `--tls-cipher13` o `--tls-cipher` (el cifrado se especifica de forma inequívoca con un parámetro `--tls-connect`)

Conexiones entrantes

Es un poco más complicado con las conexiones entrantes porque las reglas son específicas para los componentes y la configuración.

Para Zabbix **agent**:

Configuración de conexión del agent	Configuración de cifrado
TLSConnect=cert	TLSCipherCert, TLSCipherCert13
TLSConnect=psk	TLSCipherPSK, TLSCipherPSK13
TLSAccept=cert	TLSCipherCert, TLSCipherCert13
TLSAccept=psk	TLSCipherPSK, TLSCipherPSK13
TLSAccept=cert,psk	TLSCipherAll, TLSCipherAll13

Para Zabbix **server** y **proxy**:

Configuración de conexión	Configuración de cifrado
Conexiones salientes usando PSK	TLSCipherPSK, TLSCipherPSK13
Conexiones entrantes usando certificados	TLSCipherAll, TLSCipherAll13
Conexiones entrantes usando PSK si el server no tiene certificado	TLSCipherPSK, TLSCipherPSK13
Conexiones entrantes usando PSK si el server tiene certificado	TLSCipherAll, TLSCipherAll13

Se puede observar algún patrón en las dos tablas anteriores:

- TLSCipherAll y TLSCipherAll13 solo pueden especificarse si se usa una lista combinada de conjuntos de cifrado basados en certificados **y** en PSK. Esto ocurre en dos casos: server (proxy) con un certificado configurado (los conjuntos de cifrado PSK siempre están configurados en el server, y en el proxy si la biblioteca criptográfica admite PSK), y agent configurado para aceptar conexiones entrantes basadas tanto en certificados como en PSK
- en otros casos, TLSCipherCert* y/o TLSCipherPSK* son suficientes

Las siguientes tablas muestran los valores predeterminados integrados de `TLSCipher*`. Pueden ser un buen punto de partida para sus propios valores personalizados.

Parameter	GnuTLS 3.6.12
TLSCipherCert	NONE:+VERS-TLS1.2:+ECDHE-RSA:+RSA:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL:+COMP-NULL:+SIGN-ALL:+CTYPE-X.509
TLSCipherPSK	NONE:+VERS-TLS1.2:+ECDHE-PSK:+PSK:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL:+COMP-NULL:+SIGN-ALL
TLSCipherAll	NONE:+VERS-TLS1.2:+ECDHE-RSA:+RSA:+ECDHE-PSK:+PSK:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL:+COMP-NULL:+SIGN-ALL:+CTYPE-X.509

Parameter	OpenSSL 1.1.1d [1]
TLSCipherCert13	
TLSCipherCert	EECDH+aRSA+AES128:RSA+aRSA+AES128
TLSCipherPSK13	TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
TLSCipherPSK	kECDHEPSK+AES128:kPSK+AES128
TLSCipherAll13	
TLSCipherAll	EECDH+aRSA+AES128:RSA+aRSA+AES128:kECDHEPSK+AES128:kPSK+AES128

[1] Los valores predeterminados son diferentes para versiones anteriores de OpenSSL (1.0.1, 1.0.2, 1.1.0), para LibreSSL y si OpenSSL se compila sin soporte para PSK.

Ejemplos de conjuntos de cifrado configurados por el usuario

Vea a continuación los siguientes ejemplos de conjuntos de cifrado configurados por el usuario:

- Prueba de cadenas de cifrado y permitir solo conjuntos de cifrado PFS
- Cambio de AES128 a AES256

Probar las cadenas de cifrado y permitir únicamente conjuntos de cifrado PFS

Para ver qué conjuntos de cifrado se han seleccionado, debe establecer `DebugLevel=4` en el archivo de configuración o utilizar la opción `-vv` para zabbix_sender.

Es posible que deba experimentar con los parámetros `TLSCipher*` antes de obtener los conjuntos de cifrado deseados. Resulta incómodo reiniciar el Zabbix server, proxy o agent varias veces solo para ajustar los parámetros `TLSCipher*`. Una opción más conveniente es utilizar zabbix_sender o el comando `openssl`. Veamos ambos métodos.

1. Uso de zabbix_sender.

Creemos un archivo de configuración de prueba, por ejemplo, `/home/zabbix/test.conf`, con la sintaxis de un archivo `zabbix_agentd.conf`:

```
Hostname=nonexisting
ServerActive=nonexisting

TLSConnect=cert
TLSCAFile=/home/zabbix/ca.crt
TLSCertFile=/home/zabbix/agent.crt
TLSKeyFile=/home/zabbix/agent.key
TLSPSKIdentity=nonexisting
TLSPSKFile=/home/zabbix/agent.psk
```

Para este ejemplo, necesita certificados válidos de CA y del agent, así como una PSK. Ajuste las rutas y los nombres de los archivos de certificados y PSK según su entorno.

Si no utiliza certificados y solo utiliza PSK, puede crear un archivo de prueba más sencillo:

```
Hostname=nonexisting
ServerActive=nonexisting

TLSConnect=psk
TLSPSKIdentity=nonexisting
TLSPSKFile=/home/zabbix/agentd.psk
```

Los conjuntos de cifrado seleccionados pueden verse ejecutando zabbix_sender (ejemplo compilado con OpenSSL 1.1.d):

```
$ zabbix_sender -vv -c /home/zabbix/test.conf -k nonexisting_item -o 1 2>&1 | grep ciphersuites
zabbix_sender [41271]: DEBUG: zbx_tls_init_child() certificate ciphersuites: TLS_AES_256_GCM_SHA384 TLS_
zabbix_sender [41271]: DEBUG: zbx_tls_init_child() PSK ciphersuites: TLS_CHACHA20_POLY1305_SHA256 TLS_AE
zabbix_sender [41271]: DEBUG: zbx_tls_init_child() certificate and PSK ciphersuites: TLS_AES_256_GCM_SHA
```

Aquí se muestran los conjuntos de cifrado seleccionados de forma predeterminada. Estos valores predeterminados se eligen para garantizar la interoperabilidad con los Zabbix agents que se ejecutan en sistemas con versiones anteriores de OpenSSL (desde la 1.0.1).

En sistemas más recientes, puede optar por reforzar la seguridad permitiendo solo unos pocos conjuntos de cifrado, por ejemplo, únicamente conjuntos de cifrado con PFS (Perfect Forward Secrecy, secreto perfecto hacia adelante). Configuremos los parámetros `TLSCipher*` para permitir solo conjuntos de cifrado compatibles con PFS.

Attention:
El resultado no será interoperable con sistemas que utilicen OpenSSL 1.0.1 y 1.0.2 si se utiliza PSK. El cifrado basado en certificados debería funcionar.

Añada dos líneas al archivo de configuración `test.conf`:

```
TLSCipherCert=EECDH+aRSA+AES128
TLSCipherPSK=kECDHEPSK+AES128
```

y vuelva a realizar la prueba:

```
$ zabbix_sender -vv -c /home/zabbix/test.conf -k nonexisting_item -o 1 2>&1 | grep ciphersuites
zabbix_sender [42892]: DEBUG: zbx_tls_init_child() certificate ciphersuites: TLS_AES_256_GCM_SHA384 TLS_
zabbix_sender [42892]: DEBUG: zbx_tls_init_child() PSK ciphersuites: TLS_CHACHA20_POLY1305_SHA256 TLS_AE
zabbix_sender [42892]: DEBUG: zbx_tls_init_child() certificate and PSK ciphersuites: TLS_AES_256_GCM_SHA
```

Las listas de «certificate ciphersuites» y «PSK ciphersuites» han cambiado: son más cortas que antes y contienen únicamente conjuntos de cifrado TLS 1.3 y conjuntos de cifrado TLS 1.2 ECDHE-*, como se esperaba.

2. TLSCipherAll y TLSCipherAll13 no se pueden probar con zabbix_sender; no afectan al valor «certificate and PSK ciphersuites» mostrado en el ejemplo anterior. Para ajustar TLSCipherAll y TLSCipherAll13, debe experimentar con el agent, proxy o server.

Por lo tanto, para permitir únicamente conjuntos de cifrado PFS, es posible que deba añadir hasta tres parámetros:

```
TLSCipherCert=EECDH+aRSA+AES128
TLSCipherPSK=kECDHEPSK+AES128
TLSCipherAll=EECDH+aRSA+AES128:kECDHEPSK+AES128
```

a `zabbix_agentd.conf`, `zabbix_proxy.conf` y `zabbix_server.conf` si cada uno tiene un certificado configurado y el agent también tiene una PSK.

Si su entorno de Zabbix utiliza únicamente cifrado basado en PSK y no utiliza certificados, solo necesita uno:

```
TLSCipherPSK=kECDHEPSK+AES128
```

Ahora que comprende cómo funciona, puede probar la selección de conjuntos de cifrado incluso fuera de Zabbix mediante el comando `openssl`. Probemos los tres valores de los parámetros `TLSCipher*`:

```
$ openssl ciphers EECDH+aRSA+AES128 | sed 's/:/ /g'
TLS_AES_256_GCM_SHA384 TLS_CHACHA20_POLY1305_SHA256 TLS_AES_128_GCM_SHA256 ECDHE-RSA-AES128-GCM-SHA256 E
$ openssl ciphers kECDHEPSK+AES128 | sed 's/:/ /g'
TLS_AES_256_GCM_SHA384 TLS_CHACHA20_POLY1305_SHA256 TLS_AES_128_GCM_SHA256 ECDHE-PSK-AES128-CBC-SHA256 E
$ openssl ciphers EECDH+aRSA+AES128:kECDHEPSK+AES128 | sed 's/:/ /g'
TLS_AES_256_GCM_SHA384 TLS_CHACHA20_POLY1305_SHA256 TLS_AES_128_GCM_SHA256 ECDHE-RSA-AES128-GCM-SHA256 E
```

Puede preferir `openssl ciphers` con la opción `-V` para obtener una salida más detallada:

```
$ openssl ciphers -V EECDH+aRSA+AES128:kECDHEPSK+AES128
           0x13,0x02 - TLS_AES_256_GCM_SHA384  TLSv1.3 Kx=any      Au=any  Enc=AESGCM(256) Mac=AEAD
           0x13,0x03 - TLS_CHACHA20_POLY1305_SHA256 TLSv1.3 Kx=any      Au=any  Enc=CHACHA20/POLY1305(256
           0x13,0x01 - TLS_AES_128_GCM_SHA256  TLSv1.3 Kx=any      Au=any  Enc=AESGCM(128) Mac=AEAD
           0xC0,0x2F - ECDHE-RSA-AES128-GCM-SHA256 TLSv1.2 Kx=ECDH     Au=RSA  Enc=AESGCM(128) Mac=AEAD
           0xC0,0x27 - ECDHE-RSA-AES128-SHA256 TLSv1.2 Kx=ECDH     Au=RSA  Enc=AES(128)  Mac=SHA256
           0xC0,0x13 - ECDHE-RSA-AES128-SHA    TLSv1 Kx=ECDH     Au=RSA  Enc=AES(128)  Mac=SHA1
           0xC0,0x37 - ECDHE-PSK-AES128-CBC-SHA256 TLSv1 Kx=ECDHEPSK Au=PSK  Enc=AES(128)  Mac=SHA256
           0xC0,0x35 - ECDHE-PSK-AES128-CBC-SHA TLSv1 Kx=ECDHEPSK Au=PSK  Enc=AES(128)  Mac=SHA1
```

De forma similar, puede probar las cadenas de prioridad para GnuTLS:

```
$ gnutls-cli -l --priority=NONE:+VERS-TLS1.2:+ECDHE-RSA:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+CURVE-A
Cipher suites for NONE:+VERS-TLS1.2:+ECDHE-RSA:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+CURVE-ALL:+COMP-
TLS_ECDHE_RSA_AES_128_GCM_SHA256                        0xc0, 0x2f      TLS1.2
TLS_ECDHE_RSA_AES_128_CBC_SHA256                        0xc0, 0x27      TLS1.2

Protocols: VERS-TLS1.2
Ciphers: AES-128-GCM, AES-128-CBC
MACs: AEAD, SHA256
Key Exchange Algorithms: ECDHE-RSA
Groups: GROUP-SECP256R1, GROUP-SECP384R1, GROUP-SECP521R1, GROUP-X25519, GROUP-X448, GROUP-FFDHE2048, GR
PK-signatures: SIGN-RSA-SHA256, SIGN-RSA-PSS-SHA256, SIGN-RSA-PSS-RSAE-SHA256, SIGN-ECDSA-SHA256, SIGN-E
```

Cambio de AES128 a AES256

Zabbix usa AES128 como valor predeterminado integrado para los datos. Supongamos que está usando certificados y desea cambiar a AES256, en OpenSSL 1.1.1.

Esto se puede lograr agregando los parámetros correspondientes en `zabbix_server.conf`:

```
TLSCAFile=/home/zabbix/ca.crt
TLSCertFile=/home/zabbix/server.crt
TLSKeyFile=/home/zabbix/server.key
TLSCipherCert13=TLS_AES_256_GCM_SHA384
TLSCipherCert=EECDH+aRSA+AES256:-SHA1:-SHA384
TLSCipherPSK13=TLS_CHACHA20_POLY1305_SHA256
TLSCipherPSK=kECDHEPSK+AES256:-SHA1
TLSCipherAll13=TLS_AES_256_GCM_SHA384
TLSCipherAll=EECDH+aRSA+AES256:-SHA1:-SHA384
```

Attention:
Aunque solo se usarán conjuntos de cifrado relacionados con certificados, los parámetros `TLSCipherPSK*` también se definen para evitar sus valores predeterminados, que incluyen cifrados menos seguros para una interoperabilidad más amplia. Los conjuntos de cifrado PSK no se pueden deshabilitar por completo en server/proxy.

Y en `zabbix_agentd.conf`:

```
TLSConnect=cert
TLSAccept=cert
TLSCAFile=/home/zabbix/ca.crt
TLSCertFile=/home/zabbix/agent.crt
TLSKeyFile=/home/zabbix/agent.key
TLSCipherCert13=TLS_AES_256_GCM_SHA384
TLSCipherCert=EECDH+aRSA+AES256:-SHA1:-SHA384
```

Solución de problemas de cifrado Recomendaciones generales:

- Empiece por entender qué componente actúa como cliente TLS y cuál actúa como servidor TLS en el caso problemático.
 Zabbix server, proxies y agents, según la interacción entre ellos, pueden funcionar tanto como servidores TLS como clientes TLS.
 Por ejemplo, Zabbix server al conectarse a un agent para una comprobación pasiva actúa como cliente TLS. El agent desempeña el papel de servidor TLS.
 Zabbix agent, al solicitar una lista de comprobaciones activas a un proxy, actúa como cliente TLS. El proxy desempeña el papel de servidor TLS.
 Las utilidades `zabbix_get` y `zabbix_sender` siempre actúan como clientes TLS.
- Zabbix utiliza autenticación mutua.
 Cada lado verifica a su par y puede rechazar la conexión.
 Por ejemplo, Zabbix server al conectarse a un agent puede cerrar la conexión inmediatamente si el certificado del agent no es válido. Y viceversa: Zabbix agent al aceptar una conexión del server puede cerrar la conexión si el server no es de confianza para el agent.
- Examine los archivos de registro en ambos lados: en el cliente TLS y en el servidor TLS.
 El lado que rechaza la conexión puede registrar un motivo preciso de por qué fue rechazada. El otro lado a menudo informa un error más general (por ejemplo, ”Connection closed by peer”, ”connection was non-properly terminated”).
- A veces, una configuración incorrecta del cifrado da lugar a mensajes de error confusos que no apuntan en absoluto a la causa real.
 En las subsecciones siguientes intentamos proporcionar una colección, lejos de ser exhaustiva, de mensajes y posibles causas que pueden ayudar en la resolución de problemas.
 Tenga en cuenta que distintos conjuntos de herramientas criptográficas (OpenSSL, GnuTLS) suelen producir mensajes de error diferentes en las mismas situaciones problemáticas.
 A veces, los mensajes de error dependen incluso de la combinación concreta de conjuntos de herramientas criptográficas en ambos lados.

Solución de problemas de tipo de conexión o problemas de permisos Server está configurado para conectarse con PSK a agent, pero agent solo acepta conexiones sin cifrar

En el registro de server o proxy (con *GnuTLS* 3.3.16)

```
Error al obtener el valor de agent: zbx_tls_connect(): gnutls_handshake() failed: \
    -110 La conexión TLS se terminó de forma incorrecta.
```

En el registro de server o proxy (con *OpenSSL* 1.0.2c)

```
Error al obtener el valor de agent: conexión TCP correcta, no se puede establecer TLS con [[127.0.0.1]:100
    Conexión cerrada por el par. Compruebe los tipos de conexión permitidos y los derechos de acceso
```

Un lado se conecta con certificado, pero el otro lado solo acepta PSK o viceversa

En cualquier registro (con *GnuTLS*):

```
failed to accept an incoming connection: from 127.0.0.1: zbx_tls_accept(): gnutls_handshake() failed:\
    -21 Could not negotiate a supported cipher suite.
```

En cualquier registro (con *OpenSSL* 1.0.2c):

```
failed to accept an incoming connection: from 127.0.0.1: TLS handshake returned error code 1:\
    file .\ssl\s3_srvr.c line 1411: error:1408A0C1:SSL routines:ssl3_get_client_hello:no shared cipher:\
    TLS write fatal alert "handshake failure"
```

Intentando usar Zabbix sender compilado con soporte TLS para enviar datos a Zabbix server/proxy compilado sin TLS En el log del lado de conexión:

Linux:

```
...In zbx_tls_init_child()
...Biblioteca OpenSSL (versión OpenSSL 1.1.1  11 Sep 2018) inicializada
...
...In zbx_tls_connect(): psk_identity:"PSK test sender"
...Fin de zbx_tls_connect():FAIL error:'connection closed by peer'
...error al enviar valor: TCP correcto, no se puede establecer TLS con [[localhost]:10051]: connection clo
```

Windows:

```
...Biblioteca OpenSSL (versión OpenSSL 1.1.1a  20 Nov 2018) inicializada
...
...In zbx_tls_connect(): psk_identity:"PSK test sender"
...zbx_psk_client_cb() solicitó la identidad PSK "PSK test sender"
...Fin de zbx_tls_connect():FAIL error:'SSL_connect() I/O error: [0x00000000] The operation completed succ
...error al enviar valor: TCP correcto, no se puede establecer TLS con [[192.168.1.2]:10051]: SSL_connect(
```

En el registro del lado receptor:

```
...no se pudo aceptar una conexión entrante: desde 127.0.0.1: el soporte para TLS no se compiló en la vers
```

Un lado se conecta con PSK, pero el otro lado usa LibreSSL o se ha compilado sin soporte de cifrado LibreSSL no admite PSK.

En el registro del lado que se conecta:

```
...TCP successful, cannot establish TLS to [[192.168.1.2]:10050]: SSL_connect() I/O error: [0] Success
```

En el registro del lado que acepta:

```
...failed to accept an incoming connection: from 192.168.1.2: support for PSK was not compiled in
```

En el frontend de Zabbix:

```
Get value from agent failed: TCP successful, cannot establish TLS to [[192.168.1.2]:10050]: SSL_connect()
```

Un lado se conecta con PSK, pero el otro lado usa OpenSSL con el soporte de PSK deshabilitado En el registro del lado que se conecta:

```
...TCP successful, cannot establish TLS to [[192.168.1.2]:10050]: SSL_connect() set result code to SSL_ERR
```

En el registro del lado que acepta:

```
...failed to accept an incoming connection: from 192.168.1.2: TLS handshake set result code to 1: file ssl
```

Cifrado mediante certificados

Descripción general Zabbix puede usar certificados RSA en formato PEM, firmados por una autoridad de certificación (CA) pública o interna.

La verificación del certificado se realiza frente a un certificado de CA preconfigurado. Opcionalmente, se pueden usar listas de revocación de certificados (CRL).

Cada componente de Zabbix puede tener configurado solo un certificado.

Para obtener más información sobre cómo configurar y operar una CA interna, generar y firmar solicitudes de certificado, y revocar certificados, consulte tutoriales como el OpenSSL PKI Tutorial v2.0.

Considere y pruebe cuidadosamente las extensiones de su certificado. Para más detalles, consulte Limitaciones al usar extensiones de certificado X.509 v3.

Parámetros de configuración de certificados Los siguientes parámetros de configuración son compatibles para configurar certificados en los componentes de Zabbix.

Parameter	Mandatory	Description
TLSCAFile	yes	Ruta completa de un archivo que contiene los certificados de la CA de nivel superior para la verificación del certificado del par. Si se utiliza una cadena de certificados con varios miembros, ordene los certificados con los certificados de las CA de nivel inferior primero, seguidos por los certificados de las CA de nivel superior. Los certificados de varias CA pueden incluirse en un solo archivo.
TLSCRLFile	no	Ruta completa de un archivo que contiene listas de revocación de certificados (CRL).
TLSCertFile	yes	Ruta completa de un archivo que contiene el certificado. Si se utiliza una cadena de certificados con varios miembros, ordene los certificados con el certificado del server, proxy o agent primero, seguidos por los certificados de las CA de nivel inferior y, al final, los certificados de las CA de nivel superior.
TLSKeyFile	yes	Ruta completa de un archivo que contiene la clave privada. Asegúrese de que este archivo solo sea legible por el usuario de Zabbix estableciendo los permisos de acceso adecuados.
TLSServerCertIssuer	no	Emisor permitido del certificado del server.
TLSServerCertSubject	no	Asunto permitido del certificado del server.

Ejemplos de configuración Después de configurar los certificados necesarios, configure los componentes de Zabbix para usar cifrado basado en certificados.

A continuación se detallan los pasos para configurar:

- Zabbix server
- Zabbix proxy
- Zabbix agent

Zabbix server

1. Prepare el archivo de certificado de CA.

Para verificar los certificados de los pares, Zabbix server debe tener acceso al archivo que contiene los certificados de CA raíz autofirmados de nivel superior. Por ejemplo, si se necesitan certificados de dos CA raíz independientes, colóquelos en un archivo en `/home/zabbix/zabbix_ca_file.crt`:

```
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 1 (0x1)
    Signature Algorithm: sha1WithRSAEncryption
        Issuer: DC=com, DC=zabbix, O=Zabbix SIA, OU=Development group, CN=Root1 CA
            ...
        Subject: DC=com, DC=zabbix, O=Zabbix SIA, OU=Development group, CN=Root1 CA
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
```

```
                Public-Key: (2048 bit)
            ...
        X509v3 extensions:
            X509v3 Key Usage: critical
                Certificate Sign, CRL Sign
            X509v3 Basic Constraints: critical
                CA:TRUE
            ...
-----BEGIN CERTIFICATE-----
MIID2jCCAsKgAwIBAgIBATANBgkqhkiG9w0BAQUFADB+MRMwEQYKCZImiZPyLGQB
....
9wEzdN8uTrqoyU78gi12npLj08LegRKjb5hFTVmO
-----END CERTIFICATE-----
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 1 (0x1)
    Signature Algorithm: sha1WithRSAEncryption
        Issuer: DC=com, DC=zabbix, O=Zabbix SIA, OU=Development group, CN=Root2 CA
            ...
        Subject: DC=com, DC=zabbix, O=Zabbix SIA, OU=Development group, CN=Root2 CA
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
            ....
        X509v3 extensions:
            X509v3 Key Usage: critical
                Certificate Sign, CRL Sign
            X509v3 Basic Constraints: critical
                CA:TRUE
            ....
-----BEGIN CERTIFICATE-----
MIID3DCCAsSgAwIBAgIBATANBgkqhkiG9w0BAQUFADB/MRMwEQYKCZImiZPyLGQB
...
vdGNYoSfvu41GQAR5Vj5FnRJRzv5XQOZ3B6894GY1zY=
-----END CERTIFICATE-----
```

2. Coloque el certificado o la cadena de certificados de Zabbix server en un archivo, por ejemplo, en `/home/zabbix/zabbix_server.crt`. El primer certificado es el certificado de Zabbix server, seguido del certificado de la CA intermedia:

```
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 1 (0x1)
    Signature Algorithm: sha1WithRSAEncryption
        Issuer: DC=com, DC=zabbix, O=Zabbix SIA, OU=Development group, CN=Signing CA
        ...
        Subject: DC=com, DC=zabbix, O=Zabbix SIA, OU=Development group, CN=Zabbix server
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
                ...
        X509v3 extensions:
            X509v3 Key Usage: critical
                Digital Signature, Key Encipherment
            X509v3 Basic Constraints:
                CA:FALSE
            ...
-----BEGIN CERTIFICATE-----
MIIECDCCAvCgAwIBAgIBATANBgkqhkiG9w0BAQUFADCBgTETMBEGCgmSJomT8ixk
...
h02u1GHiy46GI+xfR3LsPwFKlkTaaLaL/6aaoQ==
-----END CERTIFICATE-----
```

```
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 2 (0x2)
    Signature Algorithm: sha1WithRSAEncryption
        Issuer: DC=com, DC=zabbix, O=Zabbix SIA, OU=Development group, CN=Root1 CA
        ...
        Subject: DC=com, DC=zabbix, O=Zabbix SIA, OU=Development group, CN=Signing CA
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
            ...
        X509v3 extensions:
            X509v3 Key Usage: critical
                Certificate Sign, CRL Sign
            X509v3 Basic Constraints: critical
                CA:TRUE, pathlen:0
        ...
-----BEGIN CERTIFICATE-----
MIID4TCCAsmgAwIBAgIBAjANBgkqhkiG9w0BAQUFADB+MRMwEQYKCZImiZPyLGQB
...
dyCeWnvL7u5sd6ffo8iRny0QzbHKmQt/wUtcVIvWXdMIFJMOHw==
-----END CERTIFICATE-----
```

Note:
Utilice únicamente los atributos mencionados anteriormente para los certificados de cliente y de server, a fin de evitar afectar al proceso de verificación de certificados. Por ejemplo, OpenSSL podría no establecer una conexión cifrada si se utilizan las extensiones *X509v3 Subject Alternative Name* o *Netscape Cert Type*. Para obtener más información, consulte Limitaciones del uso de extensiones de certificados X.509 v3.

3. Coloque la clave privada de Zabbix server en un archivo, por ejemplo, en `/home/zabbix/zabbix_server.key`:

```
-----BEGIN PRIVATE KEY-----
MIIEwAIBADANBgkqhkiG9w0BAQEFAASCBKowggSmAgEAAoIBAQC9tIXIJoVnNXDl
...
IJLkhbybBYEf47MLhffWa7XvZTY=
-----END PRIVATE KEY-----
```

4. Edite los parámetros de configuración de TLS en el archivo de configuración de server de Zabbix:

```
TLSCAFile=/home/zabbix/zabbix_ca_file.crt
TLSCertFile=/home/zabbix/zabbix_server.crt
TLSKeyFile=/home/zabbix/zabbix_server.key
```

Zabbix proxy

1. Prepare los archivos con los certificados de CA de nivel superior, el certificado o la cadena de certificados del Zabbix proxy y la clave privada, tal como se describe en la sección Zabbix server. A continuación, edite los parámetros `TLSCAFile`, `TLSCertFile` y `TLSKeyFile` en el archivo de configuración del proxy de Zabbix según corresponda.

2. Edite los parámetros TLS adicionales en el archivo de configuración del proxy de Zabbix:

- Para un proxy activo: `TLSConnect=cert`
- Para un proxy pasivo: `TLSAccept=cert`

Note:
Para mejorar la seguridad del proxy, también puede establecer los parámetros `TLSServerCertIssuer` y `TLSServerCertSubject`. Para obtener más información, consulte Restringir el emisor y el asunto del certificado permitidos.

Los parámetros TLS del archivo de configuración final del proxy pueden tener el siguiente aspecto:

```
TLSConnect=cert
TLSAccept=cert
TLSCAFile=/home/zabbix/zabbix_ca_file.crt
TLSServerCertIssuer=CN=Signing CA,OU=Development group,O=Zabbix SIA,DC=zabbix,DC=com
```

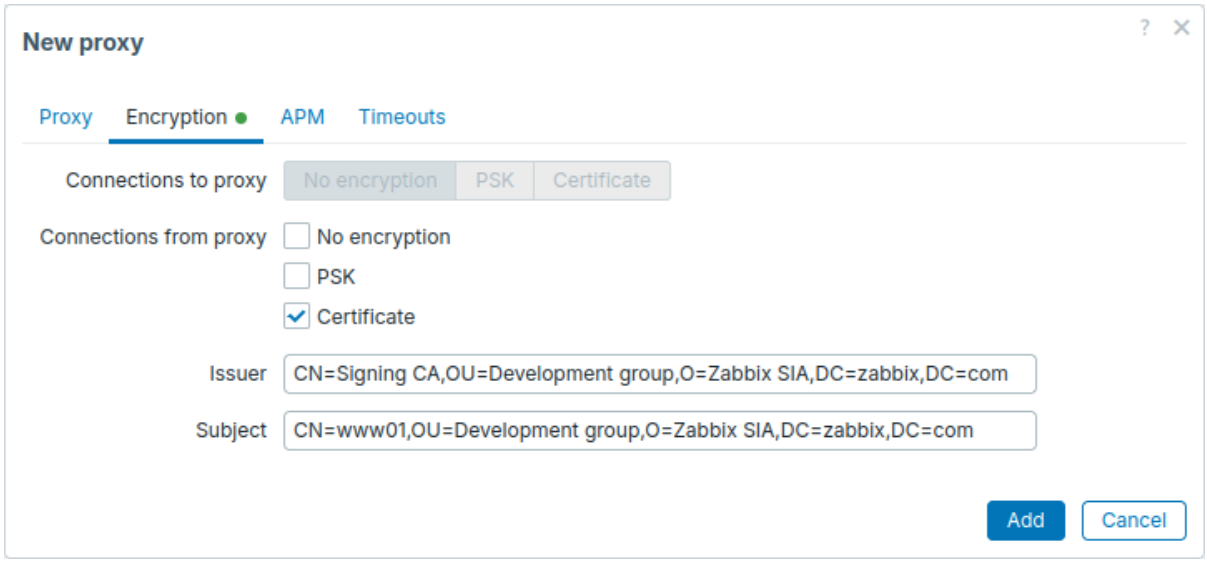
```
TLSServerCertSubject=CN=Zabbix server,OU=Development group,O=Zabbix SIA,DC=zabbix,DC=com
TLSCertFile=/home/zabbix/zabbix_proxy.crt
TLSKeyFile=/home/zabbix/zabbix_proxy.key
```

3. Configure el cifrado para este proxy en el frontend de Zabbix:

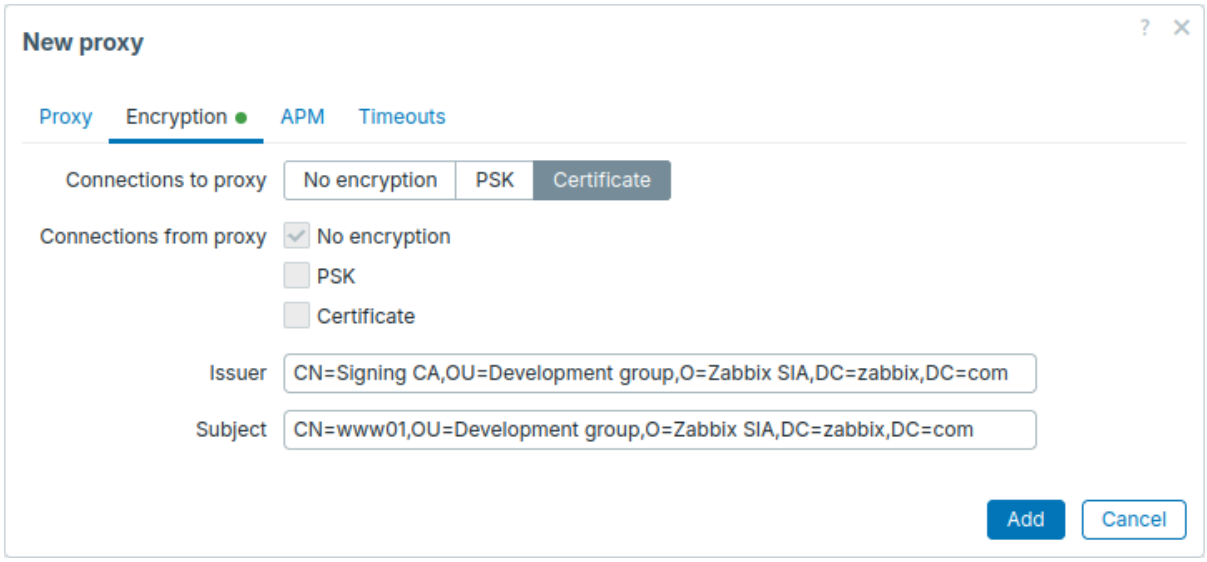
- Vaya a: *Administración* > *Proxies*.
- Seleccione el proxy y haga clic en la pestaña *Cifrado*.

En los ejemplos siguientes, los campos *Emisor* y *Asunto* están rellenados. Para obtener más información sobre por qué y cómo utilizar estos campos, consulte Restringir el emisor y el asunto del certificado permitidos.

Para un proxy activo:



Para un proxy pasivo:



Zabbix agent

1. Prepare los archivos con los certificados CA de nivel superior, el certificado o la cadena de certificados del Zabbix agent y la clave privada, tal como se describe en la sección Zabbix server. A continuación, edite los parámetros `TLSCAFile`, `TLSCertFile` y `TLSKeyFile` del archivo de configuración del agent de Zabbix según corresponda.

2. Edite los parámetros TLS adicionales en el archivo de configuración del agent de Zabbix:

- Para un agent activo: `TLSConnect=cert`
- Para un agent pasivo: `TLSAccept=cert`

Note:
Para mejorar la seguridad del agent, puede establecer los parámetros `TLSServerCertIssuer` y `TLSServerCertSubject`. Para obtener más información, consulte Restringir el emisor y el asunto de certificado permitidos.

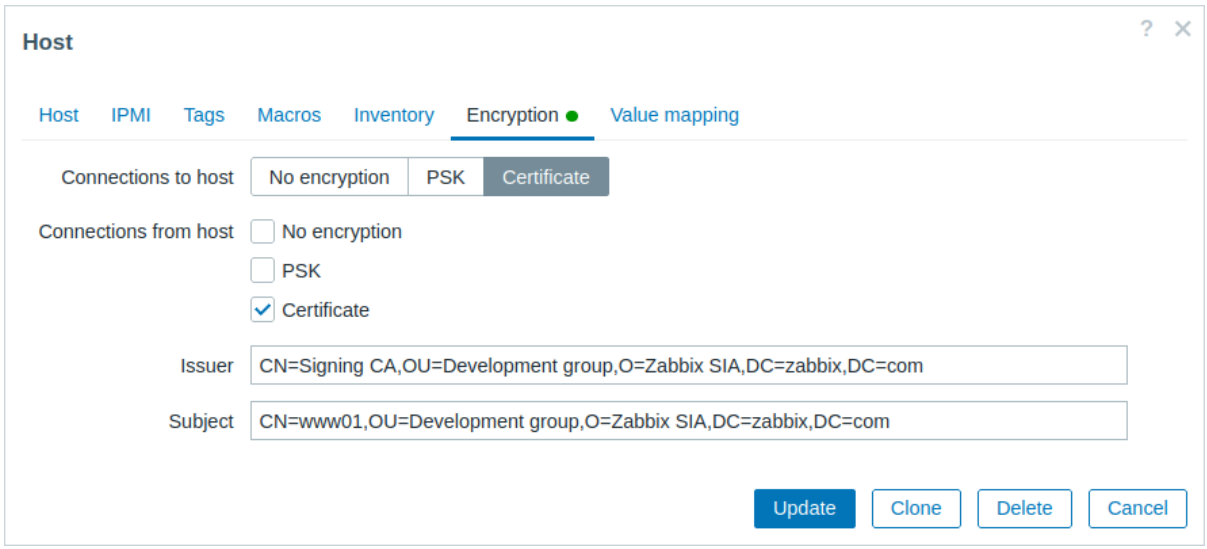
Los parámetros TLS del archivo de configuración final del agent pueden tener el siguiente aspecto. Tenga en cuenta que el ejemplo supone que el host está siendo monitorizado por un proxy, por lo que este se especifica como el Subject del certificado:

```
TLSConnect=cert
TLSAccept=cert
TLSCAFile=/home/zabbix/zabbix_ca_file.crt
TLSServerCertIssuer=CN=Signing CA,OU=Development group,O=Zabbix SIA,DC=zabbix,DC=com
TLSServerCertSubject=CN=Zabbix proxy,OU=Development group,O=Zabbix SIA,DC=zabbix,DC=com
TLSCertFile=/home/zabbix/zabbix_agentd.crt
TLSKeyFile=/home/zabbix/zabbix_agentd.key
```

3. Configure el cifrado en el frontend de Zabbix para el host monitorizado por este agent.

- Vaya a: *Recopilación de datos* > *Hosts*.
- Seleccione el host y haga clic en la pestaña *Cifrado*.

En el ejemplo siguiente, los campos *Issuer* y *Subject* están rellenados. Para obtener más información sobre por qué y cómo utilizar estos campos, consulte Restringir el emisor y el asunto de certificado permitidos.



Zabbix web service

1. Prepare los archivos con los certificados de CA de nivel superior, el certificado o la cadena de certificados del Zabbix web service y la clave privada, tal como se describe en la sección Zabbix server. A continuación, edite los parámetros `TLSCAFile`, `TLSCertFile` y `TLSKeyFile` en el archivo de configuración del web service de Zabbix según corresponda.

2. Edite un parámetro TLS adicional en el archivo de configuración del web service de Zabbix: `TLSAccept=cert`

Los parámetros TLS del archivo de configuración final del web service pueden tener el siguiente aspecto:

```
TLSAccept=cert
TLSCAFile=/home/zabbix/zabbix_ca_file.crt
TLSCertFile=/home/zabbix/zabbix_web_service.crt
TLSKeyFile=/home/zabbix/zabbix_web_service.key
```

3. Configure Zabbix server para conectarse al Zabbix web service configurado con TLS editando el parámetro `WebServiceURL` en el archivo de configuración del server de Zabbix:

```
WebServiceURL=https://example.com:443/report
```

Restricción del emisor y el sujeto de certificado permitidos Cuando dos componentes de Zabbix (por ejemplo, el server y el agent) establecen una conexión TLS, validan mutuamente sus certificados. Si el certificado del par está firmado por una CA de confianza (con un certificado de nivel superior preconfigurado en `TLSCAFile`), es válido, no ha caducado y supera las demás comprobaciones, la comunicación entre los componentes puede continuar. En este caso sencillo, no se verifican el emisor ni el sujeto del certificado.

Sin embargo, esto supone un riesgo: cualquiera que tenga un certificado válido puede suplantar la identidad de cualquier otra entidad (por ejemplo, un certificado de host podría utilizarse para suplantar a un server). Aunque esto puede ser aceptable en entornos pequeños, donde los certificados están firmados por una CA interna dedicada y el riesgo de suplantación es bajo, puede no ser suficiente en entornos más grandes o con mayores requisitos de seguridad.

Si la CA de nivel superior emite certificados que no deberían ser aceptados por Zabbix, o si desea reducir el riesgo de suplantación, puede restringir los certificados permitidos especificando su emisor y sujeto.

Por ejemplo, en el archivo de configuración del proxy de Zabbix, podría especificar:

```
TLSServerCertIssuer=CN=Signing CA,OU=Development group,O=Zabbix SIA,DC=zabbix,DC=com
TLSServerCertSubject=CN=Zabbix server,OU=Development group,O=Zabbix SIA,DC=zabbix,DC=com
```

Con esta configuración, un proxy activo no se comunicará con un server de Zabbix cuyo certificado tenga un emisor o sujeto diferente. Del mismo modo, un proxy pasivo no aceptará solicitudes de dicho server.

Reglas para la coincidencia de cadenas `Issuer` y `Subject`

Las reglas para la coincidencia de cadenas `Issuer` y `Subject` son las siguientes:

- Las cadenas `Issuer` y `Subject` se comprueban de forma independiente. Ambas son opcionales.
- Una cadena no especificada significa que se acepta cualquier cadena.
- Las cadenas se comparan *tal cual* y deben coincidir exactamente.
- Se admiten caracteres UTF-8. Sin embargo, no se admiten comodines (*) ni expresiones regulares.
- Se implementan los siguientes requisitos de RFC 4514: los caracteres que requieren escape (con una barra invertida '\', U+005C):
  - en cualquier parte de la cadena: '"' (U+0022), '+' (U+002B), ',' (U+002C), ';' (U+003B), '<' (U+003C), '>' (U+003E), '\\' (U+005C);
  - al principio de la cadena: espacio (' ', U+0020) o signo de número ('#', U+0023);
  - al final de la cadena: espacio (' ', U+0020).
- Los caracteres nulos (U+0000) no son compatibles. Si se encuentra un carácter nulo, la coincidencia fallará.
- Los estándares RFC 4517 y RFC 4518 no son compatibles.

Por ejemplo, si las cadenas de organización (O) de `Issuer` y `Subject` contienen espacios finales y la cadena de unidad organizativa (OU) de `Subject` contiene comillas dobles, estos caracteres deben escaparse:

```
TLSServerCertIssuer=CN=Signing CA,OU=Development head,O=\ Example SIA\ ,DC=example,DC=com
TLSServerCertSubject=CN=Zabbix server,OU=Development group \"5\",O=\ Example SIA\ ,DC=example,DC=com
```

Orden y formato de los campos

Zabbix sigue las recomendaciones de RFC 4514, que especifica un orden "inverso" para estos campos, comenzando con los campos de nivel más bajo (CN), continuando con los campos de nivel medio (OU, O) y concluyendo con los campos de nivel más alto (DC).

```
TLSServerCertIssuer=CN=Signing CA,OU=Development group,O=Zabbix SIA,DC=zabbix,DC=com
TLSServerCertSubject=CN=Zabbix proxy,OU=Development group,O=Zabbix SIA,DC=zabbix,DC=com
```

En cambio, OpenSSL muestra de forma predeterminada las cadenas `Issuer` y `Subject` en orden de nivel superior a nivel inferior. En el siguiente ejemplo, los campos `Issuer` y `Subject` comienzan con el nivel superior (DC) y terminan con el campo de nivel inferior (CN). El formato con espacios y separadores de campos también varía según las opciones utilizadas, por lo que no coincidirá con el formato requerido por Zabbix.

```
$ openssl x509 -noout -in /home/zabbix/zabbix_proxy.crt -issuer -subject
issuer= /DC=com/DC=zabbix/O=Zabbix SIA/OU=Development group/CN=Signing CA
subject= /DC=com/DC=zabbix/O=Zabbix SIA/OU=Development group/CN=Zabbix proxy

$ openssl x509 -noout -text -in /home/zabbix/zabbix_proxy.crt
Certificate:
    ...
        Issuer: DC=com, DC=zabbix, O=Zabbix SIA, OU=Development group, CN=Signing CA
        ...
        Subject: DC=com, DC=zabbix, O=Zabbix SIA, OU=Development group, CN=Zabbix proxy
```

Para formatear correctamente las cadenas *Issuer* y *Subject* para Zabbix, invoque OpenSSL con las siguientes opciones:

```
$ openssl x509 -noout -issuer -subject \
    -nameopt esc_2253,esc_ctrl,utf8,dump_nostr,dump_unknown,dump_der,sep_comma_plus,dn_rev,sname\
    -in /home/zabbix/zabbix_proxy.crt
```

La salida estará entonces en orden inverso, separada por comas y será utilizable en los archivos de configuración de Zabbix y en el frontend:

```
issuer=CN=Signing CA,OU=Development group,O=Zabbix SIA,DC=zabbix,DC=com
subject=CN=Zabbix proxy,OU=Development group,O=Zabbix SIA,DC=zabbix,DC=com
```

Limitaciones en el uso de extensiones de certificados X.509 v3 Al implementar certificados X.509 v3 en Zabbix, es posible que algunas extensiones no estén totalmente admitidas o que provoquen un comportamiento inconsistente.

Extensión Subject Alternative Name

Zabbix no admite la extensión *Subject Alternative Name*, que se utiliza para especificar nombres DNS alternativos, como direcciones IP o direcciones de correo electrónico. Zabbix solo puede validar el valor del campo *Subject* del certificado (consulte Restricting Allowed Certificate Issuer and Subject). Si los certificados incluyen el campo `subjectAltName`, el resultado de la validación del certificado puede variar según los conjuntos de herramientas criptográficas específicos utilizados para compilar los componentes de Zabbix. Como resultado, Zabbix puede aceptar o rechazar certificados en función de estas combinaciones.

Extensión Extended Key Usage

Zabbix admite la extensión *Extended Key Usage*. Sin embargo, si se utiliza, por lo general es necesario especificar tanto el atributo *clientAuth* (para la autenticación de cliente WWW TLS) como el atributo *serverAuth* (para la autenticación de servidor WWW TLS). Por ejemplo:

- En las comprobaciones pasivas, donde el agent de Zabbix actúa como un servidor TLS, el atributo *serverAuth* debe incluirse en el certificado del agent.
- En las comprobaciones activas, donde el agent actúa como un cliente TLS, el atributo *clientAuth* debe incluirse en el certificado del agent.

Aunque GnuTLS puede emitir una advertencia por infracciones del uso de claves, normalmente permite que la comunicación continúe a pesar de estas advertencias.

Extensión Name Constraints

La compatibilidad con la extensión *Name Constraints* varía entre los conjuntos de herramientas criptográficas. Asegúrese de que el conjunto de herramientas elegido admita esta extensión. Esta extensión puede impedir que Zabbix cargue certificados CA si esta sección está marcada como crítica, según el conjunto de herramientas específico en uso.

Listas de revocación de certificados (CRL) Si un certificado se ve comprometido, la Autoridad de Certificación (CA) puede revocarlo incluyéndolo en una Lista de revocación de certificados (CRL). Las CRL se administran mediante archivos de configuración y se pueden especificar usando el parámetro `TLSCRLFile` en los archivos de configuración de server, proxy y agent. Por ejemplo:

```
TLSCRLFile=/home/zabbix/zabbix_crl_file.crt
```

En este caso, `zabbix_crl_file.crt` puede contener CRL de varias CA y podría verse así:

```
-----BEGIN X509 CRL-----
MIIB/DCB5QIBATANBgkqhkiG9w0BAQUFADCBgTETMBEGCgmSJomT8ixkARkWA2Nv
...
treZeUPjb7LSmZ3K2hpbZN7SoOZcAoHQ3GWd9npuctg=
-----END X509 CRL-----
-----BEGIN X509 CRL-----
MIIB+TCB4gIBATANBgkqhkiG9w0BAQUFADB/MRMwEQYKCZImiZPyLGQBGRYDY29t
...
CAEebS2CND3ShBedZ8YSil59O6JvaDP61lR5lNs=
-----END X509 CRL-----
```

El archivo CRL se carga solo cuando Zabbix se inicia. Para actualizar la CRL, reinicie Zabbix.

> **Attention:**
> Si los componentes de Zabbix se compilan con OpenSSL y se usan CRL, asegúrese de que cada CA de nivel superior e intermedia en las cadenas de certificados tenga una CRL correspondiente (aunque esté vacía) incluida en `TLSCRLFile`.

Solución de problemas de certificados OpenSSL usado con CRL y para algún CA en la cadena de certificados cuya CRL no está incluida en `TLSCRLFile`

En el registro del server TLS en caso de un par *OpenSSL*:

```
failed to accept an incoming connection: from 127.0.0.1: TLS handshake with 127.0.0.1 returned error code
    file s3_srvr.c line 3251: error:14089086: SSL routines:ssl3_get_client_certificate:certificate verify
```

```
    TLS write fatal alert "unknown CA"
```

En el registro del server TLS en caso de un par *GnuTLS*:

```
failed to accept an incoming connection: from 127.0.0.1: TLS handshake with 127.0.0.1 returned error code
    file rsa_pk1.c line 103: error:0407006A: rsa routines:RSA_padding_check_PKCS1_type_1:\
    block type is not 01 file rsa_eay.c line 705: error:04067072: rsa routines:RSA_EAY_PUBLIC_DECRYPT:padd
```

CRL caducada o que caduca durante la operación del server

OpenSSL, en el log del server:

- antes de la caducidad:

```
cannot connect to proxy "proxy-openssl-1.0.1e": TCP successful, cannot establish TLS to [[127.0.0.1]:20004
    SSL_connect() returned SSL_ERROR_SSL: file s3_clnt.c line 1253: error:14090086:\
    SSL routines:ssl3_get_server_certificate:certificate verify failed:\
    TLS write fatal alert "certificate revoked"
```

- después de la caducidad:

```
cannot connect to proxy "proxy-openssl-1.0.1e": TCP successful, cannot establish TLS to [[127.0.0.1]:20004
    SSL_connect() returned SSL_ERROR_SSL: file s3_clnt.c line 1253: error:14090086:\
    SSL routines:ssl3_get_server_certificate:certificate verify failed:\
    TLS write fatal alert "certificate expired"
```

El punto aquí es que, con una CRL válida, un certificado revocado se informa como ”certificate revoked”. Cuando la CRL caduca, el mensaje de error cambia a ”certificate expired”, lo cual es bastante engañoso.

GnuTLS, en el log del server:

- antes y después de la caducidad, lo mismo:

```
cannot connect to proxy "proxy-openssl-1.0.1e": TCP successful, cannot establish TLS to [[127.0.0.1]:20004
      invalid peer certificate: The certificate is NOT trusted. The certificate chain is revoked.
```

Certificado autofirmado, CA desconocida

OpenSSL, en el log:

```
error:'self signed certificate: SSL_connect() set result code to SSL_ERROR_SSL: file ../ssl/statem/statem_
      line 1924: error:1416F086:SSL routines:tls_process_server_certificate:certificate verify failed:\
      TLS write fatal alert "unknown CA"'
```

Esto se observó cuando el certificado del server, por error, tenía la misma cadena de Issuer y Subject, aunque estaba firmado por una CA. Issuer y Subject son iguales en el certificado CA de nivel superior, pero no pueden ser iguales en el certificado del server. (Lo mismo aplica a los certificados de proxy y agent.)

Para comprobar si un certificado contiene las mismas entradas de Issuer y Subject, ejecute:

```
openssl x509 -in <yourcertificate.crt> -noout -text
```

Es aceptable que el certificado raíz (de nivel superior) tenga valores idénticos para Issuer y Subject.

Cifrado mediante claves precompartidas

Descripción general Cada clave precompartida (PSK) en Zabbix consta en realidad de un par de:

- cadena de identidad de PSK no secreta,
- valor de cadena de PSK secreto.

La cadena de identidad de PSK es una cadena UTF-8 no vacía. Por ejemplo, ”PSK ID 001 Zabbix agentd”. Es un nombre único mediante el cual los componentes de Zabbix hacen referencia a esta PSK específica. No incluya información confidencial en la cadena de identidad de PSK, ya que se transmite por la red sin cifrar.

El valor de PSK es una cadena de dígitos hexadecimales difícil de adivinar, por ejemplo, e560cb0d918d26d31b4f642181f5f570ad89a39093

> **Warning:**
> Nunca utilice en entornos de producción los valores de PSK de ejemplo proporcionados en la documentación. En su lugar, genere una PSK única y criptográficamente segura.

Límites de tamaño Existen límites de tamaño para la identidad y el valor de PSK en Zabbix; en algunos casos, una biblioteca criptográfica puede tener un límite inferior:

Componente	Tamaño máximo de la identidad PSK	Tamaño mínimo del valor PSK	Tamaño máximo del valor PSK
Zabbix	128 caracteres UTF-8	128 bits (PSK de 16 bytes, introducida como 32 dígitos hexadecimales)	2048 bits (PSK de 256 bytes, introducida como 512 dígitos hexadecimales)
GnuTLS	128 bytes (puede incluir caracteres UTF-8)	-	2048 bits (PSK de 256 bytes, introducida como 512 dígitos hexadecimales)
OpenSSL 1.0.x, 1.1.0	127 bytes (puede incluir caracteres UTF-8)	-	2048 bits (PSK de 256 bytes, introducida como 512 dígitos hexadecimales)
OpenSSL 1.1.1	127 bytes (puede incluir caracteres UTF-8)	-	512 bits (PSK de 64 bytes, introducida como 128 dígitos hexadecimales)
OpenSSL 1.1.1a y posteriores	127 bytes (puede incluir caracteres UTF-8)	-	2048 bits (PSK de 256 bytes, introducida como 512 dígitos hexadecimales)

Attention:
El frontend de Zabbix permite configurar una cadena de identidad PSK de hasta 128 caracteres y una PSK de hasta 2048 bits, independientemente de las bibliotecas criptográficas utilizadas.
 Si algunos componentes de Zabbix admiten límites inferiores, es responsabilidad del usuario configurar la identidad y el valor de PSK con una longitud permitida para dichos componentes.
 Superar los límites de longitud provoca fallos de comunicación entre los componentes de Zabbix.

Antes de que el server de Zabbix se conecte al agent mediante PSK, el server busca en la base de datos la identidad y el valor de PSK configurados para ese agent (en realidad, en la caché de configuración). Al recibir una conexión, el agent utiliza la identidad y el valor de PSK de su archivo de configuración. Si ambas partes tienen la misma cadena de identidad PSK y el mismo valor de PSK, la conexión puede realizarse correctamente.

Attention:
Cada identidad PSK debe estar asociada a un único valor. Es responsabilidad del usuario asegurarse de que no haya dos PSK con la misma cadena de identidad, pero con valores diferentes. No hacerlo puede provocar errores impredecibles o interrupciones de la comunicación entre los componentes de Zabbix que utilizan PSK con esta cadena de identidad PSK.

Generación de PSK Por ejemplo, se puede generar una PSK de 256 bits (32 bytes) mediante los siguientes comandos:

- con *OpenSSL*:

```
$ openssl rand -hex 32
af8ced32dfe8714e548694e2d29e1a14ba6fa13f216cb35c19d0feb1084b0429
```

- con *GnuTLS*:

```
$ psktool -u psk_identity -p database.psk -s 32
Generating a random key for user 'psk_identity'
Key stored to database.psk

$ cat database.psk
psk_identity:9b8eafedfaae00cece62e85d5f4792c7d9c9bcc851b23216a1d300311cc4f7cb
```

Tenga en cuenta que el comando "psktool" anterior genera un archivo de base de datos con una identidad PSK y su PSK asociada. Zabbix espera únicamente una PSK en el archivo PSK, por lo que la cadena de identidad y los dos puntos (':') deben eliminarse del archivo.

Configuración de PSK para la comunicación entre el server y el agent (ejemplo) En el host del agent, escriba el valor de PSK generado anteriormente en un archivo, por ejemplo, `/home/zabbix/zabbix_agentd.psk`. El archivo debe contener el valor de PSK en la primera línea.

Establezca los permisos de acceso al archivo de PSK: solo el usuario de Zabbix debe poder leerlo.

Edite los parámetros TLS en el archivo de configuración del agent de Zabbix (`zabbix_agentd.conf`). Por ejemplo, establezca:

```
TLSConnect=psk
TLSAccept=psk
TLSPSKFile=/home/zabbix/zabbix_agentd.psk
TLSPSKIdentity=PSK 001
```

El agent se conectará al server (comprobaciones activas) y aceptará del server y de `zabbix_get` únicamente conexiones que utilicen PSK. La identidad de PSK será "PSK 001".

Reinicie el agent. Ahora puede probar la conexión mediante `zabbix_get`, por ejemplo:

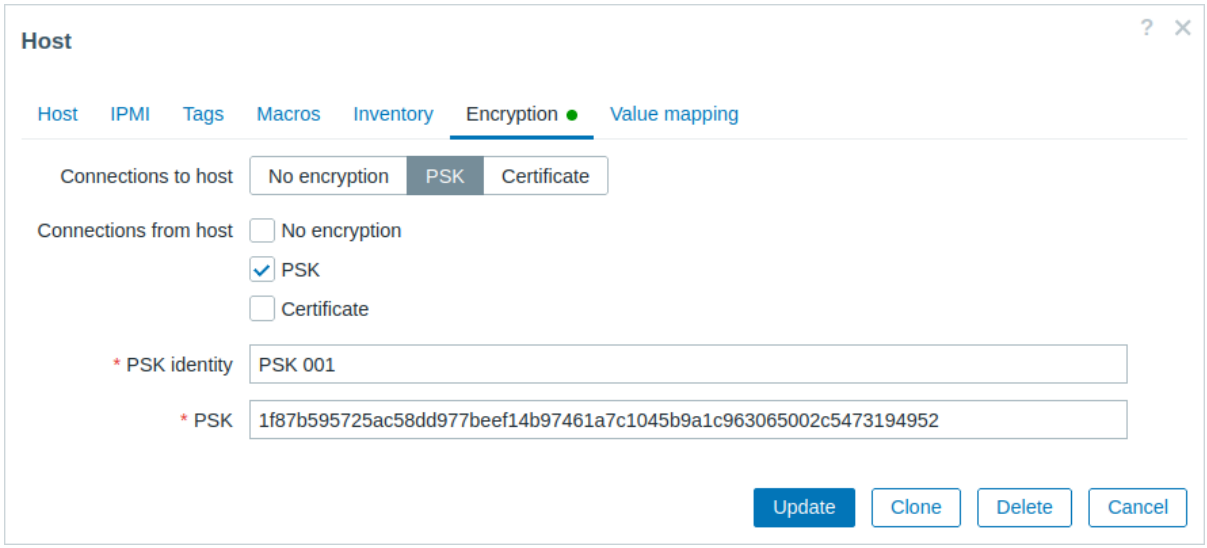
```
zabbix_get -s 127.0.0.1 -k "system.cpu.load[all,avg1]" --tls-connect=psk --tls-psk-identity="PSK 001" --tl
```

(Para minimizar el tiempo de inactividad, consulte cómo cambiar el tipo de conexión en Gestión del cifrado de conexiones).

Configure el cifrado PSK para este agent en el frontend de Zabbix:

1. Vaya a: *Recopilación de datos* > *Hosts*.
2. Seleccione el host y haga clic en la pestaña *Cifrado*.
3. Configure los detalles del cifrado PSK.

Ejemplo:



Todos los campos de entrada obligatorios están marcados con un asterisco rojo.

Cuando la caché de configuración se sincronice con la base de datos, las nuevas conexiones utilizarán PSK. Compruebe los archivos de registro del server y del agent para ver si hay mensajes de error.

Configuración de PSK para la comunicación entre el server y un proxy activo (ejemplo) En el proxy, escriba el valor de PSK en un archivo, por ejemplo, `/home/zabbix/zabbix_proxy.psk`. El archivo debe contener la PSK en la primera cadena de texto, por ejemplo:

```
e560cb0d918d26d31b4f642181f5f570ad89a390931102e5391d08327ba434e9
```

Establezca los permisos de acceso al archivo de PSK; solo el usuario de Zabbix debe poder leerlo.

Edite los parámetros TLS en el archivo de configuración del proxy de Zabbix (`zabbix_proxy.conf`). Por ejemplo, establezca:

```
TLSConnect=psk
TLSPSKFile=/home/zabbix/zabbix_proxy.psk
TLSPSKIdentity=PSK 002
```

El proxy se conectará al server mediante PSK. La identidad de PSK será "PSK 002".

(Para minimizar el tiempo de inactividad, consulte cómo cambiar el tipo de conexión en Gestión del cifrado de conexiones).

Configure la PSK para este proxy en el frontend de Zabbix. Vaya a *Administración* > *Proxies*, seleccione el proxy y vaya a la pestaña *Cifrado*. En "Conexiones desde el proxy", marque PSK. Pegue "PSK 002" en el campo "Identidad de PSK" y "e560cb0d918d26d31b4f642181f5f570ad89a390931102e5391d08327ba434e9" en el campo "PSK". Haga clic en "Actualizar".

Reinicie el proxy. Comenzará a utilizar conexiones cifradas basadas en PSK con el server. Compruebe los archivos de registro del server y del proxy para detectar mensajes de error.

Para un proxy pasivo, el procedimiento es muy similar. La única diferencia consiste en establecer `TLSAccept=psk` en el archivo de configuración del proxy de Zabbix y establecer *Conexiones al proxy* en el frontend de Zabbix como PSK.

Solución de problemas de PSK PSK contiene un número impar de dígitos hexadecimales

El proxy o el agent no se inicia, mensaje en el log del proxy o del agent:

```
PSK no válida en el archivo "/home/zabbix/zabbix_proxy.psk"
```

Se pasa a GnuTLS una cadena de identidad PSK de más de 128 bytes

En el registro del lado cliente de TLS:

```
gnutls_handshake() failed: -110 The TLS connection was non-properly terminated.
```

En el registro del lado servidor de TLS.

```
gnutls_handshake() failed: -90 The SRP username supplied is illegal.
```

Valor PSK demasiado largo usado con OpenSSL 1.1.1

En el registro del lado de la conexión:

```
...OpenSSL library (version OpenSSL 1.1.1  11 Sep 2018) initialized
...
...In zbx_tls_connect(): psk_identity:"PSK 1"
...zbx_psk_client_cb() requested PSK identity "PSK 1"
...End of zbx_tls_connect():FAIL error:'SSL_connect() set result code to SSL_ERROR_SSL: file ssl\statem\ex
```

En el registro del lado de aceptación:

```
...Message from 123.123.123.123 is missing header. Message ignored.
```

Este problema suele aparecer al actualizar OpenSSL de 1.0.x o 1.1.0 a 1.1.1 y si el valor PSK es más largo que 512 bits (PSK de 64 bytes, introducido como 128 dígitos hexadecimales).

Véase también: Límites de tamaño de valor

Conexión segura a la base de datos

Resumen

Esta sección proporciona pasos de configuración de Zabbix y ejemplos de configuración para conexiones TLS seguras entre:

Database	Zabbix components
MySQL	Zabbix frontend, Zabbix server, Zabbix proxy
PostgreSQL	Zabbix frontend, Zabbix server, Zabbix proxy

Para configurar el cifrado de la conexión dentro del DBMS, consulte la documentación oficial del proveedor para obtener más detalles:

- MySQL: servidores de base de datos de replicación de origen y réplica.
- MySQL: servidores de base de datos de replicación en grupo, etc.
- PostgreSQL opciones de cifrado.

Todos los ejemplos se basan en las versiones GA de MySQL CE (8.0) y PgSQL (13) disponibles a través de los repositorios oficiales usando CentOS 8.

Requisitos

Se requiere lo siguiente para configurar el cifrado:

- Sistema operativo compatible con el desarrollador con OpenSSL >=1.1.X o una alternativa.

> **Note:**
> Se recomienda evitar sistemas operativos en estado de fin de vida, especialmente en el caso de nuevas instalaciones

- Motor de base de datos (RDBMS) instalado y mantenido desde el repositorio oficial proporcionado por el desarrollador. Los sistemas operativos suelen incluir versiones desactualizadas del software de base de datos para las que no está implementado el soporte de cifrado; por ejemplo, sistemas basados en RHEL 7 y PostgreSQL 9.2, MariaDB 5.5 sin soporte de cifrado.

Terminología

Configurar esta opción obliga a usar una conexión TLS a la base de datos desde el server/proxy de Zabbix y el frontend hacia la base de datos:

- `required` - conectar usando TLS como modo de transporte sin comprobaciones de identidad
- `verify_ca` - conectar usando TLS y verificar el certificado
- `verify_full` - conectar usando TLS, verificar el certificado y verificar que la identidad de la base de datos (CN) especificada por DBHost coincida con su certificado

Configuración de Zabbix

Frontend a la base de datos

Se puede configurar una conexión segura a la base de datos durante la instalación del frontend:

- Marque la casilla *Cifrado TLS de la base de datos* en el paso Configurar la conexión a la base de datos para habilitar el cifrado del transporte.
- Marque la casilla *Verificar el certificado de la base de datos* que aparece al seleccionar el campo *Cifrado TLS* para habilitar el cifrado con certificados.

Note:
Para MySQL, la casilla *Cifrado TLS de la base de datos* está deshabilitada si *Host de la base de datos* está establecido en localhost, porque la conexión que utiliza un archivo de socket (en Unix) o memoria compartida (en Windows) no se puede cifrar.
 Para PostgreSQL, la casilla *Cifrado TLS* está deshabilitada si el valor del campo *Host de la base de datos* comienza con una barra diagonal o el campo está vacío.

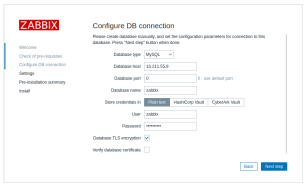
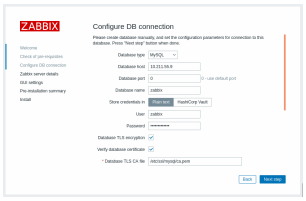
Los siguientes parámetros están disponibles en el modo de cifrado TLS con certificados (si ambas casillas están seleccionadas):

Parámetro	Descripción
Archivo de CA TLS de la base de datos	Especifique la ruta completa a un archivo válido de la autoridad certificadora (CA) TLS.
Archivo de clave TLS de la base de datos	Especifique la ruta completa a un archivo de clave TLS válido.
Archivo de certificado TLS de la base de datos	Especifique la ruta completa a un archivo de certificado TLS válido.
Verificación del host de la base de datos	Marque esta casilla para activar la verificación del host. Deshabilitada para MySQL porque la biblioteca PHP de MySQL no permite omitir el paso de validación del certificado del par.
Lista de cifrados TLS de la base de datos	Especifique una lista personalizada de cifrados válidos. El formato de la lista de cifrados debe cumplir con el estándar de OpenSSL. Disponible únicamente para MySQL.

Attention:
Los parámetros TLS deben apuntar a archivos válidos. Si apuntan a archivos inexistentes o no válidos, se producirá un error de autorización.
 Si los archivos de certificado tienen permisos de escritura, el frontend genera una advertencia en el informe Información del sistema: "Los archivos de certificado TLS deben ser de solo lectura." (se muestra únicamente si el usuario de PHP es el propietario del certificado).
No se admiten certificados protegidos mediante contraseñas.

Casos de uso

El frontend de Zabbix utiliza una interfaz GUI para definir las opciones posibles: `required`, `verify_ca`, `verify_full`. Especifique las opciones necesarias en el paso del asistente de instalación *Configurar conexiones de DB*. Estas opciones se asignan al archivo de configuración del frontend (`zabbix.conf.php`) de la siguiente manera:

Configuración de la GUI	Archivo de configuración	Descripción	Resultado
	... // Used for TLS connection. $DB['ENCRYPTION'] = true; $DB['KEY_FILE'] = ''; $DB['CERT_FILE'] = ''; $DB['CA_FILE'] = ''; $DB['VERIFY_HOST'] = false; $DB['CIPHER_LIST'] = ''; ...	1. Marque *Cifrado TLS de la base de datos* 2. Deje sin marcar *Verificar el certificado de la base de datos*	Habilita el modo `required`.
	... $DB['ENCRYPTION'] = true; $DB['KEY_FILE'] = ''; $DB['CERT_FILE'] = ''; $DB['CA_FILE'] = '/etc/ssl/mysql/ca.pem'; $DB['VERIFY_HOST'] = false; $DB['CIPHER_LIST'] = ''; ...	1. Marque *Cifrado TLS de la base de datos* y *Verificar el certificado de la base de datos* 2. Especifique la ruta al *archivo de CA TLS de la base de datos*	Habilita el modo `verify_ca`.

Configuración de la GUI	Archivo de configuración	Descripción	Resultado

	...
	// Used for TLS connection with strictly defined Cipher list.
	`$DB['ENCRYPTION'] = true;`
	`$DB['KEY_FILE'] = '<key_file_path>';`
	`$DB['CERT_FILE'] = '<key_file_path>';`
	`$DB['CA_FILE'] = '<key_file_path>';`
	`$DB['VERIFY_HOST'] = true;`
	`$DB['CIPHER_LIST'] = '<cipher_list>';`
	...
	
	O bien:
	
	...
	// Used for TLS connection without Cipher list defined - selected by MySQL server
	`$DB['ENCRYPTION'] = true;`
	`$DB['KEY_FILE'] = '<key_file_path>';`
	`$DB['CERT_FILE'] = '<key_file_path>';`
	`$DB['CA_FILE'] = '<key_file_path>';`
	`$DB['VERIFY_HOST'] = true;`
	`$DB['CIPHER_LIST'] = '';`
	...
		1. Marque *Cifrado TLS de la base de datos* y *Verificar el certificado de la base de datos*
		2. Especifique la ruta al *archivo de clave TLS de la base de datos*
		3. Especifique la ruta al *archivo de CA TLS de la base de datos*
		4. Especifique la ruta al *archivo de certificado TLS de la base de datos*
		5. Especifique la *lista de cifrados TLS de la base de datos* (opcional)
			Habilita el modo `verify_full` para MySQL.

Configuración de la GUI	Archivo de configuración	Descripción	Resultado

	...
	`$DB['ENCRYPTION'] = true;`
	`$DB['KEY_FILE'] = '<key_file_path>';`
	`$DB['CERT_FILE'] = '<key_file_path>';`
	`$DB['CA_FILE'] = '<key_file_path>';`
	`$DB['VERIFY_HOST'] = true;`
	`$DB['CIPHER_LIST'] = ' ';`
	...
		1. Marque *Cifrado TLS de la base de datos* y *Verificar el certificado de la base de datos*
		2. Especifique la ruta al *archivo de clave TLS de la base de datos*
		3. Especifique la ruta al *archivo de CA TLS de la base de datos*
		4. Especifique la ruta al *archivo de certificado TLS de la base de datos*
		5. Marque *Verificación del host de la base de datos*
			Habilita el modo `verify_full` para PostgreSQL.

Véase también: Ejemplos de configuración del cifrado para MySQL, Ejemplos de configuración del cifrado para PostgreSQL.

Configuración del server/proxy

Las conexiones seguras a la base de datos se pueden configurar con los parámetros correspondientes en el archivo de configuración del server y/o del proxy de Zabbix.

Configuración	Resultado
Ninguna	Conexión a la base de datos sin cifrado.
Establecer `DBTLSConnect=required`	El server/proxy establece una conexión TLS con la base de datos. No se permite una conexión sin cifrado.
1. Establecer `DBTLSConnect=verify_ca` 2. Establecer `DBTLSCAFile`: especificar el archivo de la autoridad certificadora TLS	El server/proxy establece una conexión TLS con la base de datos después de verificar el certificado de la base de datos.
1. Establecer `DBTLSConnect=verify_full` 2. Establecer `DBTLSCAFile`: especificar el archivo de la autoridad certificadora TLS	El server/proxy establece una conexión TLS con la base de datos después de verificar el certificado de la base de datos y la identidad del host de la base de datos.
1. Establecer `DBTLSCAFile`: especificar el archivo de la autoridad certificadora TLS 2. Establecer `DBTLSCertFile`: especificar el archivo del certificado de clave pública del cliente 3. Establecer `DBTLSKeyFile`: especificar el archivo de clave privada del cliente	El server/proxy proporciona un certificado de cliente al conectarse a la base de datos.
Establecer `DBTLSCipher`: la lista de cifrados de encriptación que el cliente permite para las conexiones que utilizan protocolos TLS hasta TLS 1.2 o `DBTLSCipher13`: la lista de cifrados de encriptación que el cliente permite para las conexiones que utilizan el protocolo TLS 1.3	(MySQL) La conexión TLS se establece utilizando un cifrado de la lista proporcionada. (PostgreSQL) Establecer esta opción se considerará un error.

Configuración de cifrado de MySQL

Resumen

Esta sección proporciona varios ejemplos de configuración de cifrado para CentOS 8.2 y MySQL 8.4.0 y puede utilizarse como una guía de inicio rápido para cifrar la conexión a la base de datos.

Attention:
Si el host de MySQL está configurado como localhost, las opciones de cifrado no estarán disponibles. En este caso, una conexión entre el frontend de Zabbix y la base de datos utiliza un archivo de socket (en Unix) o memoria compartida (en Windows) y no puede cifrarse.

Note:
La lista de combinaciones de cifrado no se limita a las que se enumeran en esta página. Hay muchas más combinaciones disponibles.

Requisitos previos

Instale la base de datos MySQL desde el repositorio oficial.

Consulte la documentación de MySQL para obtener detalles sobre cómo usar el repositorio de MySQL.

El server de MySQL está listo para aceptar conexiones seguras mediante un certificado autofirmado.

Para ver qué usuarios están usando una conexión cifrada, ejecute la siguiente consulta (Performance Schema debe estar activado):

```
SELECT sbt.variable_value AS tls_version, t2.variable_value AS cipher, processlist_user AS user, processli
FROM performance_schema.status_by_thread  AS sbt
JOIN performance_schema.threads AS t ON t.thread_id = sbt.thread_id
JOIN performance_schema.status_by_thread AS t2 ON t2.thread_id = t.thread_id
WHERE sbt.variable_name = 'Ssl_version' and t2.variable_name = 'Ssl_cipher'
ORDER BY tls_version;
```

Cifrado solo de transporte

Configuración de MySQL

Las versiones modernas de la base de datos están listas para usar con el modo de cifrado `required`. Después de la configuración inicial y el inicio, se creará un certificado del lado del server.

Cree usuarios y roles para los componentes principales (reemplace `<strong_password>` por una contraseña segura y única):

```
mysql> CREATE USER
 'zbx_srv'@'%' IDENTIFIED WITH caching_sha2_password BY '<strong_password>',
 'zbx_web'@'%' IDENTIFIED WITH caching_sha2_password BY '<strong_password>'
 REQUIRE SSL
 PASSWORD HISTORY 5;

mysql> CREATE ROLE 'zbx_srv_role', 'zbx_web_role';

mysql> GRANT SELECT, UPDATE, DELETE, INSERT, CREATE, DROP, ALTER, INDEX, REFERENCES ON zabbix.* TO 'zbx_sr
mysql> GRANT SELECT, UPDATE, DELETE, INSERT ON zabbix.* TO 'zbx_web_role';

mysql> GRANT 'zbx_srv_role' TO 'zbx_srv'@'%';
mysql> GRANT 'zbx_web_role' TO 'zbx_web'@'%';

mysql> SET DEFAULT ROLE 'zbx_srv_role' TO 'zbx_srv'@'%';
mysql> SET DEFAULT ROLE 'zbx_web_role' TO 'zbx_web'@'%';
```

Tenga en cuenta que el protocolo X.509 no se utiliza para comprobar la identidad, sino que el usuario está configurado para utilizar únicamente conexiones cifradas. Consulte la documentación de MySQL para obtener más información sobre la configuración de usuarios.

Ejecute lo siguiente para comprobar la conexión (no se puede utilizar una conexión mediante socket para probar conexiones seguras):

```
mysql -u zbx_srv -p -h 10.211.55.9 --ssl-mode=REQUIRED
```

Compruebe el estado actual y los conjuntos de cifrado disponibles:

```
mysql> status
--------------
mysql Ver 8.4.0 for Linux on x86_64 (MySQL Community Server - GPL)

Connection id: 62
```

```
Current database:
Current user: zbx_srv@bfdb.local
SSL: Cipher in use is TLS_AES_256_GCM_SHA384

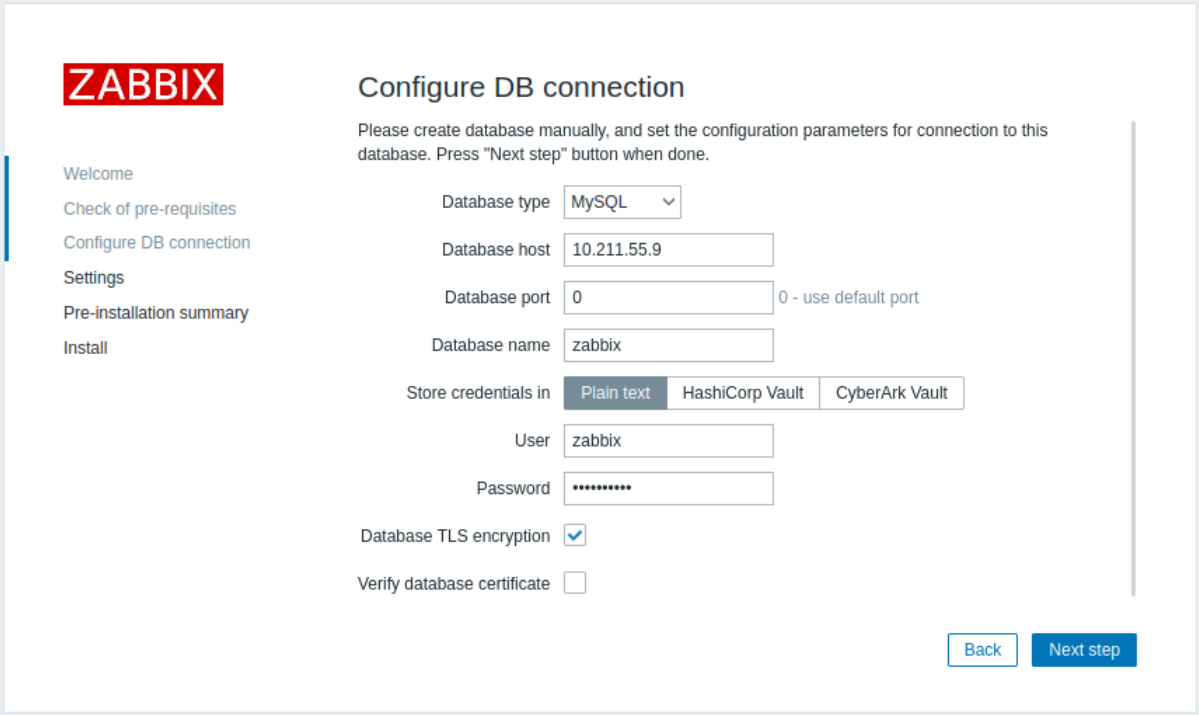

mysql> SHOW SESSION STATUS LIKE 'Ssl_cipher_list'\G;
*************************** 1. row ***************************
Variable_name: Ssl_cipher_list
Value: TLS_AES_256_GCM_SHA384:TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256:TLS_AES_128_CCM_SHA256:E
1 row in set (0.00 sec)

ERROR:
No query specified
```

Frontend

Para habilitar el cifrado solo de transporte para las conexiones entre el frontend de Zabbix y la base de datos:

- Marque *Database TLS encryption*
- Deje sin marcar *Verify database certificate*



server

Para habilitar el cifrado exclusivo del transporte para las conexiones entre el server y la base de datos, edite el archivo de configuración del server:

```
...
DBHost=10.211.55.9
DBName=zabbix
DBUser=zbx_srv
DBPassword=<strong_password>
DBTLSConnect=required
...
```

Cifrado con verificación de la autoridad certificadora

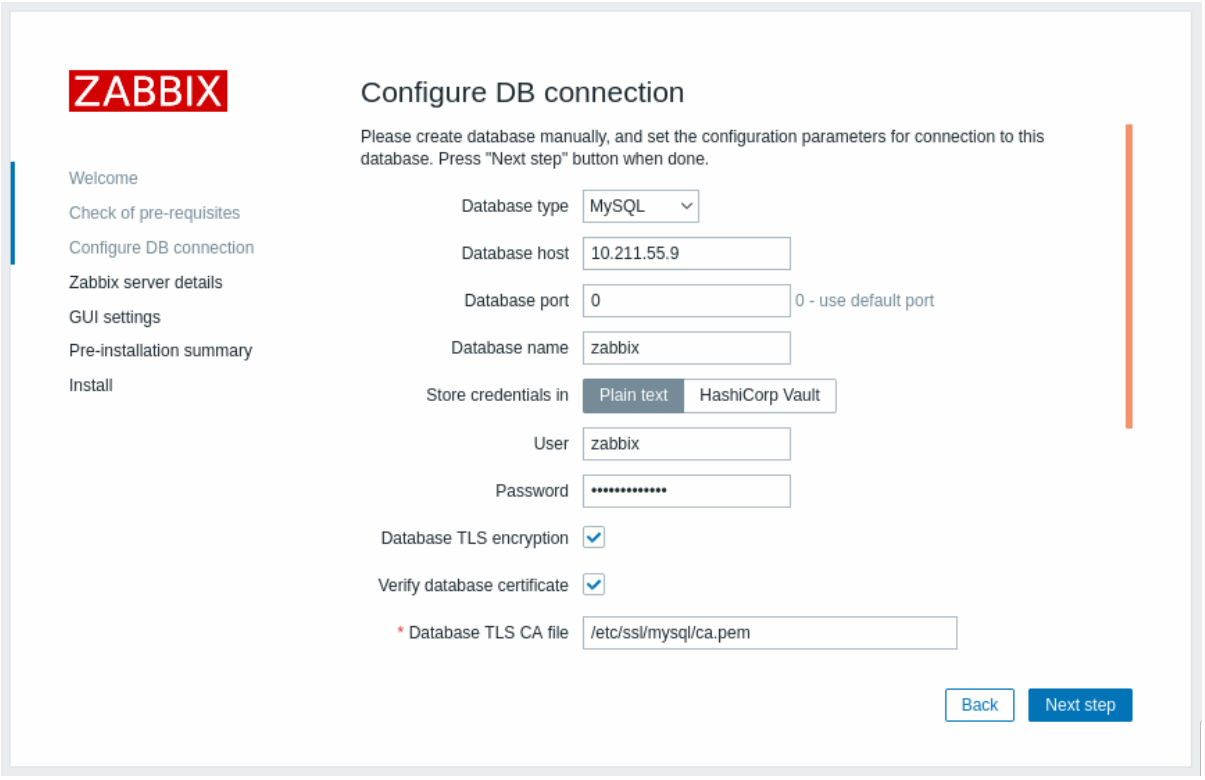
Copie el CA de MySQL requerido al server frontend de Zabbix y asigne los permisos adecuados para permitir que el servidor web lea este archivo.

> **Note:**
> Este modo no funciona en RHEL 7 debido a bibliotecas de MySQL más antiguas.

frontend

Para habilitar el cifrado con verificación de certificados en las conexiones entre el frontend de Zabbix y la base de datos:

- Marque *Cifrado TLS de la base de datos* y *Verificar el certificado de la base de datos*
- Especifique la ruta al archivo de CA TLS de la base de datos



Como alternativa, esto se puede configurar en el archivo de configuración del frontend (`zabbix.conf.php`):

```
...
$DB['ENCRYPTION'] = true;
$DB['KEY_FILE'] = '';
$DB['CERT_FILE'] = '';
$DB['CA_FILE'] = '/etc/ssl/mysql/ca.pem';
$DB['VERIFY_HOST'] = false;
$DB['CIPHER_LIST'] = '';
...
```

Solucione los problemas utilizando la herramienta de línea de comandos para comprobar si la conexión es posible con el usuario requerido:

```
    mysql -u zbx_web -p -h 10.211.55.9 --ssl-mode=REQUIRED --ssl-ca=/var/lib/mysql/ca.pem
```

Server

Para habilitar el cifrado con verificación de certificados para las conexiones entre el server de Zabbix y la base de datos, edite el archivo de configuración del server:

```
...
DBHost=10.211.55.9
DBName=zabbix
DBUser=zbx_srv
DBPassword=<strong_password>
DBTLSConnect=verify_ca
DBTLSCAFile=/etc/ssl/mysql/ca.pem
...
```

Cifrado con verificación completa

Configuración de MySQL

Establezca la opción de configuración del servidor MySQL CE (*/etc/my.cnf.d/server-tls.cnf*) en:

```
[mysqld]
...
#### en este ejemplo, las claves se encuentran en el directorio datadir de MySQL CE
```

```
ssl_ca=ca.pem
ssl_cert=server-cert.pem
ssl_key=server-key.pem

require_secure_transport=ON
tls_version=TLSv1.3
...
```

Las claves para el servidor y el cliente de MySQL CE (frontend de Zabbix) deben crearse manualmente de acuerdo con la documentación de MySQL CE: Creating SSL and RSA certificates and keys using MySQL oCreating SSL certificates and keys using openssl

Attention:
El certificado del servidor MySQL debe contener el campo Common Name establecido con el nombre FQDN, ya que el frontend de Zabbix usará el nombre DNS para comunicarse con la base de datos o la dirección IP del host de la base de datos.

Cree el usuario de MySQL:

```
mysql> CREATE USER
  'zbx_srv'@'%' IDENTIFIED WITH caching_sha2_password BY '<strong_password>',
  'zbx_web'@'%' IDENTIFIED WITH caching_sha2_password BY '<strong_password>'
  REQUIRE X509
  PASSWORD HISTORY 5;
```

Compruebe si es posible iniciar sesión con ese usuario:

```
mysql -u zbx_web -p -h 10.211.55.9 --ssl-mode=VERIFY_IDENTITY --ssl-ca=/var/lib/mysql/ca.pem --ssl-cert=/v
```

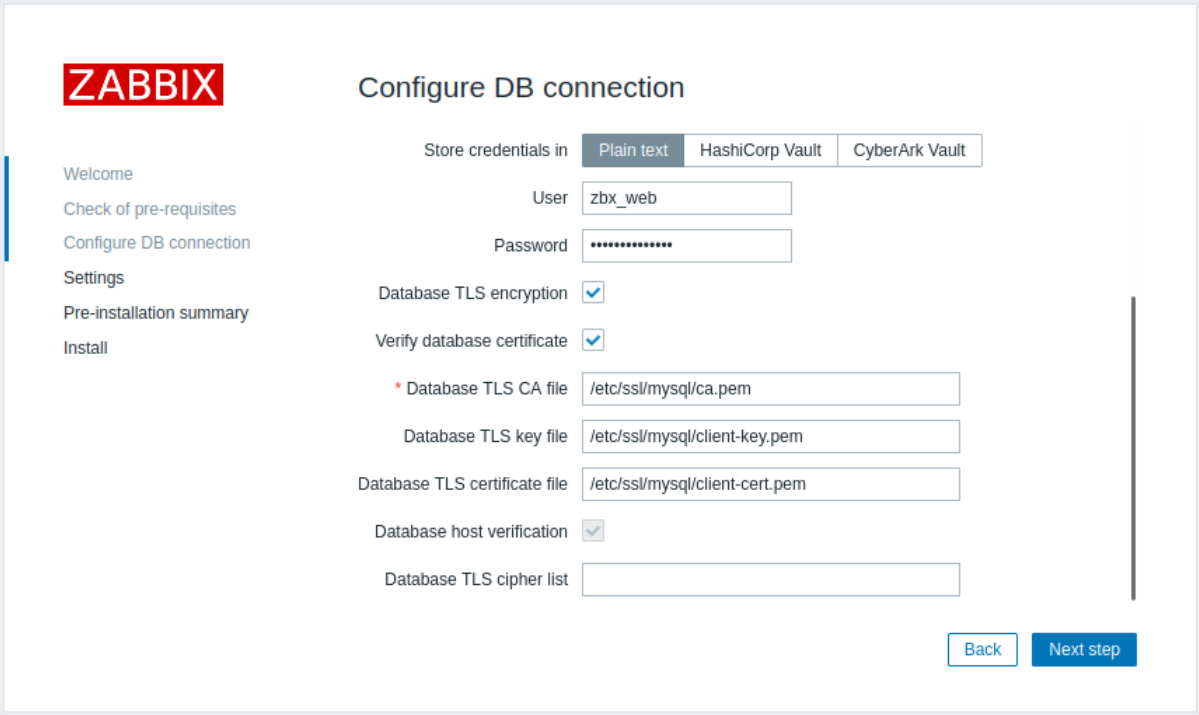
Frontend

Para habilitar el cifrado con verificación completa de las conexiones entre el frontend de Zabbix y la base de datos:

- Marque *Cifrado TLS de la base de datos* y *Verificar certificado de la base de datos*
- Especifique la ruta al *archivo de clave TLS de la base de datos*
- Especifique la ruta al *archivo de CA TLS de la base de datos*
- Especifique la ruta al *archivo de certificado TLS de la base de datos*

Tenga en cuenta que *Verificación del host de la base de datos* está marcado y aparece atenuado: este paso no se puede omitir para MySQL.

Warning:
Si el campo *Lista de cifrados TLS de la base de datos* se deja vacío, se habilitarán los cifrados comunes permitidos tanto por el frontend (cliente) como por el server. Como alternativa, los cifrados se pueden establecer explícitamente, de acuerdo con los requisitos de configuración de cifrados.



Como alternativa, esto se puede establecer en el archivo de configuración del frontend (`zabbix.conf.php`):

```
...
// Se utiliza para la conexión TLS con una lista de cifrados definida estrictamente.
$DB['ENCRYPTION'] = true;
$DB['KEY_FILE'] = '/etc/ssl/mysql/client-key.pem';
$DB['CERT_FILE'] = '/etc/ssl/mysql/client-cert.pem';
$DB['CA_FILE'] = '/etc/ssl/mysql/ca.pem';
$DB['VERIFY_HOST'] = true;
$DB['CIPHER_LIST'] = 'TLS_AES_256_GCM_SHA384:TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256:TLS_AES_1
...
// o
...
// Se utiliza para la conexión TLS sin una lista de cifrados definida; la selecciona el server de MySQL.
$DB['ENCRYPTION'] = true;
$DB['KEY_FILE'] = '/etc/ssl/mysql/client-key.pem';
$DB['CERT_FILE'] = '/etc/ssl/mysql/client-cert.pem';
$DB['CA_FILE'] = '/etc/ssl/mysql/ca.pem';
$DB['VERIFY_HOST'] = true;
$DB['CIPHER_LIST'] = '';
...
```

Server

Para habilitar el cifrado con verificación completa para las conexiones entre Zabbix server y la base de datos, edite el archivo de configuración del server:

```
...
DBHost=10.211.55.9
DBName=zabbix
DBUser=zbx_srv
DBPassword=<strong_password>
DBTLSConnect=verify_full
DBTLSCAFile=/etc/ssl/mysql/ca.pem
DBTLSCertFile=/etc/ssl/mysql/client-cert.pem
DBTLSKeyFile=/etc/ssl/mysql/client-key.pem
...
```

Configuración de cifrado de PostgreSQL

Descripción general

Esta sección proporciona varios ejemplos de configuración de cifrado para CentOS 8.2 y PostgreSQL 13.

Note:
La conexión entre Zabbix frontend y PostgreSQL no puede cifrarse (los parámetros en la interfaz gráfica están deshabilitados) si el valor del campo *Database host* comienza con una barra diagonal o si el campo está vacío.

Requisitos previos

Instale la base de datos PostgreSQL usando el repositorio oficial.

PostgreSQL no está configurado para aceptar conexiones TLS de forma predeterminada. Siga las instrucciones de la documentación de PostgreSQL para la preparación de certificados con postgresql.conf y también para el control de acceso de usuarios a través de `ph_hba.conf`.

De forma predeterminada, el socket de PostgreSQL está vinculado a localhost; para permitir conexiones remotas de red, debe escuchar en la interfaz de red real.

La configuración de PostgreSQL para todos los modes puede verse así:

`/var/lib/pgsql/13/data/postgresql.conf`:

```
...
ssl = on
ssl_ca_file = 'root.crt'
ssl_cert_file = 'server.crt'
ssl_key_file = 'server.key'
ssl_ciphers = 'HIGH:MEDIUM:+3DES:!aNULL'
ssl_prefer_server_ciphers = on
ssl_min_protocol_version = 'TLSv1.3'
...
```

Para el control de acceso, ajuste `/var/lib/pgsql/13/data/pg_hba.conf`:

```
...
### require
hostssl all all 0.0.0.0/0 md5

### verify CA
hostssl all all 0.0.0.0/0 md5 clientcert=verify-ca

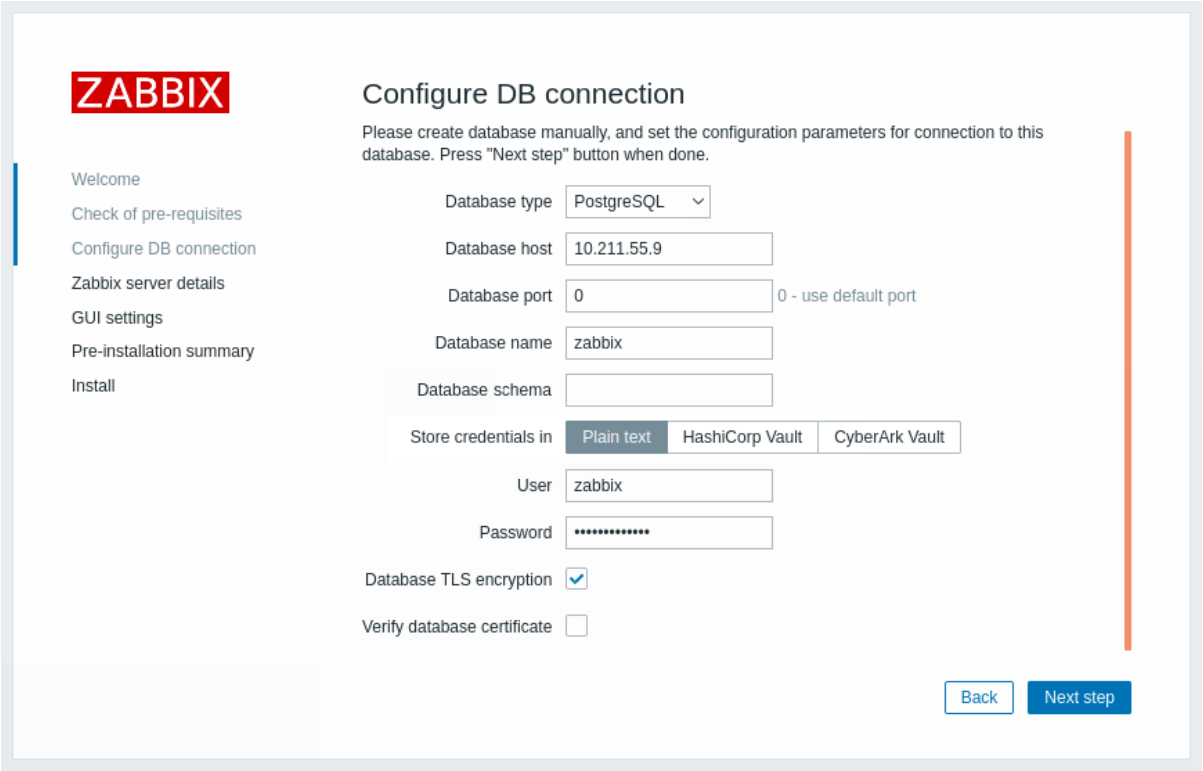
### verify full
hostssl all all 0.0.0.0/0 md5 clientcert=verify-full
...
```

Cifrado solo de transporte

Frontend

Para habilitar el cifrado solo de transporte para las conexiones entre el frontend de Zabbix y la base de datos:

- Marque *Database TLS encryption*
- Deje sin marcar *Verify database certificate*



server

Para habilitar el cifrado exclusivo del transporte para las conexiones entre el server y la base de datos, edite el archivo de configuración del server:

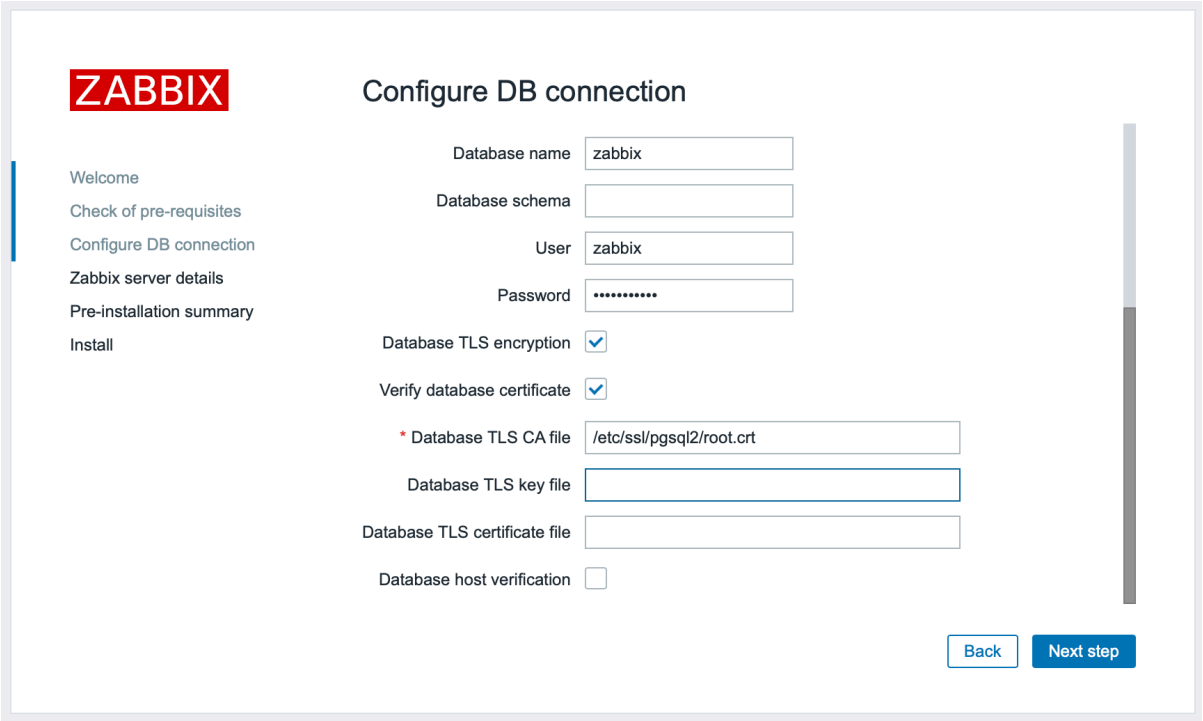
```
...
DBHost=10.211.55.9
DBName=zabbix
DBUser=zbx_srv
DBPassword=<strong_password>
DBTLSConnect=required
...
```

Cifrado con verificación de la autoridad de certificación

frontend

Para habilitar el cifrado con verificación de la autoridad certificadora para las conexiones entre el frontend de Zabbix y la base de datos:

- Marque *Cifrado TLS de la base de datos* y *Verificar el certificado de la base de datos*
- Especifique la ruta al *archivo de CA TLS de la base de datos*



Como alternativa, esto se puede configurar en el archivo de configuración del frontend (`zabbix.conf.php`):

```
...
$DB['ENCRYPTION'] = true;
$DB['KEY_FILE'] = '';
$DB['CERT_FILE'] = '';
$DB['CA_FILE'] = '/etc/ssl/pgsql/root.crt';
$DB['VERIFY_HOST'] = false;
$DB['CIPHER_LIST'] = '';
...
```

Server

Para habilitar el cifrado con verificación de certificados para las conexiones entre Zabbix server y la base de datos, edite el archivo de configuración del server (reemplace `<strong_password>` por una contraseña segura y única):

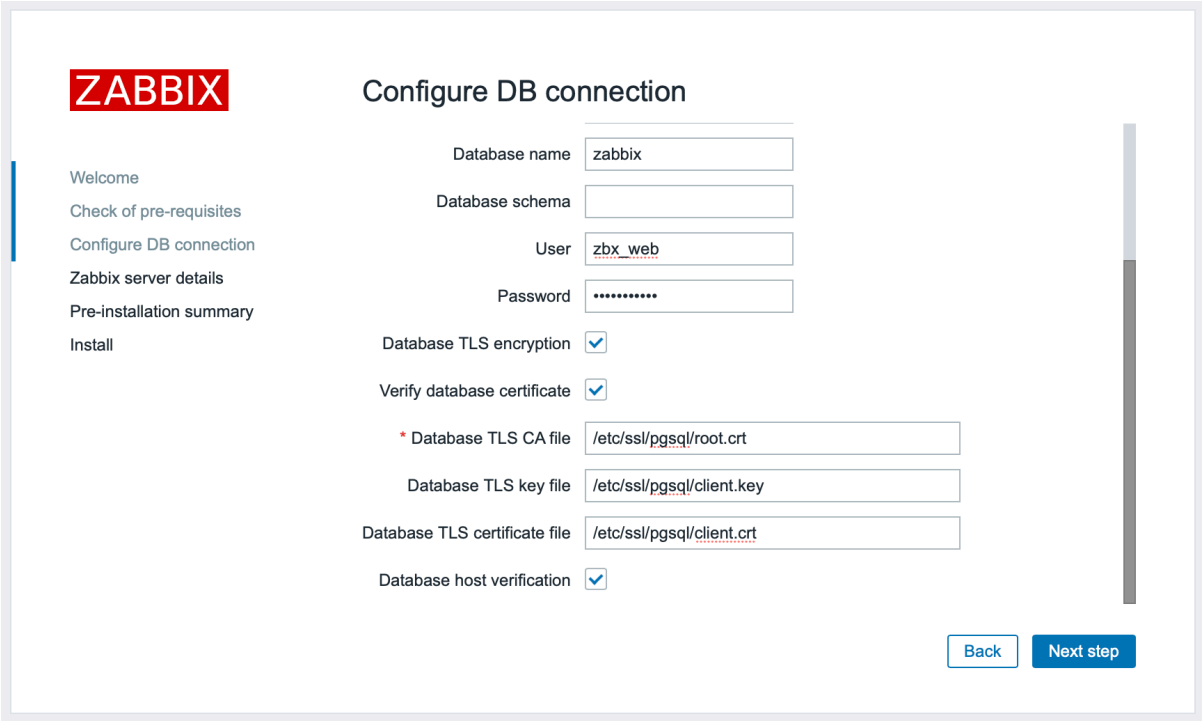
```
...
DBHost=10.211.55.9
DBName=zabbix
DBUser=zbx_srv
DBPassword=<strong_password>
DBTLSConnect=verify_ca
DBTLSCAFile=/etc/ssl/pgsql/root.crt
...
```

Cifrado con verificación completa

Frontend

Para habilitar el cifrado con verificación del certificado y de la identidad del host de la base de datos para las conexiones entre el frontend de Zabbix y la base de datos:

- Marque *Cifrado TLS de la base de datos* y *Verificar el certificado de la base de datos*
- Especifique la ruta al *archivo de clave TLS de la base de datos*
- Especifique la ruta al *archivo de CA TLS de la base de datos*
- Especifique la ruta al *archivo de certificado TLS de la base de datos*
- Marque *Verificación del host de la base de datos*



Como alternativa, esto se puede configurar en el archivo de configuración del frontend (`zabbix.conf.php`):

```
$DB['ENCRYPTION'] = true;
$DB['KEY_FILE'] = '';
$DB['CERT_FILE'] = '';
$DB['CA_FILE'] = '/etc/ssl/pgsql/root.crt';
$DB['VERIFY_HOST'] = true;
$DB['CIPHER_LIST'] = '';
...
```

Server

Para habilitar el cifrado con verificación de identidad del certificado y del host de la base de datos para las conexiones entre el server de Zabbix y la base de datos, edite el archivo de configuración del server:

```
...
DBHost=10.211.55.9
DBName=zabbix
DBUser=zbx_srv
DBPassword=<strong_password>
DBTLSConnect=verify_full
DBTLSCAFile=/etc/ssl/pgsql/root.crt
DBTLSCertFile=/etc/ssl/pgsql/client.crt
DBTLSKeyFile=/etc/ssl/pgsql/client.key
...
```

Conexión segura entre el frontend y el server

Descripción general

Esta sección proporciona pasos de configuración de Zabbix y ejemplos de configuración para conexiones TLS seguras entre el frontend de Zabbix y el server de Zabbix.

Configuración

De forma predeterminada, la comunicación entre el frontend de Zabbix y el server de Zabbix no está cifrada. Para mejorar la seguridad, habilite TLS en ambos lados. A continuación se muestra un ejemplo de la forma más sencilla de hacerlo.

1. Genere certificados y claves.

Cree un directorio de trabajo:

```
sudo mkdir -p /etc/zabbix/ssl && cd /etc/zabbix/ssl
```

Cree un certificado de CA (ajuste el valor `MyZabbixCA` para que coincida con el nombre común real):

```
sudo openssl genrsa -out ca.key 4096
sudo openssl req -new -x509 -days 3650 -key ca.key -out ca.crt -subj "/CN=MyZabbixCA/"
```

Genere una clave privada y un certificado para el server de Zabbix (ajuste el valor `zabbix-server.example.com` para que coincida con el nombre común real):

```
sudo openssl genrsa -out server.key 2048
sudo openssl req -new -key server.key -out server.csr -subj "/CN=zabbix-server.example.com/"
sudo openssl x509 -req -days 365 -in server.csr -CA ca.crt -CAkey ca.key -CAcreateserial -sha256 -out serv
```

Genere una clave privada y un certificado para el frontend de Zabbix (ajuste el valor `zabbix-frontend-node` para que coincida con el nombre común real):

```
sudo openssl genrsa -out frontend.key 2048
sudo openssl req -new -key frontend.key -out frontend.csr -subj "/CN=zabbix-frontend-node/"
sudo openssl x509 -req -days 365 -in frontend.csr -CA ca.crt -CAkey ca.key -CAcreateserial -sha256 -out fr
```

2. Establezca los permisos adecuados.

Para el server de Zabbix (ajuste el propietario/grupo según el usuario del demonio del server de Zabbix de su distribución):

```
sudo chown root:zabbix /etc/zabbix/ssl/server.{crt,key} /etc/zabbix/ssl/ca.crt
sudo chmod 640 /etc/zabbix/ssl/server.key
sudo chmod 644 /etc/zabbix/ssl/server.crt /etc/zabbix/ssl/ca.crt
```

Para el frontend (ajuste el propietario/grupo según el usuario del webserver de su distribución):

```
sudo chown root:www-data /etc/zabbix/ssl/frontend.{crt,key}
sudo chmod 640 /etc/zabbix/ssl/frontend.key
sudo chmod 644 /etc/zabbix/ssl/frontend.crt
```

3. Configure el server de Zabbix.

En el archivo de configuración del server (`zabbix_server.conf`), añada:

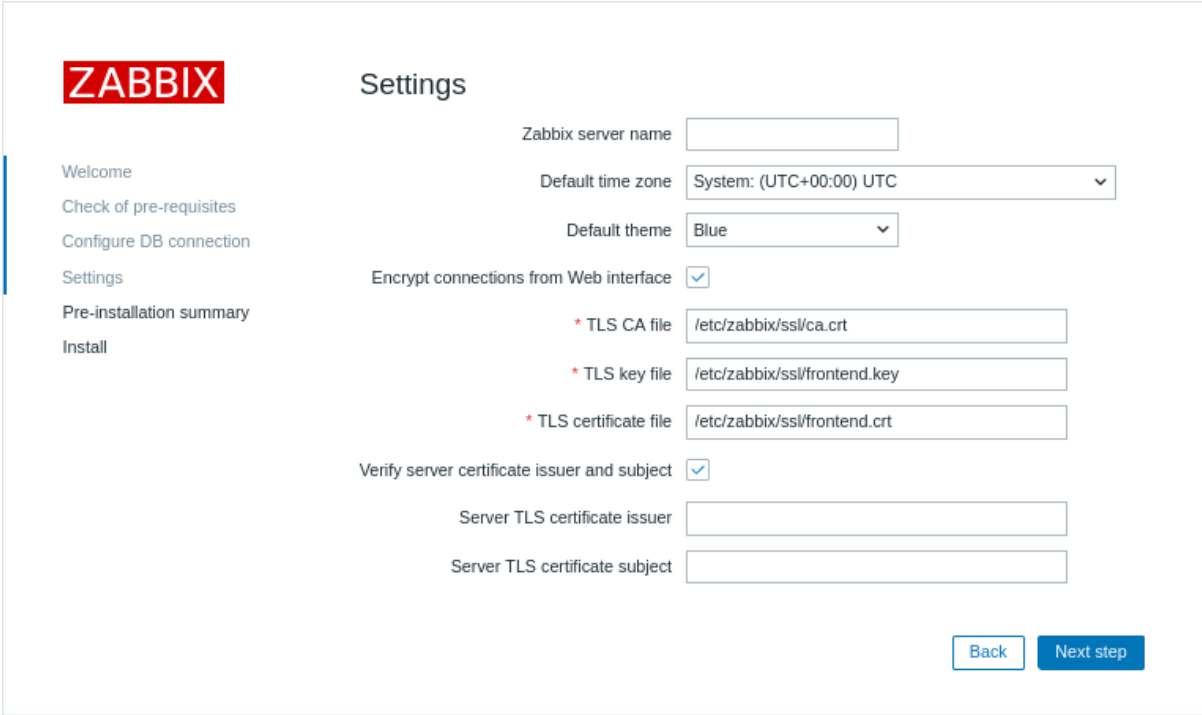
```
TLSFrontendAccept=cert
TLSCertFile=/etc/zabbix/ssl/server.crt
TLSKeyFile=/etc/zabbix/ssl/server.key
TLSCAFile=/etc/zabbix/ssl/ca.crt
### Opcionalmente:
### TLSFrontendCertIssuer=CN=MyZabbixCA
### TLSFrontendCertSubject=CN=zabbix-frontend-node
```

A continuación, reinicie el server:

```
sudo systemctl restart zabbix-server
```

4. Configure el frontend de Zabbix.

Durante la instalación de la interfaz web, habilite la opción *Cifrar las conexiones desde la interfaz web* (y la opción *Verificar el emisor y el asunto del certificado del server*, si es necesario) y complete los campos *Archivo de CA TLS*, *Archivo de clave TLS*, *Archivo de certificado TLS* (y los campos *Emisor del certificado TLS del server* y *Asunto del certificado TLS del server*, si es necesario):



Parámetro	Descripción
Archivo de CA TLS	Especifique la ruta completa al archivo de certificado de la autoridad certificadora (CA) utilizado para verificar el certificado del server.
Archivo de clave TLS	Especifique la ruta completa al archivo de clave privada del cliente correspondiente al certificado del cliente.
Archivo de certificado TLS	Especifique la ruta completa al archivo de certificado del cliente si se requiere autenticación TLS mutua.
Emisor del certificado TLS del server	Especifique un nombre distinguido (DN) del emisor que se comparará con el certificado del server.
Asunto del certificado TLS del server	Especifique un nombre distinguido (DN) del asunto que se comparará con el certificado del server.

En las instalaciones existentes, edite los siguientes parámetros en el archivo de configuración del frontend:

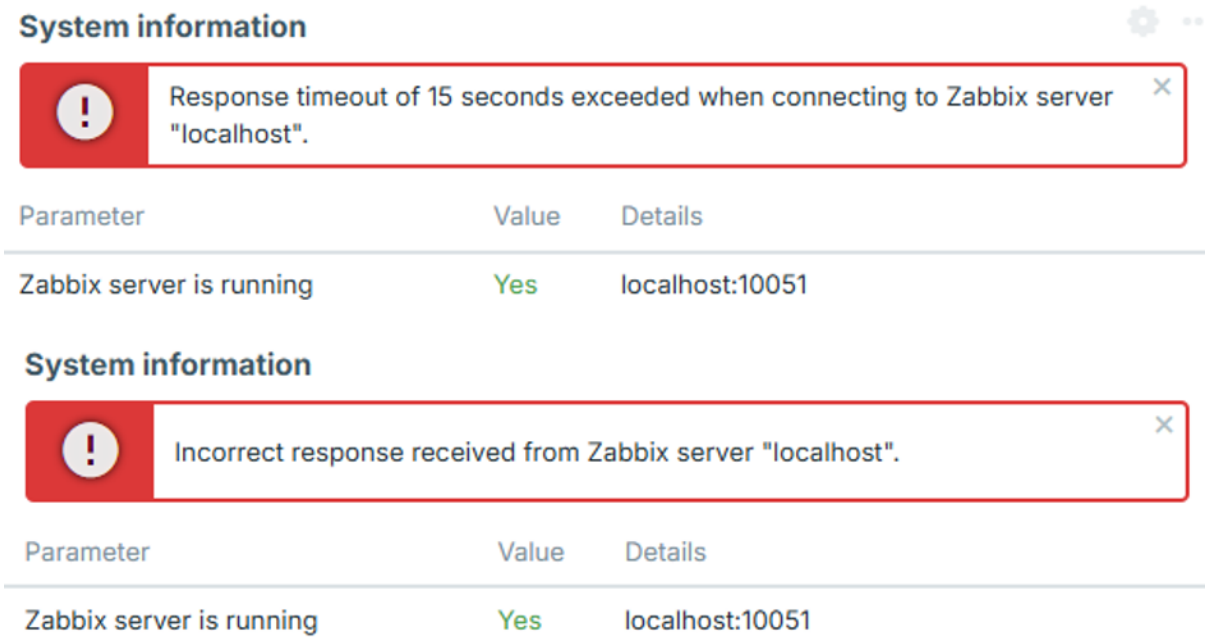
```
$ZBX_SERVER_TLS['ACTIVE'] = 'true';
$ZBX_SERVER_TLS['CA_FILE'] = '/etc/zabbix/ssl/ca.crt';
$ZBX_SERVER_TLS['KEY_FILE'] = '/etc/zabbix/ssl/frontend.key';
$ZBX_SERVER_TLS['CERT_FILE'] = '/etc/zabbix/ssl/frontend.crt';
// Opcionalmente:
// $ZBX_SERVER_TLS['CERTIFICATE_ISSUER']  = 'CN=MyZabbixCA';
// $ZBX_SERVER_TLS['CERTIFICATE_SUBJECT'] = 'CN=zabbix-server.example.com';
```

5. Verifique el cifrado confirmando que no haya mensajes de error en el frontend de Zabbix ni en el archivo de registro del server de Zabbix:

```
tail -f /var/log/zabbix/zabbix_server.log
```

Solución de problemas

Una configuración TLS incorrecta o incompleta puede provocar errores en el frontend de Zabbix o indicar que el server de Zabbix está en ejecución, pero que la conexión no se ha establecido correctamente:



Para identificar el problema exacto de configuración, establezca el nivel de registro del server de Zabbix en 4, reproduzca el error en el frontend y compruebe el registro del server de Zabbix para encontrar el mensaje de error correspondiente.

Los siguientes casos describen errores comunes y sus posibles causas.

La dirección IP del frontend no está permitida

El registro del server de Zabbix contiene un error similar a:

```
54283:20260819:105926.400 frontend connection from "127.0.0.1" is not allowed by FrontendAllowedIP
```

La conexión se interrumpe después del protocolo de enlace TCP porque la dirección IP del frontend no está autorizada.

Añada la dirección IP del frontend al parámetro `FrontendAllowedIP` en `zabbix_server.conf` y reinicie el server de Zabbix.

No se permiten conexiones sin cifrar

El registro del Zabbix server contiene un error similar al siguiente:

```
56884:20260819:111154.285 frontend connection of type "unencrypted" is not allowed by TLSFrontendAccept
```

El Zabbix server está configurado para aceptar únicamente conexiones TLS, por ejemplo, con `TLSFrontendAccept=cert`, pero el frontend intenta conectarse sin cifrado.

Habilite el cifrado del frontend y configure los archivos TLS necesarios en `zabbix.conf.php`, tal como se describe en el paso 4 de Configuración.

El frontend intenta usar TLS mientras el server espera una conexión sin cifrar

El registro del server de Zabbix contiene un error similar a:

```
61091:20260819:115916.019 End of zbx_tls_accept():FAIL error:'TLS handshake set result code to 1: file ../
```

El mensaje de error exacto depende de la versión de OpenSSL.

El server de Zabbix está configurado para aceptar únicamente conexiones sin cifrar, por ejemplo, con `TLSFrontendAccept=unencrypted`, mientras que el frontend intenta establecer una conexión TLS.

Configure el server de Zabbix para aceptar conexiones TLS estableciendo `TLSFrontendAccept=cert` y definiendo `TLSCertFile` y `TLSKeyFile` en `zabbix_server.conf`. Como alternativa, desactive el cifrado del frontend en `zabbix.conf.php`.

El certificado del frontend no es de confianza

El registro del Zabbix server contiene un error similar a:

```
62248:20260819:124203.812 End of zbx_tls_accept():FAIL error:'self-signed certificate: TLS handshake set r
```

El frontend presentó un certificado, pero el Zabbix server no confía en la Autoridad certificadora (CA) que lo firmó.

El mismo problema también puede aparecer en la información del sistema del frontend como:

```
fread(): SSL operation failed with code 1. OpenSSL Error messages:

error:0A000418:SSL routines::tlsv1 alert unknown ca [zabbix.php:17 → require_once() → ZBase->run() → ZBase
```

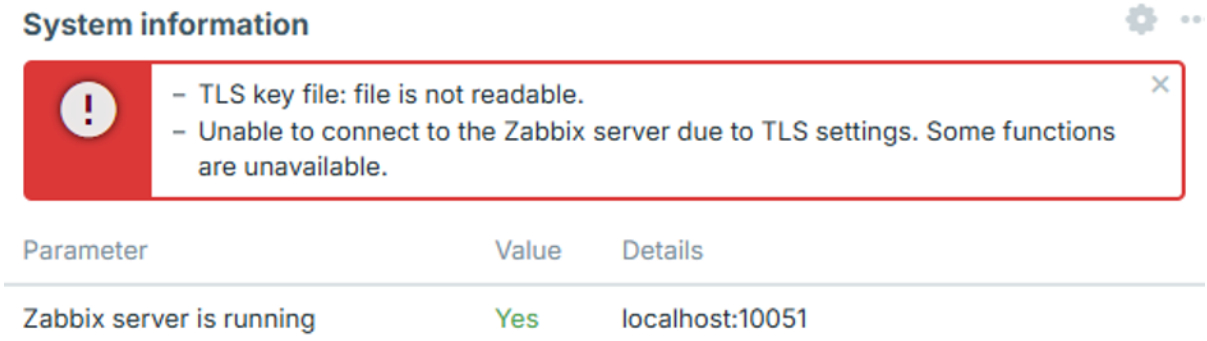
El frontend también puede mostrar «Incorrect response received from Zabbix server "localhost"».

Verifique que el certificado del frontend esté firmado por la CA especificada mediante `TLSCAFile` en `zabbix_server.conf`.

El archivo de clave TLS no se puede leer

El frontend puede mostrar «Archivo de clave TLS: no se puede leer».

También puede mostrar «No se puede conectar al server de Zabbix debido a la configuración de TLS. Algunas funciones no están disponibles».

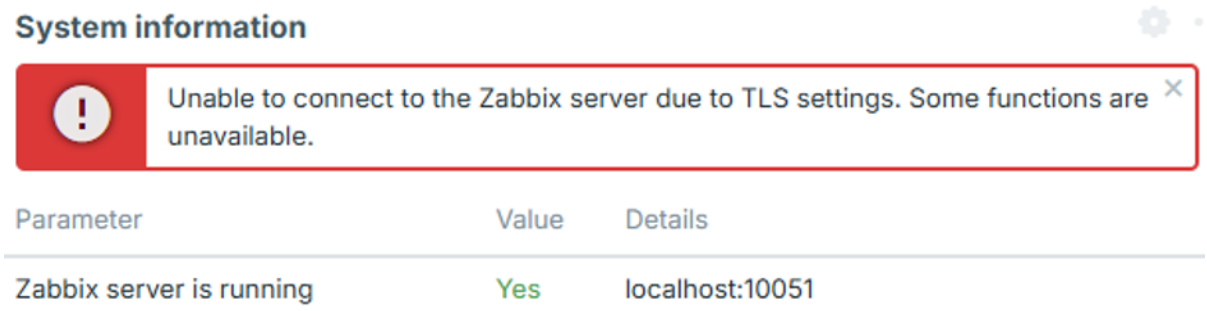


El usuario del servidor web no tiene permisos suficientes para leer uno o más archivos de certificado, clave o CA configurados en `zabbix.conf.php`. Esto puede ocurrir cuando los archivos de certificado se almacenan en un directorio restringido, como /root/ o el directorio de inicio de un usuario.

Mueva los archivos de certificado a un directorio accesible para el servidor web y establezca los permisos adecuados, tal como se describe en el paso 2 de Configuración.

Los archivos TLS no son válidos o están dañados

El frontend puede mostrar «No se puede conectar al Zabbix server debido a la configuración de TLS. Algunas funciones no están disponibles».



Esto puede ocurrir cuando los archivos de certificado o de clave configurados existen y se pueden leer, pero su contenido está vacío, dañado o no es válido por algún otro motivo. En este caso, el frontend no establece una conexión con el Zabbix server y no aparece ningún intento de conexión correspondiente en el registro del Zabbix server.

Verifique que los archivos de certificado y de clave contengan datos válidos y con el formato correcto, y que la clave privada corresponda al certificado.

Control de acceso

Descripción general

Esta sección contiene las mejores prácticas para configurar el control de acceso de forma segura.

Principio del menor privilegio

Las cuentas de usuario, en todo momento, deben ejecutarse con la menor cantidad de privilegios posible. Esto significa que las cuentas de usuario en el frontend de Zabbix, los usuarios de la base de datos o el usuario para los procesos de Zabbix server/proxy/agent solo deben tener los privilegios que sean esenciales para realizar las funciones previstas.

Attention:
Otorgar privilegios adicionales al usuario 'zabbix' le permitirá acceder a archivos de configuración y ejecutar operaciones que pueden comprometer la seguridad de la infraestructura.

Al configurar los privilegios de las cuentas de usuario, deben considerarse los tipos de usuario del frontend de Zabbix. Tenga en cuenta que, aunque el tipo de usuario *Admin* tiene menos privilegios que el tipo de usuario *Super Admin*, aún puede administrar la configuración y ejecutar scripts personalizados.

Note:
Parte de la información está disponible incluso para usuarios sin privilegios. Por ejemplo, aunque *Alerts* → *Scripts* solo está disponible para usuarios *Super Admin*, los scripts también se pueden recuperar a través de la API de Zabbix. En este caso, limitar los permisos de los scripts y excluir información sensible de los scripts (por ejemplo, credenciales de acceso) puede ayudar a evitar la exposición de información sensible disponible en los scripts globales.

Usuario seguro para Zabbix agent

De forma predeterminada, los procesos de Zabbix server, proxy y agent (o agent 2) comparten un único usuario `zabbix`. Para evitar que Zabbix agent/agent 2 (que se ejecuta en la misma máquina que server/proxy) acceda a detalles sensibles en la configuración de server/proxy (por ejemplo, credenciales de base de datos), el agent debe ejecutarse bajo un usuario diferente:

Para Zabbix agent:

1. Cree un grupo y usuario seguros (por ejemplo, `zabbix-agent`).
2. Establezca este usuario en el parámetro User del archivo de configuración del agent.
3. Reinicie el agent para descartar privilegios y pasar al nuevo usuario.

Para Zabbix agent 2, la configuración debe aplicarse a nivel de service, porque el archivo de configuración de agent 2 no admite el parámetro `User`. Como ejemplo, consulte ZBX-26442.

Revocar el acceso de escritura a la configuración de SSL (Windows)

Si ha compilado Zabbix agent en Windows, con OpenSSL ubicado en un directorio no protegido (por ejemplo, `C:\zabbix`, `c:\openssl-64bit`, `C:\OpenSSL-Win64-111-static` o `C:\dev\openssl`), asegúrese de revocar el acceso de escritura a este directorio para los usuarios que no sean administradores. De lo contrario, el agent carga la configuración de SSL desde una ruta que puede ser modificada por usuarios sin privilegios, lo que puede dar lugar a una posible vulnerabilidad de seguridad.

Refuerzo de la seguridad de los componentes de Zabbix

Algunas funciones se pueden desactivar para reforzar la seguridad de los componentes de Zabbix:

- La ejecución global de scripts en Zabbix server se puede desactivar estableciendo `EnableGlobalScripts=0` en la configuración del server.
- La ejecución global de scripts en Zabbix proxy está desactivada de forma predeterminada (se puede activar estableciendo `EnableRemoteCommands=1` en la configuración del proxy).
- La ejecución global de scripts en los agentes de Zabbix está desactivada de forma predeterminada (se puede activar añadiendo un parámetro `AllowKey=system.run[<command>,*]` o `AllowKeyRegexp=^system\.run\[.*\]$` para cada comando permitido en la configuración del agent).
- El acceso a la configuración de la autenticación HTTP de los usuarios está activado de forma predeterminada; se puede desactivar estableciendo `$ZBX_FEATURE_FLAGS['http_auth_enabled'] = false` en la configuración del frontend.
- El acceso a la configuración de los módulos está activado de forma predeterminada; se puede desactivar estableciendo `$ZBX_FEATURE_FLAGS['modules_config_enabled'] = false` en la configuración del frontend.
- La edición de tipos de medios específicos se puede restringir estableciendo `$ZBX_FEATURE_FLAGS['media_type_denylist'] = [<value1>, <value2>, ...]` en la configuración del frontend. Valores posibles: `'email'`, `'script'`, `'sms'`, `'webhook'` y `'push'`. Se pueden especificar uno o varios valores.

La reinstalación del frontend mediante la ejecución de `setup.php` está restringida si existe el archivo de configuración `zabbix.conf.php`.

Acceso a rutas UNC en Windows mediante Zabbix agent

Los Zabbix agent en Windows siguen rutas UNC (recursos compartidos SMB como `\\server\share\file.txt`) en items como `vfs.file.*`, `vfs.dir.*`, `modbus.get` y `perf_counter*`. Esto puede suponer un riesgo de seguridad en algunos contextos.

Cuando se solicita a Windows que acceda a una ruta UNC, intenta autenticarse en ese server. Esto significa que una solicitud maliciosa al Zabbix agent puede exponer el hash NTLM al server del solicitante. Los usuarios pueden mitigar este riesgo configurando según sea necesario los parámetros de configuración AllowKey, AllowKeyRegexp, DenyKey y DenyKeyRegexp.

Asegurar MySQL/MariaDB

Descripción general

Esta sección contiene las mejores prácticas para proteger una base de datos MySQL/MariaDB.

> **Note:**
> Para una configuración básica, consulte las instrucciones predeterminadas de creación de base de datos MySQL/MariaDB, que incluyen la creación del usuario 'zabbix' con privilegios completos sobre la base de datos de Zabbix. Este usuario es el propietario de la base de datos y también tiene los privilegios necesarios para modificar la estructura de la base de datos al actualizar Zabbix.

Para mejorar la seguridad, se recomienda crear roles y usuarios adicionales de base de datos con privilegios mínimos. Estos roles y usuarios deben configurarse según el principio de privilegio mínimo, es decir, solo deben tener los privilegios esenciales para realizar las funciones previstas.

Creación de roles de usuario

Cree los siguientes roles con los privilegios correspondientes:

- **zbx_srv** - rol para ejecutar Zabbix server y proxy:

```
CREATE ROLE 'zbx_srv';
GRANT DELETE, INSERT, SELECT, UPDATE ON zabbix.* TO 'zbx_srv';
FLUSH PRIVILEGES;
```

- **zbx_web** - rol para ejecutar Zabbix frontend y API:

```
CREATE ROLE 'zbx_web';
GRANT DELETE, INSERT, SELECT, UPDATE ON zabbix.* TO 'zbx_web';
FLUSH PRIVILEGES;
```

- **zbx_bckp** - rol para copia de seguridad de tablas:

```
CREATE ROLE 'zbx_bckp';
GRANT LOCK TABLES, TRIGGER, SELECT ON zabbix.* TO 'zbx_bckp';
GRANT process ON *.* TO 'zbx_bckp';
FLUSH PRIVILEGES;
```

> **Note:**
> La restauración y la actualización de tablas deben ser realizadas por el propietario de la base de datos.

- **zbx_part** - rol con un conjunto reducido de privilegios para la partición de la base de datos; tenga en cuenta que este rol solo puede crearse después de haber creado la base de datos, ya que concede privilegios sobre tablas específicas de la base de datos:

```
CREATE ROLE 'zbx_part';
GRANT SELECT, ALTER, DROP ON zabbix.history TO 'zbx_part';
GRANT SELECT, ALTER, DROP ON zabbix.history_uint TO 'zbx_part';
GRANT SELECT, ALTER, DROP ON zabbix.history_str TO 'zbx_part';
GRANT SELECT, ALTER, DROP ON zabbix.history_text TO 'zbx_part';
GRANT SELECT, ALTER, DROP ON zabbix.history_log TO 'zbx_part';
GRANT SELECT, ALTER, DROP ON zabbix.trends TO 'zbx_part';
GRANT SELECT, ALTER, DROP ON zabbix.trends_uint TO 'zbx_part';
-- For MariaDB: skip the next line (GRANT session_variables_admin ON *.* TO 'zbx_part';)
GRANT session_variables_admin ON *.* TO 'zbx_part';
GRANT SELECT ON zabbix.dbversion TO 'zbx_part';
GRANT SELECT, DELETE ON zabbix.housekeeper TO 'zbx_part';
FLUSH PRIVILEGES;
```

Después de crear los roles, se pueden asignar a los usuarios.

Asignación de roles de usuario

Para asignar los roles de usuario creados, cree usuarios y asígneles los roles correspondientes.
Reemplace <user>, <host>, <role> y <password> por los valores adecuados.
Asegúrese de que <password> sea segura y única.

```
CREATE USER '<user>'@'<host>' IDENTIFIED BY '<password>';
GRANT '<role>' TO '<user>'@'<host>';
SET DEFAULT ROLE '<role>' TO '<user>'@'<host>';
-- Para MariaDB: SET DEFAULT ROLE '<role>' FOR '<user>'@'<host>'
FLUSH PRIVILEGES;
```

Por ejemplo, para crear y asignar el rol para ejecutar Zabbix server y proxy:

```
CREATE USER 'usr_srv'@'localhost' IDENTIFIED BY 'strong-password';
GRANT 'zbx_srv' TO 'usr_srv'@'localhost';
SET DEFAULT ROLE ALL TO 'usr_srv'@'localhost';
FLUSH PRIVILEGES;
```

Asegurar PostgreSQL/TimescaleDB

Descripción general

Esta sección contiene las mejores prácticas para proteger una base de datos PostgreSQL.

> **Note:**
> Para una configuración básica, consulte las instrucciones predeterminadas de creación de la base de datos PostgreSQL, que incluyen la creación del usuario 'zabbix' con privilegios completos sobre la base de datos de Zabbix. Este usuario es el propietario de la base de datos y también tiene los privilegios necesarios para modificar la estructura de la base de datos al actualizar Zabbix.

Para mejorar la seguridad, se recomienda configurar un patrón de uso seguro del esquema, así como crear roles y usuarios adicionales de base de datos con privilegios mínimos. Estos roles y usuarios deben configurarse según el principio de privilegio mínimo, es decir, solo deben tener los privilegios que sean esenciales para realizar las funciones previstas.

Configuración de la base de datos

Cree el usuario que será el propietario de la base de datos y cree la base de datos Zabbix; el propietario de la base de datos es el usuario que se especifica al crear la base de datos:

```
createuser -U postgres -h localhost --pwprompt usr_owner
createdb -U postgres -h localhost -O usr_owner -E Unicode -T template0 zabbix
```

> **Note:**
> Una instalación limpia o una actualización de la base de datos debe ser realizada por el propietario de la base de datos. Esto se debe a que el derecho a eliminar un objeto de la base de datos o a modificar su definición es un privilegio inherente al propietario de la base de datos y no puede concederse ni revocarse.

> **Attention:**
> Los siguientes comandos de esta página deben ejecutarse mientras la conexión a PostgreSQL se realiza específicamente a la base de datos `zabbix`.

Cree el esquema `zabbix` y establezca al propietario de la base de datos (`usr_owner`) como propietario de este esquema:

```
CREATE SCHEMA zabbix AUTHORIZATION usr_owner;
```

Configure un patrón de uso seguro del esquema:

```
REVOKE CREATE ON SCHEMA public FROM PUBLIC;
REVOKE ALL ON DATABASE zabbix FROM PUBLIC;
-- Note: search_path should point to the "zabbix" schema:
ALTER ROLE ALL IN DATABASE zabbix SET search_path = "zabbix";
```

Después de configurar la base de datos, continúe con la creación de roles de usuario.

Creación de roles de usuario

Cree los siguientes roles con los privilegios correspondientes:

- **zbx_srv** - rol para ejecutar Zabbix server y proxy:

```
CREATE ROLE zbx_srv;
GRANT CONNECT ON DATABASE zabbix TO zbx_srv;
```

```
GRANT USAGE ON SCHEMA zabbix TO zbx_srv;
ALTER DEFAULT PRIVILEGES FOR ROLE usr_owner IN SCHEMA zabbix GRANT DELETE, INSERT, SELECT, UPDATE ON TABLE
ALTER DEFAULT PRIVILEGES FOR ROLE usr_owner IN SCHEMA zabbix GRANT SELECT, UPDATE, USAGE ON sequences TO z
```

- **zbx_web** - rol para ejecutar Zabbix frontend y API:

```
CREATE ROLE zbx_web;
GRANT CONNECT ON DATABASE zabbix TO zbx_web;
GRANT USAGE ON SCHEMA zabbix TO zbx_web;
ALTER DEFAULT PRIVILEGES FOR ROLE usr_owner IN SCHEMA zabbix GRANT DELETE, INSERT, SELECT, UPDATE ON TABLE
ALTER DEFAULT PRIVILEGES FOR ROLE usr_owner IN SCHEMA zabbix GRANT SELECT, UPDATE, USAGE ON sequences TO z
```

- **zbx_bckp** - rol para la copia de seguridad de tablas:

```
CREATE ROLE zbx_bckp;
GRANT CONNECT ON DATABASE zabbix TO zbx_bckp;
GRANT USAGE ON SCHEMA zabbix TO zbx_bckp;
ALTER DEFAULT PRIVILEGES FOR ROLE usr_owner IN SCHEMA zabbix GRANT SELECT ON TABLES TO zbx_bckp;
ALTER DEFAULT PRIVILEGES FOR ROLE usr_owner IN SCHEMA zabbix GRANT SELECT, UPDATE, USAGE ON sequences TO z
```

Note:
La restauración de tablas solo es posible por el propietario de la base de datos.

Después de crear los roles, se pueden asignar a los usuarios.

Asignación de roles de usuario

Para asignar los roles de usuario creados, cree usuarios y asígneles los roles correspondientes.
Reemplace <user>, <role> y <password> por los valores adecuados.
Asegúrese de que <password> sea segura y única.

```
CREATE USER <user> WITH ENCRYPTED password '<password>';
GRANT <role> TO <user>;
```

Por ejemplo, para crear y asignar el rol para ejecutar Zabbix server y proxy:

```
CREATE USER usr_srv WITH ENCRYPTED password 'strong-password';
GRANT zbx_srv TO usr_srv;
```

Particionado de base de datos con TimescaleDB

El particionado de la base de datos está facilitado por TimescaleDB. Para utilizar TimescaleDB, el server de Zabbix requiere privilegios de propietario de la base de datos.

Si el esquema PostgreSQL zabbix ya se ha creado en la base de datos zabbix, puede habilitar TimescaleDB con el siguiente comando:

```
echo "CREATE EXTENSION IF NOT EXISTS timescaledb WITH SCHEMA zabbix CASCADE;" | sudo -u postgres psql zabb
```

Criptografía

Descripción general

Esta sección contiene las mejores prácticas para configurar la criptografía de forma segura.

Configuración de SSL para el frontend de Zabbix

En sistemas basados en RHEL, instale el paquete mod_ssl:

```
dnf install mod_ssl
```

Cree un directorio para las claves SSL:

```
mkdir -p /etc/httpd/ssl/private
chmod 700 /etc/httpd/ssl/private
```

Cree el certificado SSL:

```
openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/httpd/ssl/private/apache-selfsigned.key -
```

Complete los campos solicitados de forma adecuada. La línea más importante es la que solicita el `Common Name`. Debe introducir el nombre de dominio que desea asociar con su server. Puede introducir la dirección IP pública en su lugar si no dispone de un nombre de dominio.

```
Country Name (2 letter code) [XX]:
State or Province Name (full name) []:
Locality Name (eg, city) [Default City]:
Organization Name (eg, company) [Default Company Ltd]:
Organizational Unit Name (eg, section) []:
Common Name (eg, your name or your server's hostname) []:example.com
Email Address []:
```

Edite el archivo de configuración SSL de Apache (`/etc/httpd/conf.d/ssl.conf`):

```
DocumentRoot "/usr/share/zabbix"
ServerName example.com:443
SSLCertificateFile /etc/httpd/ssl/apache-selfsigned.crt
SSLCertificateKeyFile /etc/httpd/ssl/private/apache-selfsigned.key
```

Reinicie el servicio de Apache para aplicar los cambios:

```
systemctl restart httpd.service
```

Asegurar el web server

Descripción general

Esta sección contiene las mejores prácticas para configurar el web server de forma segura.

Forzar la redirección de la URL raíz a Zabbix SSL

En sistemas basados en RHEL, agregue un host virtual a la configuración de Apache (`/etc/httpd/conf/httpd.conf`) y establezca una redirección permanente para la raíz del documento hacia la URL SSL de Zabbix. Tenga en cuenta que `example.com` debe reemplazarse por el nombre real del server.

```
### Add lines:

<VirtualHost *:*>
    ServerName example.com
    Redirect permanent / https://example.com
</VirtualHost>
```

Reinicie el servicio de Apache para aplicar los cambios:

```
systemctl restart httpd.service
```

Habilitación de HTTP Strict Transport Security (HSTS) en el web server

Para proteger el frontend de Zabbix frente a ataques de degradación de protocolo, recomendamos habilitar la directiva HSTS en el web server.

Para habilitar la directiva HSTS para su frontend de Zabbix en la configuración de Apache, siga estos pasos:

1. Localice el archivo de configuración de su host virtual:
   - `/etc/httpd/conf/httpd.conf` en sistemas basados en RHEL
   - `/etc/apache2/sites-available/000-default.conf` en Debian/Ubuntu

2. Añada la siguiente directiva al archivo de configuración de su host virtual:

```
<VirtualHost *:*>
    Header set Strict-Transport-Security "max-age=31536000"
</VirtualHost>
```

3. Reinicie el servicio de Apache para aplicar los cambios:

```
### En sistemas basados en RHEL:
systemctl restart httpd.service

### En Debian/Ubuntu
systemctl restart apache2.service
```

Aplicación de cookies de sesión seguras y SameSite en Zabbix

Al configurar Zabbix, es esencial aplicar los atributos secure y SameSite a las cookies de sesión para mejorar la seguridad y evitar ataques de falsificación de solicitudes entre sitios (CSRF). Sin embargo, aplicar `SameSite=Strict` puede causar problemas en ciertos escenarios, como:

- Los widgets de URL del dashboard muestran "user not logged in" al incrustar iframes del mismo dominio.
- Los usuarios que acceden al dashboard mediante HTTP en lugar de HTTPS pueden tener problemas de inicio de sesión.
- Imposibilidad de compartir URLs a secciones específicas del menú de Zabbix o a hosts.

Para mitigar estos problemas, los usuarios deberían tener una forma de ajustar la política de SameSite.

1. Cookies seguras

Establecer la marca `secure` garantiza que las cookies solo se transmitan a través de HTTPS, evitando su exposición en conexiones no cifradas.

Para habilitar cookies seguras en Zabbix, agregue o modifique la siguiente configuración en la configuración del servidor web:

Para Apache:

```
Header always edit Set-Cookie ^(.*)$ $1;Secure
```

Para Nginx:

```
proxy_cookie_path / "/; Secure";
```

Asegúrese de que su frontend de Zabbix se acceda mediante HTTPS; de lo contrario, las cookies con la marca `Secure` no se enviarán.

2. Configuración del atributo SameSite

La configuración del servidor web también puede aplicar el atributo SameSite:

Para Apache:

```
<IfModule mod_headers.c>
    Header onsuccess edit Set-Cookie (.*) "$1; SameSite=Strict"
</IfModule>
```

Para Nginx (versión 1.19.3+):

```
proxy_cookie_flags ~ samesite=Strict; # Reemplace ~ con 'zbx_session' para mayor especificidad
```

Habilitar Content Security Policy (CSP) en el web server

Para proteger el frontend de Zabbix contra Cross Site Scripting (XSS), inyección de datos y ataques similares, recomendamos habilitar Content Security Policy en el web server. Para ello, configure el web server para que devuelva el HTTP header.

> **Attention:**
> La siguiente configuración del encabezado CSP solo es válida para la instalación predeterminada del frontend de Zabbix y para los casos en que todo el contenido proviene del dominio del sitio (excluyendo subdominios). Puede ser necesaria una configuración diferente del encabezado CSP si, por ejemplo, está configurando el widget *URL* para mostrar contenido de subdominios del sitio o de dominios externos, si cambia de *OpenStreetMap* a otro motor de mapas, o si agrega CSS o widgets externos. Si utiliza el método de autenticación multifactor Duo Universal Prompt, asegúrese de agregar "duo.com" a la directiva CSP en el archivo de configuración de su virtual host.

Para habilitar CSP para su frontend de Zabbix en la configuración de Apache, siga estos pasos:

1. Localice el archivo de configuración de su virtual host:

- `/etc/httpd/conf/httpd.conf` en sistemas basados en RHEL
- `/etc/apache2/sites-available/000-default.conf` en Debian/Ubuntu

2. Añada la siguiente directiva al archivo de configuración de su virtual host:

```
<VirtualHost *:*>
    Header set Content-Security-Policy: "default-src 'self' *.openstreetmap.org; script-src 'self' 'unsafe
</VirtualHost>
```

3. Reinicie el servicio de Apache para aplicar los cambios:

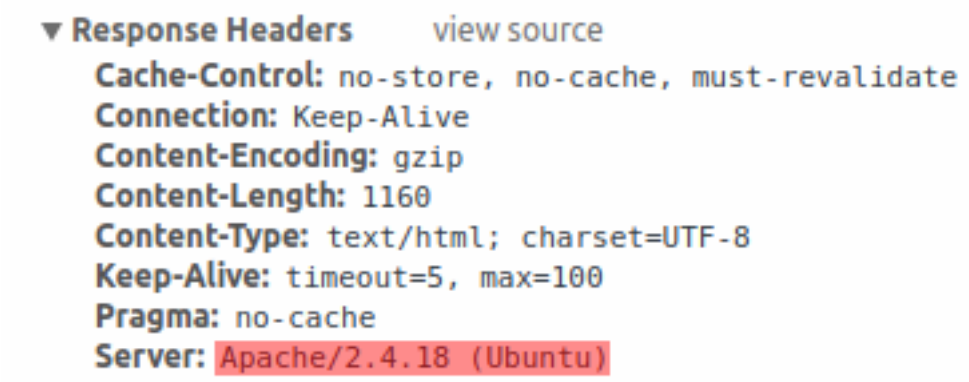
```
### En sistemas basados en RHEL:
systemctl restart httpd.service

### En Debian/Ubuntu
systemctl restart apache2.service
```

Deshabilitar la exposición de información del servidor web

Para mejorar la seguridad, se recomienda deshabilitar todas las firmas del servidor web.

De forma predeterminada, el servidor web expone la firma del software:



La firma se puede deshabilitar agregando los siguientes parámetros al archivo de configuración de Apache:

```
ServerSignature Off
ServerTokens Prod
```

La firma de PHP (encabezado HTTP X-Powered-By) se puede deshabilitar modificando el archivo de configuración `php.ini` (de forma predeterminada, la firma está deshabilitada):

```
expose_php = Off
```

Es necesario reiniciar el servidor web para que se apliquen los cambios en el archivo de configuración.

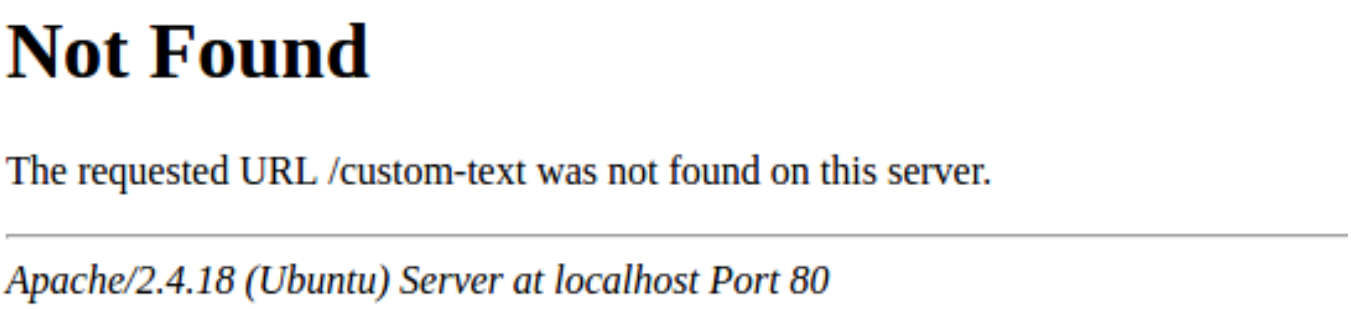
Para obtener seguridad adicional, puede utilizar la herramienta *mod_security* con Apache (paquete *libapache2-mod-security2*). Esta herramienta permite eliminar la firma del servidor en lugar de eliminar únicamente la versión de la firma del servidor. La firma del servidor se puede cambiar a cualquier valor estableciendo "SecServerSignature" en el valor deseado después de instalar *mod_security*.

Consulte la documentación de su servidor web para obtener información sobre cómo eliminar o cambiar las firmas del software.

Deshabilitar las páginas de error predeterminadas del servidor web

Para evitar la exposición de información, se recomienda deshabilitar las páginas de error predeterminadas.

De forma predeterminada, un servidor web utiliza páginas de error integradas:



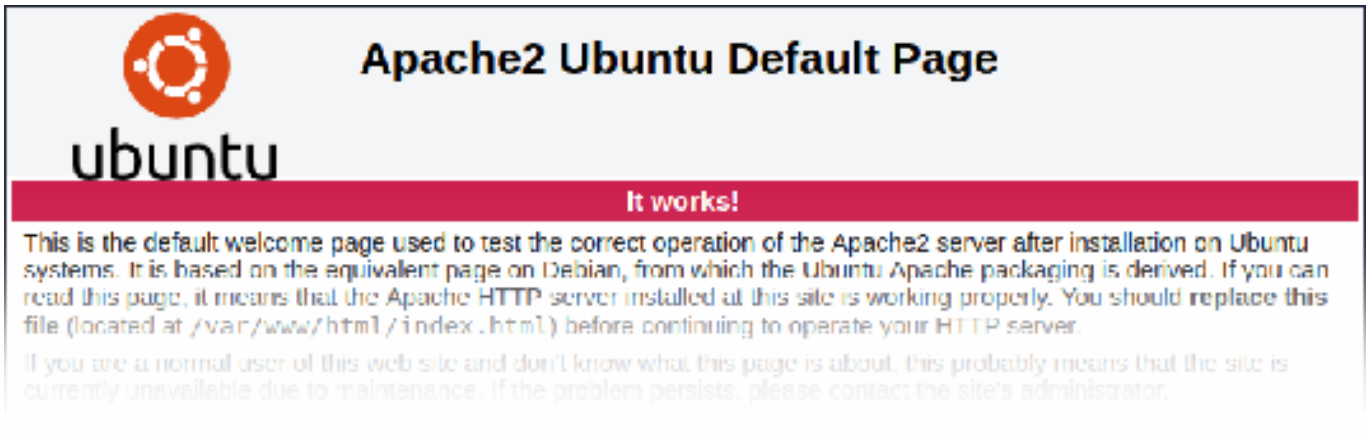
Estas páginas de error predeterminadas deben reemplazarse o eliminarse. Por ejemplo, la directiva "ErrorDocument" se puede utilizar para definir una página o texto de error personalizado para el servidor web Apache.

Consulte la documentación de su servidor web para obtener información sobre cómo reemplazar o eliminar las páginas de error predeterminadas.

Eliminación de la página de prueba del web server

Para evitar la exposición de información, se recomienda eliminar la página de prueba del web server.

De forma predeterminada, el webroot del web server Apache contiene la página de prueba `index.html`:



Consulte la documentación de su web server para obtener ayuda sobre cómo eliminar las páginas de prueba predeterminadas.

Establecer el encabezado HTTP X-Frame-Options

De forma predeterminada, Zabbix está configurado con el parámetro *Use X-Frame-Options HTTP header* establecido en `SAMEORIGIN`. Esto significa que el contenido solo puede cargarse en un marco que tenga el mismo origen que la propia página.

Los elementos del frontend de Zabbix que obtienen contenido de URL externas (es decir, el widget de dashboard URL) muestran el contenido recuperado en un sandbox con todas las restricciones de sandboxing habilitadas.

Estos ajustes mejoran la seguridad del frontend de Zabbix y proporcionan protección contra ataques XSS y clickjacking. Los usuarios *Super admin* pueden modificar los parámetros *Use iframe sandboxing* y *Use X-Frame-Options HTTP header* según sea necesario. Considere cuidadosamente los riesgos y beneficios antes de cambiar la configuración predeterminada. No se recomienda desactivar por completo el sandboxing de iframe ni el encabezado HTTP X-Frame-Options.

Ocultar el archivo con la lista de contraseñas comunes

Para aumentar la complejidad de los ataques de fuerza bruta de contraseñas, se recomienda limitar el acceso al archivo `ui/data/top_passwords.txt`. Este archivo contiene una lista de las contraseñas más comunes y específicas del contexto, y evita que los usuarios establezcan dichas contraseñas (si el parámetro *Avoid easy-to-guess passwords* está habilitado en la password policy).

Para limitar el acceso al archivo `top_passwords.txt`, modifique la configuración de su servidor web.

En Apache, el acceso al archivo se puede limitar mediante el archivo `.htaccess`:

```
<Files "top_passwords.txt">
    Order Allow,Deny
    Deny from all
</Files>
```

En NGINX, el acceso al archivo se puede limitar mediante la directiva `location`:

```
location = /data/top_passwords.txt {
    deny all;
    return 404;
}
```

Almacenamiento de secretos

Resumen Zabbix puede configurarse para recuperar información sensible de un almacén seguro. Se admiten los siguientes servicios de gestión de secretos: HashiCorp Vault KV Secrets Engine - Version 2, CyberArk Vault CV12.

Los secretos se pueden usar para recuperar:

- valores de macros de usuario
- credenciales de acceso a la base de datos

Zabbix proporciona acceso de solo lectura a los secretos en un almacén, suponiendo que los secretos son gestionados por otra persona.

Para obtener información sobre la configuración específica del proveedor de vault, consulte:

- Configuración de HashiCorp
- Configuración de CyberArk

Almacenamiento en caché de valores secretos De forma predeterminada, los valores de las macros secretas de vault son recuperados por Zabbix server cada vez que se actualizan los datos de configuración y, posteriormente, se almacenan en la caché de configuración. Zabbix proxy recibe los valores de las macros secretas de vault de Zabbix server en cada sincronización de la configuración y los almacena en su propia caché de configuración.

> **Attention:**
> El cifrado debe estar habilitado entre Zabbix server y proxy; de lo contrario, se registra un mensaje de advertencia en el server.

También es posible configurar que Zabbix server y Zabbix proxy recuperen los valores de las macros de forma independiente.

Para activar manualmente la actualización de los valores secretos almacenados en caché desde un vault, utilice la opción de línea de comandos `secrets_reload`.

El almacenamiento en caché de las credenciales de la base de datos del frontend de Zabbix está deshabilitado de forma predeterminada, pero puede habilitarse estableciendo la opción `$DB['VAULT_CACHE'] = true` en el archivo de configuración del frontend. Las credenciales se almacenarán en una caché local mediante el directorio de archivos temporales del sistema de archivos. El web server debe permitir la escritura en una carpeta temporal privada (por ejemplo, en Apache debe establecerse la opción de configuración `PrivateTmp=True`). Para controlar con qué frecuencia se actualiza o invalida la caché de datos, utilice la constante `ZBX_DATA_CACHE_TTL`.

Configuración de TLS Para configurar TLS para la comunicación entre los componentes de Zabbix y el vault, agregue un certificado firmado por una autoridad de certificación (CA) al almacén de CA predeterminado de todo el sistema. Para usar otra ubicación, especifique el directorio en el parámetro de configuración SSLCALocation de Zabbix server/proxy, coloque el archivo del certificado dentro de ese directorio y luego ejecute el comando de CLI:

```
c_rehash .
```

Configuración de CyberArk

Esta sección explica cómo configurar Zabbix para recuperar secretos de CyberArk Vault CV12.

El vault debe instalarse y configurarse como se describe en la documentación oficial de CyberArk.

Para obtener información sobre la configuración de TLS en Zabbix, consulte *Almacenamiento de secretos*.

Credenciales de la base de datos

El acceso a un secreto con las credenciales de la base de datos se configura por separado para cada componente de Zabbix.

Server y proxies

Para obtener las credenciales de la base de datos del almacén de secretos para el server o el proxy de Zabbix, especifique los siguientes parámetros de configuración en el archivo de configuración:

- `Vault`: qué proveedor de almacén de secretos se debe utilizar;
- `VaultURL`: URL HTTP(S) del server del almacén de secretos;
- `VaultDBPath`: consulta al secreto del almacén de secretos que contiene las credenciales de la base de datos, que se recuperarán mediante las claves ”Content” y ”UserName” (esta opción solo se puede utilizar si no se especifican DBUser y DBPassword);
- `VaultTLSCertFile`, `VaultTLSKeyFile`: nombres de los archivos del certificado y la clave SSL; no es obligatorio configurar estas opciones, pero se recomienda encarecidamente;

- `VaultPrefix`: prefijo personalizado para la ruta o consulta del almacén de secretos, según el almacén; si no se especifica, se utilizará el valor predeterminado más adecuado.

Attention:
Los parámetros de configuración `Vault`, `VaultURL`, `VaultTLSCertFile`, `VaultTLSKeyFile` y `VaultPrefix` también se utilizan para la autenticación en el almacén de secretos al procesar macros de secretos del almacén mediante el server de Zabbix (y el proxy de Zabbix, si está configurado). El server de Zabbix y los proxies no abrirán las macros de secretos del almacén que contengan credenciales de la base de datos de VaultDBPath.

El server de Zabbix y el proxy de Zabbix leen los parámetros de configuración relacionados con el almacén de secretos de los archivos `zabbix_server.conf` y `zabbix_proxy.conf` al iniciarse.

Ejemplo

1. En `zabbix_server.conf`, especifique los siguientes parámetros:

```
Vault=CyberArk
VaultURL=https://127.0.0.1:1858
VaultDBPath=AppID=zabbix_server&Query=Safe=passwordSafe;Object=zabbix_server_database
VaultTLSCertFile=cert.pem
VaultTLSKeyFile=key.pem
VaultPrefix=/AIMWebService/api/Accounts?
```

2. Zabbix enviará la siguiente solicitud API al vault:

```
curl \
--header "Content-Type: application/json" \
--cert cert.pem \
--key key.pem \
https://127.0.0.1:1858/AIMWebService/api/Accounts?AppID=zabbix_server&Query=Safe=passwordSafe;Object=zabbi
```

3. La respuesta del vault contendrá las claves ”Content” y ”UserName”:

```
{
    "Content": <password>,
    "UserName": <username>,
    "Address": <address>,
    "Database": <Database>,
    "PasswordChangeInProcess":<PasswordChangeInProcess>
}
```

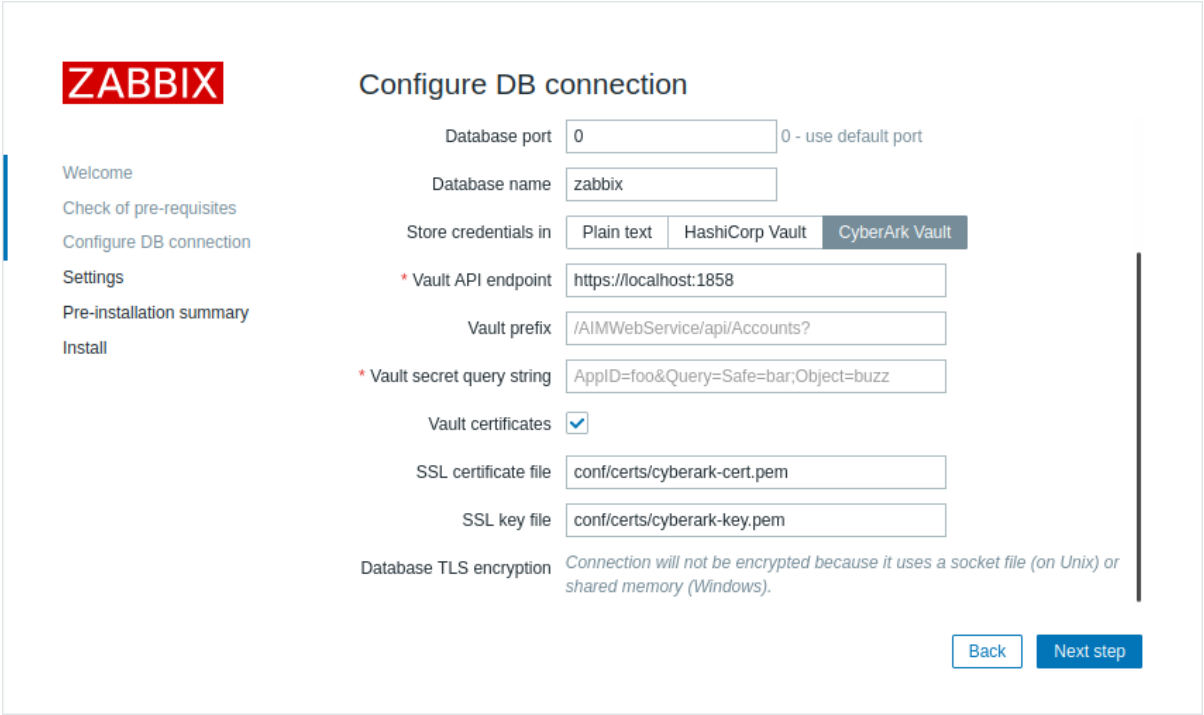
4. Como resultado, Zabbix utilizará las siguientes credenciales para la autenticación de la base de datos:

- Nombre de usuario: <username>
- Contraseña: <password>

frontend

Para obtener las credenciales de la base de datos desde el vault para el frontend de Zabbix, especifique los siguientes parámetros durante la instalación del frontend.

1. En el paso *Configure DB Connection*, establezca el parámetro *Store credentials in* en *CyberArk Vault*.



2. A continuación, complete los parámetros adicionales:

Parameter	Mandatory	Default value	Description
Vault API endpoint	yes	https://localhost:1858	Especifique la URL para conectarse al vault con el formato `scheme://host:port`
Vault prefix	no	/AIMWebService/api/Accounts?	Proporcione un prefijo personalizado para la ruta o consulta del vault. Si no se especifica, se utiliza el valor predeterminado.
Vault secret query string	yes		Consulta que especifica desde dónde deben recuperarse las credenciales de la base de datos. Ejemplo: `AppID=foo&Query=Safe=bar;Object=buzz`
Vault certificates	no		Después de marcar la casilla, aparecerán parámetros adicionales que le permitirán configurar la autenticación del cliente. Aunque este parámetro es opcional, se recomienda encarecidamente habilitarlo para la comunicación con CyberArk Vault.
SSL certificate file	no	conf/certs/cyberark-cert.pem	Ruta al archivo de certificado SSL. El archivo debe estar en formato PEM. Si el archivo de certificado también contiene la clave privada, deje vacío el parámetro del archivo de clave SSL.
SSL key file	no	conf/certs/cyberark-key.pem	Nombre del archivo de clave privada SSL utilizado para la autenticación del cliente. El archivo debe estar en formato PEM.

Valores de macros de usuario

Para usar CyberArk Vault para almacenar valores de macros de usuario de *Vault secret*, asegúrese de que:

- Zabbix server esté configurado para funcionar con CyberArk Vault;
- el parámetro *Vault provider* en *Administration > General > Other* esté establecido en "CyberArk Vault".



Note:
Zabbix server (y Zabbix proxy, si está configurado) requieren acceso a los valores de macros *Vault secret* desde el vault. Zabbix frontend no necesita dicho acceso.

El valor de la macro debe contener una consulta (como `query:key`).

Consulte *Vault secret macros* para obtener información detallada sobre el procesamiento de los valores de macro por parte de Zabbix.

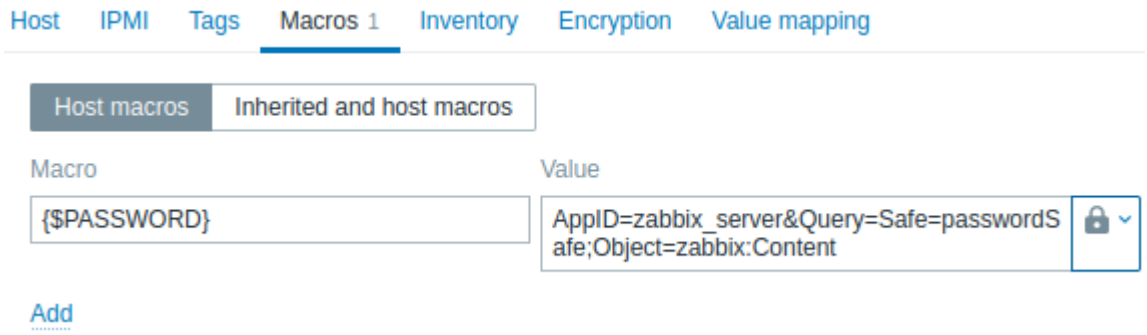
Sintaxis de consulta

El símbolo de dos puntos (`:`) está reservado para separar la consulta de la clave.

Si la consulta contiene una barra diagonal o dos puntos, estos símbolos deben codificarse mediante URL (`/` se codifica como `%2F`, `:` se codifica como `%3A`).

Ejemplo

1. En Zabbix, añada una macro de usuario {$PASSWORD} de tipo *Vault secret* y con el valor `AppID=zabbix_server&Query=Safe=pass`



2. Zabbix enviará la siguiente solicitud de API al vault:

```
curl \
--header "Content-Type: application/json" \
--cert cert.pem \
--key key.pem \
https://127.0.0.1:1858/AIMWebService/api/Accounts?AppID=zabbix_server&Query=Safe=passwordSafe;Object=zabbi
```

3. La respuesta del vault contendrá la clave `Content`:

```
{
    "Content": <password>,
    "UserName": <username>,
    "Address": <address>,
    "Database" :<Database>,
    "PasswordChangeInProcess":<PasswordChangeInProcess>
}
```

4. Como resultado, Zabbix resolverá la macro {$PASSWORD} con el valor <password>

Actualizar la configuración existente

Para actualizar una configuración existente para recuperar secretos de un CyberArk Vault:

1. Actualice los parámetros del archivo de configuración del Zabbix server o proxy, tal como se describe en la sección *Credenciales de la base de datos*.

2. Actualice los parámetros del archivo de configuración del frontend de Zabbix:

```
$DB['VAULT']                    = 'CyberArk';
$DB['VAULT_URL']                = 'https://127.0.0.1:1858';
$DB['VAULT_DB_PATH']            = 'AppID=foo&Query=Safe=bar;Object=buzz';
$DB['VAULT_TOKEN']              = '';
$DB['VAULT_CERT_FILE']          = 'conf/certs/cyberark-cert.pem';
$DB['VAULT_KEY_FILE']           = 'conf/certs/cyberark-key.pem';
$DB['VAULT_PREFIX']             = '';
```

3. Configure las macros de usuario tal como se describe en la sección *Valores de las macros de usuario*, si es necesario.

Para actualizar una configuración existente para recuperar secretos de un HashiCorp Vault, consulte *Configuración de HashiCorp*.

Configuración de HashiCorp

Descripción general

Esta sección explica cómo configurar Zabbix para recuperar secretos de HashiCorp Vault KV Secrets Engine - Version 2.

El vault debe implementarse y configurarse como se describe en la documentación oficial de HashiCorp.

Para obtener información sobre cómo configurar TLS en Zabbix, consulte *Storage of secrets*.

Recuperación de credenciales de la base de datos

Para recuperar correctamente un secreto con credenciales de la base de datos, es necesario configurar ambos elementos:

- Zabbix server/proxy
- Zabbix frontend

Server/proxy

Para configurar el server o el proxy de Zabbix, especifique los siguientes parámetros de configuración en el archivo de configuración:

- `Vault`: proveedor de vault que se debe utilizar.
- `VaultToken`: token de autenticación del vault (consulte el archivo de configuración del server/proxy de Zabbix para obtener más información); no se puede utilizar con `VaultAppRoleID`.
- `VaultAppRoleID` y `VaultAppSecretID`: ID de rol y ID secreto de autenticación del vault (consulte el archivo de configuración del server/proxy de Zabbix para obtener más información); no se pueden utilizar con `VaultToken`.
- `VaultURL`: URL HTTP(S) del server de vault.
- `VaultDBPath`: ruta al secreto del vault que contiene las credenciales de la base de datos (esta opción solo se puede utilizar si no se especifican DBUser y DBPassword). El server o proxy de Zabbix recuperará las credenciales mediante las claves "password" y "username".
- `VaultPrefix`: prefijo personalizado para la ruta o consulta del vault, según el vault; si no se especifica, se utilizará el valor predeterminado más adecuado. Tenga en cuenta que el prefijo debe comenzar y terminar con `/`.

Attention:
Los parámetros de configuración `Vault`, `VaultToken` (o `VaultAppRoleID`/`VaultAppSecretID`), `VaultURL` y `VaultPrefix` también se utilizan para la autenticación del vault al procesar macros secretas del vault mediante el server de Zabbix (y el proxy de Zabbix, si está configurado). El server de Zabbix y los proxies no abrirán las macros secretas del vault que contengan credenciales de la base de datos de VaultDBPath.

Se recomienda encarecidamente utilizar tokens diferentes para distintos proxies.

El server de Zabbix y el proxy de Zabbix leen los parámetros de configuración relacionados con el vault de `zabbix_server.conf` y `zabbix_proxy.conf` durante el inicio. Además, el server de Zabbix y el proxy de Zabbix leerán la variable de entorno `VAULT_TOKEN` una vez durante el inicio y la desactivarán para que no esté disponible mediante scripts bifurcados; se producirá un error si los parámetros `VaultToken` y `VAULT_TOKEN` contienen un valor.

Ejemplo

1. En `zabbix_server.conf`, especifique los siguientes parámetros.

Usando VaultToken:

```
Vault=HashiCorp
VaultToken=hvs.CAESIIG_PILmULFYOsEyWHxkZ2mF2a8VPKNLE8eHqd4autYGGh4KHGh2cy5aeTY0NFNSaUp3ZnpWbDF1RUNjUkNTZEg
VaultURL=https://127.0.0.1:8200
VaultDBPath=database
VaultPrefix=/v1/secret/data/zabbix/
```

O, usando VaultAppRoleID:

```
Vault=HashiCorp
VaultAppRoleID=6f1567f1-538d-7d70-e8e4-5d3e59471b2e
VaultAppSecretID=89dac8b1-f7d6-96a7-2ecf-f94d2de3d6f1
VaultURL=https://127.0.0.1:8200
VaultDBPath=database
VaultPrefix=/v1/secret/data/zabbix/
```

2. Ejecute los siguientes comandos CLI para crear el secreto requerido en el vault:

```
#### Habilite el punto de montaje "secret/" si aún no está habilitado; tenga en cuenta que debe usarse "ku
vault secrets enable -path=secret/ kv-v2

#### Coloque nuevos secretos con las claves username y password bajo el punto de montaje "secret/" y la ru
vault kv put -mount=secret zabbix/database username=zabbix password=<password>

#### Compruebe que el secreto se haya agregado correctamente.
vault kv get secret/zabbix/database

#### Finalmente, pruebe con Curl; tenga en cuenta que "data" debe agregarse manualmente después del punto
curl --header "X-Vault-Token: <VaultToken>" https://127.0.0.1:8200/v1/secret/data/zabbix/database
```

3. Como resultado, Zabbix server recuperará las siguientes credenciales para la autenticación de la base de datos:

- Usuario: zabbix
- Contraseña: <password>

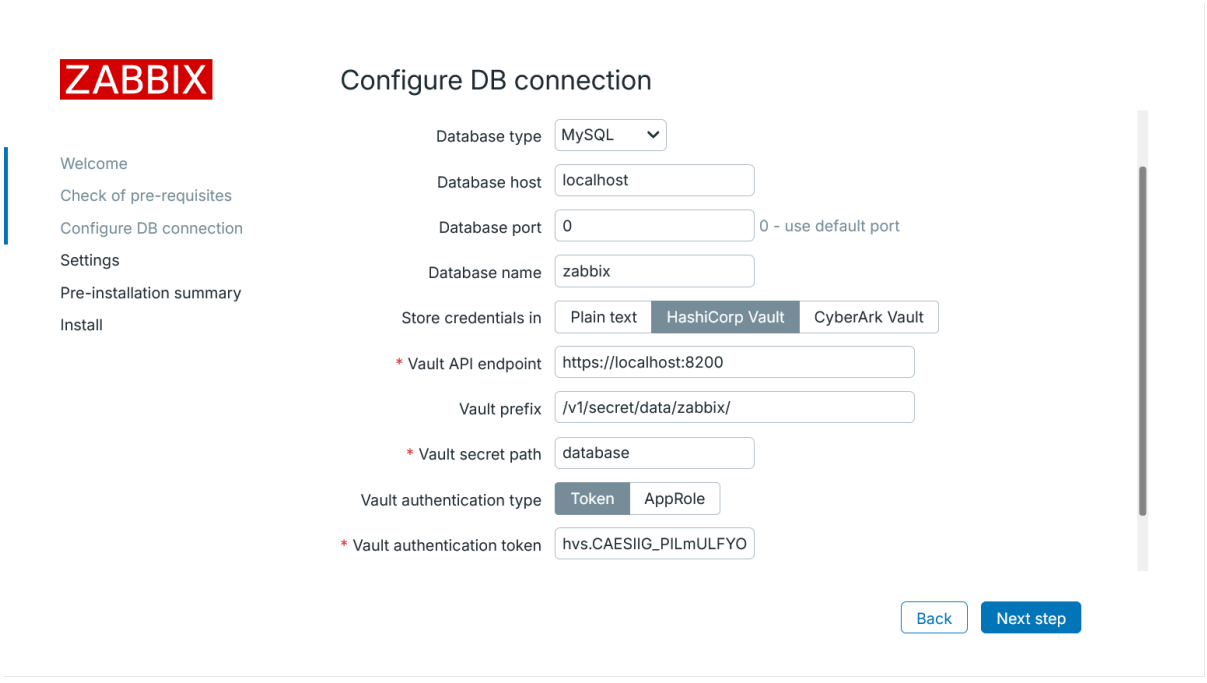
Frontend

El frontend de Zabbix se puede configurar para recuperar las credenciales de la base de datos desde el vault durante la instalación del frontend o actualizando el archivo de configuración del frontend (`zabbix.conf.php`).

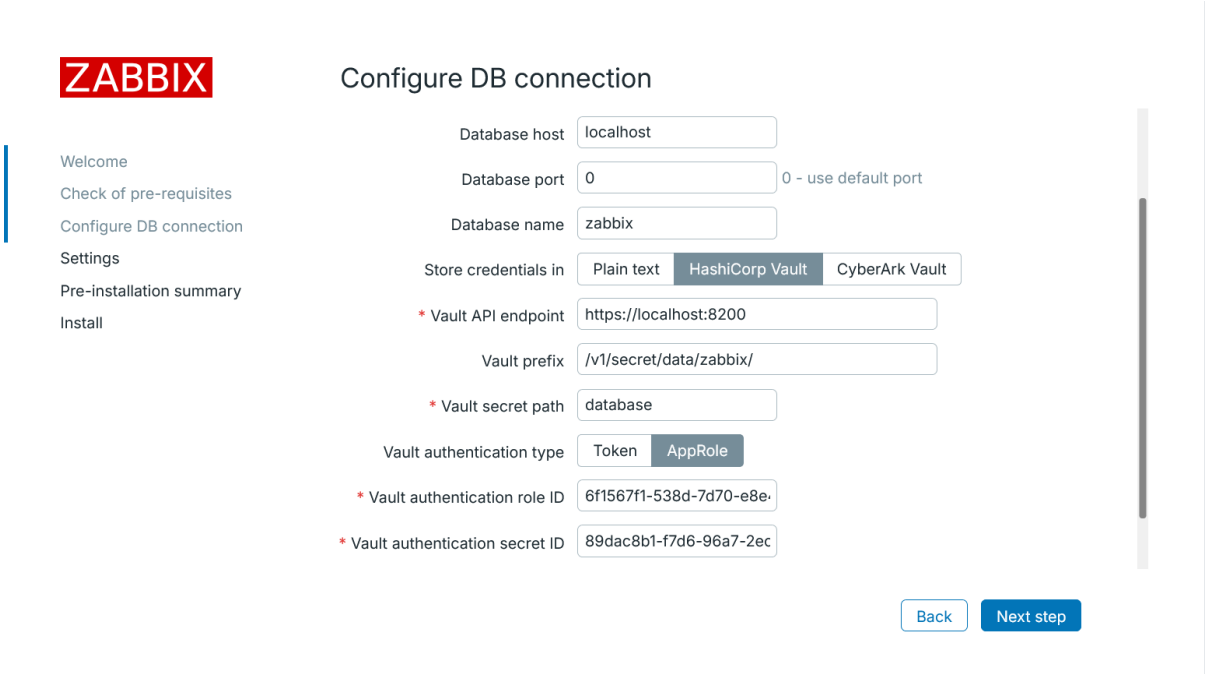
Attention:
Si las credenciales del vault han cambiado desde la instalación anterior del frontend, actualice la configuración existente.

Durante la **instalación del frontend**, los parámetros de conexión deben especificarse en el paso *Configure DB Connection*.

Ejemplo de configuración mediante autenticación con token:



Ejemplo de configuración mediante autenticación con AppRole:



Parameter	Mandatory	Default value	Description
Store credentials in	no		Seleccione **HashiCorp Vault**.
Vault API endpoint	yes	https://localhost:8200	Especifique la URL para conectarse al vault con el formato `scheme://host:port`
Vault prefix	no	/v1/secret/data/	Proporcione un prefijo personalizado para la ruta o consulta del vault. Si no se especifica, se utiliza el valor predeterminado. Ejemplo: `/v1/secret/data/zabbix/`
Vault secret path	yes		Ruta al secreto desde la que se deben recuperar las credenciales de la base de datos mediante las claves ”password” y ”username”. Ejemplo: `database`
Vault authentication type	no		Seleccione el tipo de autenticación del vault: **Token**: utiliza la autenticación con token; **AppRole**: utiliza el método de autenticación AppRole.
Vault authentication token	yes		Proporcione un token de autenticación con acceso de solo lectura a la ruta del secreto. Consulte la documentación de HashiCorp para obtener información sobre la creación de tokens y las políticas del vault. Este campo solo está disponible si se selecciona *Token* como tipo de autenticación del vault.
Vault authentication role ID	yes		Proporcione el ID de rol de AppRole. Consulte la documentación de HashiCorp para obtener información sobre AppRole. Este campo solo está disponible si se selecciona *AppRole* como tipo de autenticación del vault.
Vault authentication secret ID	yes		Proporcione el ID secreto emitido para AppRole. Consulte la documentación de HashiCorp para obtener información sobre AppRole. Este campo solo está disponible si se selecciona *AppRole* como tipo de autenticación del vault.

Recuperación de valores de macros de usuario

Para usar HashiCorp Vault para almacenar valores de macros de usuario de *Vault secret*, asegúrese de que:

- Zabbix server/proxy esté configurado para funcionar con HashiCorp Vault.
- El parámetro *Vault provider* en *Administration > General > Other* esté establecido en ”HashiCorp Vault” (valor predeterminado).



Note:
Zabbix server (y Zabbix proxy, si está configurado) requieren acceso a los valores de macro *Vault secret* desde el vault. Zabbix frontend no necesita dicho acceso.

El valor de la macro debe contener una ruta de referencia (como `path:key`, por ejemplo, `macros:password`). El token de autenticación especificado durante la configuración de Zabbix server/proxy (mediante el parámetro `VaultToken`) debe proporcionar acceso de solo lectura a esta ruta.

Consulte *Vault secret macros* para obtener información detallada sobre el procesamiento de valores de macro por parte de Zabbix.

Sintaxis de la ruta

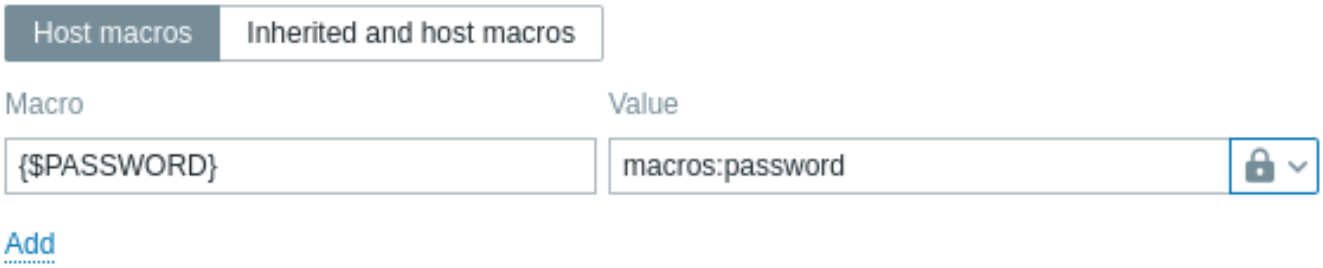
Los símbolos barra diagonal ("/") y dos puntos (":") están reservados.

Una barra diagonal solo se puede usar para separar un punto de montaje de una ruta (por ejemplo, *secret/zabbix*, donde el punto de montaje es "secret" y la ruta es "zabbix"). En el caso de los macros de Vault, un dos puntos solo se puede usar para separar una ruta/consulta de una clave.

Es posible codificar en URL los símbolos barra diagonal y dos puntos si es necesario crear un punto de montaje con un nombre que esté separado por una barra diagonal (por ejemplo, *foo/bar/zabbix*, donde el punto de montaje es "foo/bar" y la ruta es "zabbix", puede codificarse como "foo%2Fbar/zabbix") y si el nombre de un punto de montaje o una ruta necesita contener un dos puntos.

Ejemplo

1. En Zabbix, agregue una macro de usuario {$PASSWORD} de tipo "Vault secret" y con el valor `macros:password`



2. Ejecute los siguientes comandos CLI para crear el secreto requerido en el vault:

```
#### Habilite el punto de montaje "secret/" si aún no está habilitado; tenga en cuenta que se debe usar "k
vault secrets enable -path=secret/ kv-v2

#### Coloque un nuevo secreto con la clave "password" bajo el punto de montaje "secret/" y la ruta "zabbix
vault kv put -mount=secret zabbix/macros password=<password>

#### Compruebe que el secreto se haya agregado correctamente.
vault kv get secret/zabbix/macros

#### Finalmente, pruebe con Curl; tenga en cuenta que "data" debe agregarse manualmente después del punto
curl --header "X-Vault-Token: <VaultToken>" https://127.0.0.1:8200/v1/secret/data/zabbix/macros
```

3. Como resultado, Zabbix resolverá la macro {$PASSWORD} al valor: <password>

Actualización de la configuración existente

Para actualizar una configuración existente para recuperar secretos de un HashiCorp Vault:

1. Actualice los parámetros del archivo de configuración del Zabbix server o proxy, tal como se describe en la sección *Credenciales de la base de datos*.
2. Actualice los parámetros del archivo de configuración del frontend de Zabbix:

Uso del token de Vault:

```
$DB['VAULT']                    = 'HashiCorp';
$DB['VAULT_URL']                = 'https://localhost:8200';
$DB['VAULT_DB_PATH']            = 'database';
$DB['VAULT_TOKEN']              = 'hvs.CAESIIG_PILmULFYOsEyWHxkZ2mF2a8VPKNLE8eHqd4autYGGh4KHGh2cy5aeTY0NFN
$DB['VAULT_PREFIX']             = '/v1/secret/data/zabbix/';
```

O mediante el método AppRole de Vault:

```
$DB['VAULT']                    = 'HashiCorp';
$DB['VAULT_URL']                = 'https://localhost:8200';
$DB['VAULT_DB_PATH']            = 'database';
$DB['VAULT_APP_ROLE_ID']        = '6f1567f1-538d-7d70-e8e4-5d3e59471b2e';
$DB['VAULT_APP_SECRET_ID']      = '89dac8b1-f7d6-96a7-2ecf-f94d2de3d6f1';
$DB['VAULT_PREFIX']             = '/v1/secret/data/zabbix/';
```

3. Configure las macros de usuario tal como se describe en la sección *Valores de las macros de usuario*, si es necesario.

Para actualizar una configuración existente para recuperar secretos de un CyberArk Vault, consulte *Configuración de CyberArk*.

Solución de problemas

1. En el archivo de registro de Zabbix server/proxy:

```
237839:20260729:130916.425 Vault token renew started to fail: cannot login into HashiCorp vault with AppRo
```

Causa probable: HashiCorp Vault se ha (re)iniciado, pero no se ha desbloqueado (requisito en la configuración predeterminada, no necesario si Auto Unseal está configurado).

2. En el archivo de registro de Zabbix server/proxy:

```
235545:20260729:125617.944 Vault token renew started to fail: cannot login into HashiCorp vault with AppRo
```

o

```
235545:20260729:125718.537 Vault token renew failed: cannot login into HashiCorp vault with AppRole method
```

Causa probable: VaultAppSecretID ha caducado o ha alcanzado su límite de uso en HashiCorp vault.

Confianza en el certificado TLS de Google Chrome

Descripción general

Esta página proporciona pasos de configuración de Zabbix y ejemplos de configuración para usar Google Chrome con el frontend de Zabbix o con endpoints de zabbix-web-service protegidos por un certificado autofirmado o por una autoridad de certificación privada.

Estas instrucciones asumen que el servidor web de destino ya está configurado para HTTPS. Para configurar TLS en el frontend de Zabbix, consulte Secure connection to the frontend.

Configuración

Google Chrome en Linux usa una base de datos de certificados NSS por usuario para los certificados de confianza. Para que Chrome confíe en un certificado autofirmado o en una autoridad de certificación privada, añada los certificados necesarios a la base de datos NSS de la cuenta de usuario que ejecuta Chrome.

1. Instale los paquetes necesarios.

Para Debian/Ubuntu:

```
sudo apt install ca-certificates libnss3-tools
```

Para sistemas basados en RHEL:

```
sudo dnf install ca-certificates nss-tools
```

O:

```
sudo yum install ca-certificates nss-tools
```

2. Prepare los archivos de certificado.

Para cada certificado necesario: cree un archivo `.crt` basado en PEM para el certificado autofirmado, o cree archivos `.crt` separados basados en PEM para el certificado de la autoridad de certificación raíz y, si es necesario, para cada certificado intermedio.

3. Cree el directorio de la base de datos NSS.

Use el directorio personal del usuario que ejecuta Chrome.

Por ejemplo:

```
sudo mkdir -p /var/lib/zabbix/.pki/nssdb
sudo -u zabbix certutil -N -d sql:/var/lib/zabbix/.pki/nssdb
sudo chown -R zabbix:zabbix /var/lib/zabbix/.pki/nssdb
```

Si Chrome en el sistema de destino usa una ubicación distinta para la base de datos NSS, use ese directorio en su lugar.

Asegúrese de que los archivos de certificado sean legibles por el usuario que ejecuta Google Chrome. Si es necesario, cópielos a una ubicación accesible para ese usuario.

4. Importe los certificados.

Si usa un certificado autofirmado:

```
sudo -u zabbix certutil -d sql:/var/lib/zabbix/.pki/nssdb \
  -A -t "P,," \
  -n "Zabbix self-signed certificate" \
  -i /path/to/self-signed.crt
```

Si usa una autoridad de certificación privada, importe el certificado de la autoridad de certificación raíz en la base de datos:

```
sudo -u zabbix certutil -d sql:/var/lib/zabbix/.pki/nssdb \
  -A -t "C,," \
  -n "Zabbix root certificate authority" \
  -i /path/to/root-ca.crt
```

Si se usan certificados intermedios, importe cada uno por separado:

```
sudo -u zabbix certutil -d sql:/var/lib/zabbix/.pki/nssdb \
  -A -t ",," \
  -n "Zabbix intermediate certificate" \
  -i /path/to/intermediate-ca.crt
```

5. Verifique la base de datos de certificados.

```
sudo -u zabbix certutil -d sql:/var/lib/zabbix/.pki/nssdb -L
```

6. Pruebe Chrome.

Google Chrome usa la base de datos de certificados de la cuenta de usuario que lo inicia. Si la cuenta de usuario no tiene un directorio personal escribible, configure un directorio personal y directorios XDG escribibles antes de ejecutar la prueba.

Ejemplo (reemplace `google-chrome` por `chromium` si se usa Chromium.):

```
sudo -u zabbix env \
  HOME=/var/lib/zabbix-home \
  XDG_CONFIG_HOME=/var/lib/zabbix-home/.config \
  XDG_DATA_HOME=/var/lib/zabbix-home/.local/share \
  XDG_CACHE_HOME=/var/lib/zabbix-home/.cache \
  google-chrome --headless --disable-gpu \
  --user-data-dir=/var/lib/zabbix-home/chrome-profile \
  --dump-dom https://FQDN_OF_ZABBIX_SERVER
```

Si la configuración es correcta, el comando devuelve el HTML de la página solicitada.

Si Chrome no confía en el certificado, el comando devuelve una página de error de certificado que contiene mensajes como `NET::ERR_CERT_AUTHORITY_INVALID`.

Solución de problemas

Error	Posible causa
NET::ERR_CERT_AUTHORITY_INVALID	El certificado se importó en el perfil de usuario incorrecto, falta el certificado de la autoridad de certificación raíz o falta un certificado intermedio.
NET::ERR_CERT_COMMON_NAME_INVALID	El nombre del certificado no coincide con el nombre de host usado en la URL del navegador.
NET::ERR_CERT_DATE_INVALID	El certificado ha caducado, aún no es válido o el reloj del sistema es incorrecto.
Chrome sigue mostrando una advertencia de certificado después de la importación	Chrome está usando una base de datos NSS diferente para el usuario actual o las marcas de confianza del certificado son incorrectas.

Buenas prácticas de seguridad

Descripción general Esta sección contiene las prácticas recomendadas para configurar Zabbix de forma segura.

Las prácticas de esta sección no son necesarias para el funcionamiento de Zabbix, pero se recomiendan para mejorar la seguridad del sistema.

Para conocer las prácticas recomendadas para mantener segura la aplicación móvil de Zabbix y el adaptador de puente de Zabbix, consulte Acceso móvil seguro.

Codificación UTF-8 UTF-8 es la única codificación compatible con Zabbix. Se sabe que funciona sin ninguna vulnerabilidad de seguridad. Los usuarios deben tener en cuenta que existen problemas de seguridad conocidos si se utilizan algunas de las otras codificaciones.

Rutas del instalador de Windows Al usar instaladores de Windows, se recomienda utilizar las rutas predeterminadas proporcionadas por el instalador. Usar rutas personalizadas sin los permisos adecuados podría comprometer la seguridad de la instalación.

Macros en scripts globales definidos por el usuario Para mejorar la seguridad, se recomienda usar funciones de macro en lugar de macros simples en scripts globales definidos por el usuario, ya que las macros no se escapan automáticamente.

Avisos de seguridad de Zabbix y base de datos CVE Consulte *Avisos de seguridad de Zabbix y base de datos CVE*.

Plantillas de correo electrónico HTML Al crear o editar plantillas de mensajes utilizadas para correos electrónicos HTML, incluya siempre cada macro dentro de la función de macro `htmlencode()`. Por ejemplo:

```
<b>Problem started</b> at {{EVENT.TIME}.htmlencode()} on {{EVENT.DATE}.htmlencode()}<br><b>Problem name:</
{{EVENT.NAME}.htmlencode()}<br><b>Host:</b> {{HOST.NAME}.htmlencode()}
<br><b>Severity:</b>
{{EVENT.SEVERITY}.htmlencode()}<br><b>Operational
data:</b> {{EVENT.OPDATA}.htmlencode()}<br><b>Original problem ID:</b> {{EVENT.ID}.htmlencode()}<br>{{TRIG
```

El uso de `htmlencode()` garantiza que cualquier carácter HTML presente en los valores de las macros se escape y evita la inyección de HTML en las notificaciones (por ejemplo, que un atacante inserte un enlace malicioso o de phishing en una notificación).

Nota: los mensajes de correo electrónico HTML predeterminados proporcionados por Zabbix ya aplican `htmlencode()` a las macros. Esta indicación se aplica al editar plantillas existentes o crear nuevas; verifique que las macros estén codificadas antes de utilizar una plantilla para enviar un correo electrónico HTML.

Interfaz web

Descripción general Para acceder fácilmente a Zabbix desde cualquier lugar y plataforma, se proporciona una interfaz basada en la web.

Note:
Si utiliza más de una instancia de frontend, asegúrese de que las configuraciones regionales y las bibliotecas (LDAP, SAML, etc.) estén instaladas y configuradas de forma idéntica en todos los frontends.

Ayuda del frontend En las secciones y los formularios del frontend de Zabbix, en la esquina superior derecha, se proporciona un enlace de ayuda que dirige directamente a las partes correspondientes de la documentación.

Para los campos de formulario que tienen un icono de ayuda, al pasar el cursor sobre el icono se muestra la sugerencia correspondiente.

Modo quiosco En las secciones del frontend que tienen el botón *Modo quiosco* (ubicado en la esquina superior derecha), haga clic en el botón para cambiar la sección al modo quiosco. En el modo quiosco, solo se muestra el contenido de la sección.

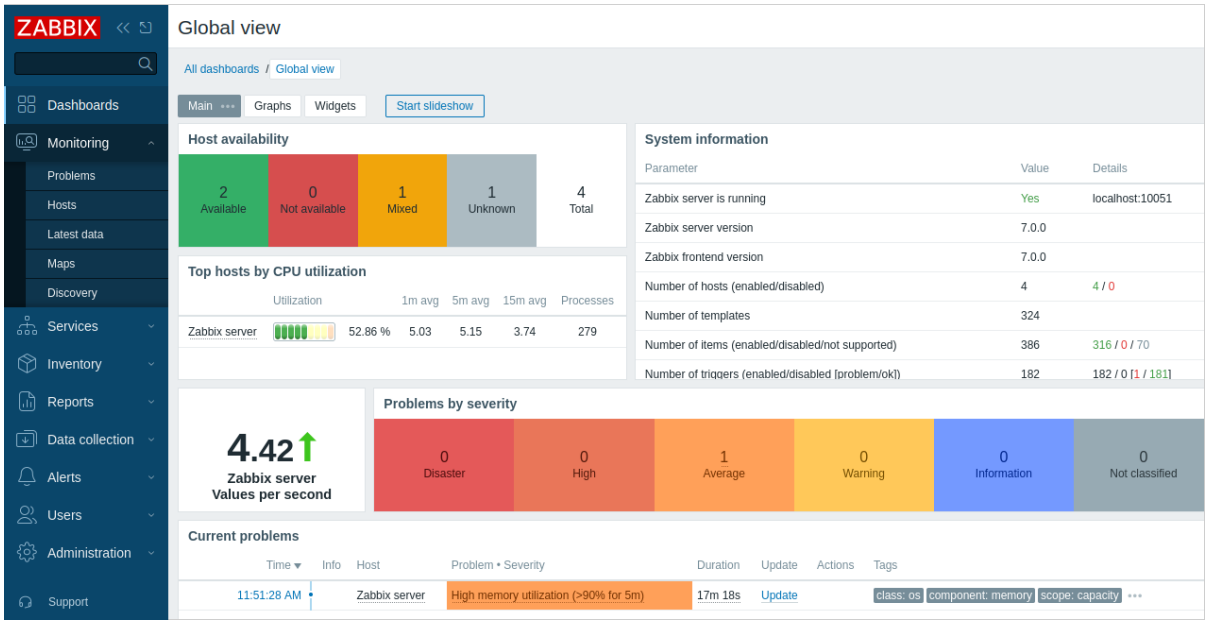
Para salir del modo quiosco, coloque el cursor sobre el contenido y haga clic en el botón *Vista normal* que aparece en la esquina superior derecha.

Menú

Descripción general

Un menú vertical en una barra lateral proporciona acceso a varias secciones de la interfaz web de Zabbix.

El menú es de color azul oscuro en el tema por defecto.

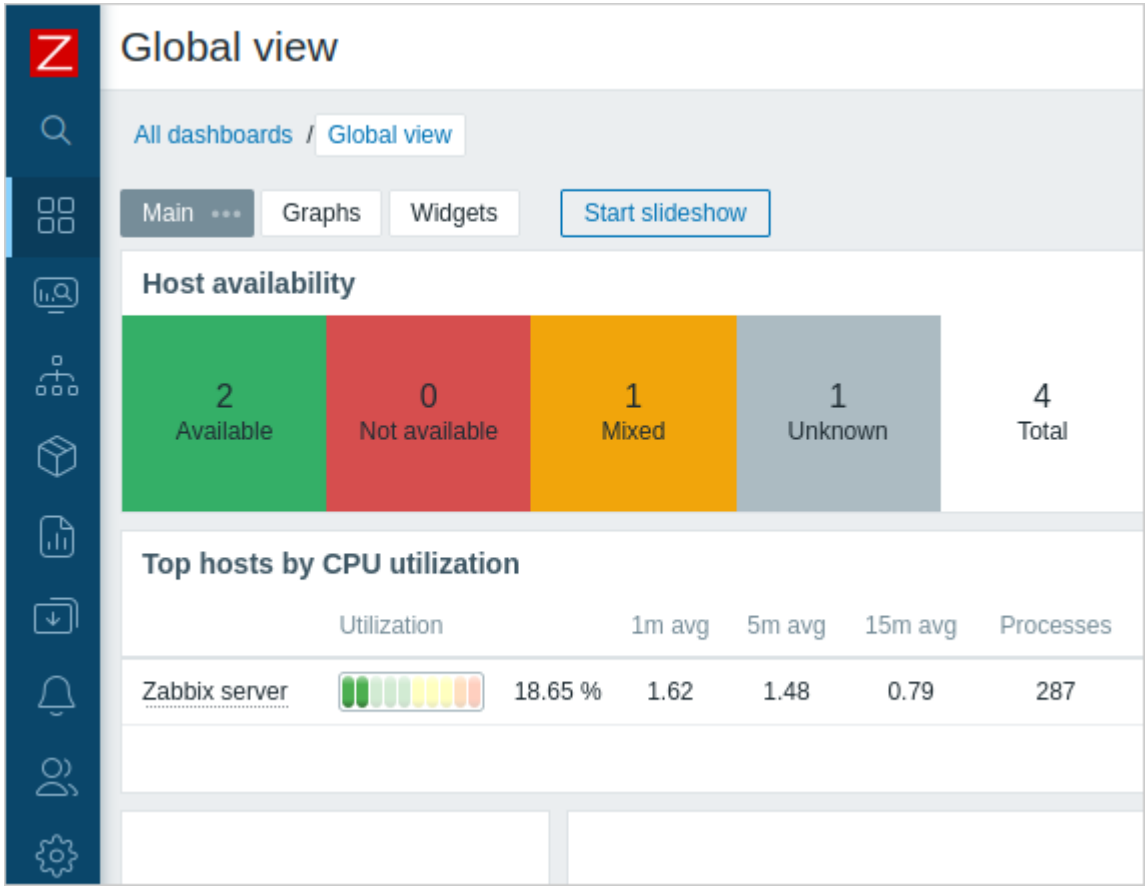


Trabajar con el menú

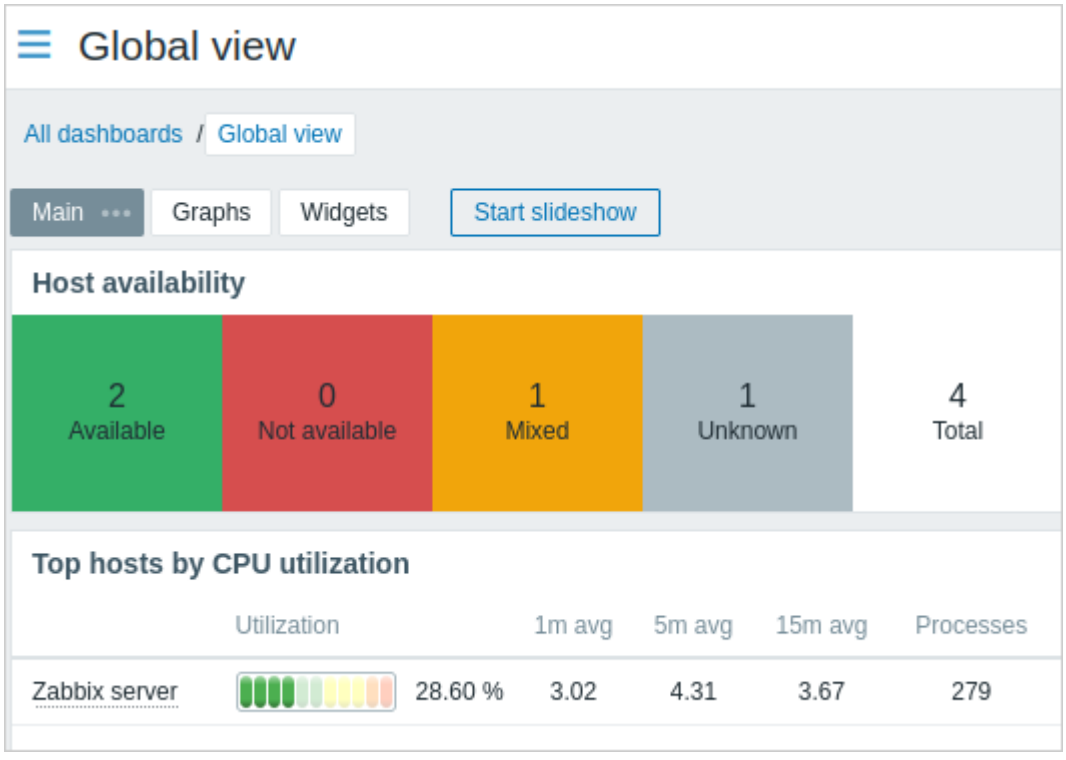
Un cuadro de búsqueda global se encuentra debajo del logotipo de Zabbix.

El menú se puede contraer o ocultar por completo:

- Para contraerlo, haga clic en junto al logotipo de Zabbix. En el menú contraído solo son visibles los iconos.



- Para ocultarlo, haga clic en  junto al logotipo de Zabbix. En el menú oculto todo queda oculto.



Menú contraído

Cuando el menú está contraído y solo muestra iconos, el menú completo vuelve a aparecer en cuanto se coloca el cursor del ratón sobre él. Tenga en cuenta que vuelve a aparecer sobre el contenido de la página; para desplazar el contenido de la página hacia la derecha, debe hacer clic en el botón de expansión. Si el cursor del ratón vuelve a colocarse fuera del menú completo, el menú volverá a contraerse después de dos segundos.

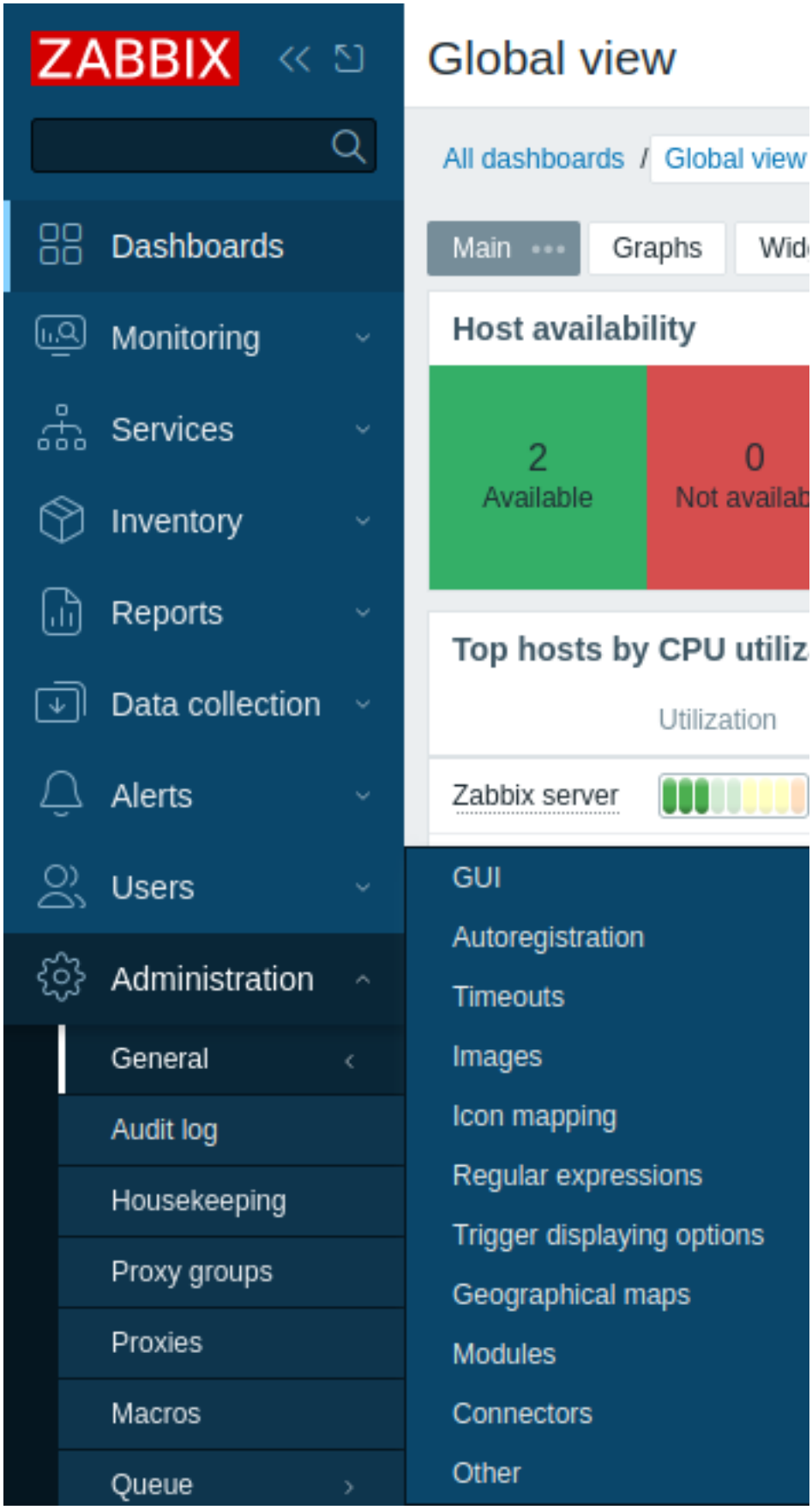
También puede hacer que un menú contraído vuelva a mostrarse por completo pulsando la tecla Tab. Al pulsar la tecla Tab repetidamente, puede situarse en el siguiente elemento del menú.

Menú oculto

Incluso cuando el menú está completamente oculto, un menú completo está a solo un clic del mouse, haciendo clic en el icono de hamburguesa . Tenga en cuenta que reaparece sobre el contenido de la página; para mover el contenido de la página hacia la derecha, debe mostrar el menú haciendo clic en el botón de mostrar barra lateral.

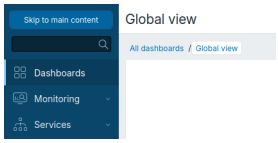
Niveles de menú

Hay hasta tres niveles en el menú.



Accesibilidad: Omitir navegación

El enlace *Saltar al contenido principal* permite a los usuarios de teclado y lectores de pantalla omitir la navegación repetitiva de la barra lateral y pasar directamente al contenido principal.



Permanece oculto durante el uso normal y solo aparece cuando recibe el foco, por ejemplo, cuando se pulsa la tecla Tab después de que se cargue la página o cuando se accede a él mediante un lector de pantalla.

Menús contextuales

Además del menú principal, Zabbix proporciona menús contextuales para host, item y event para acceder rápidamente a entidades de uso frecuente, como los últimos valores, el gráfico simple, el formulario de configuración, scripts relacionados o enlaces externos.

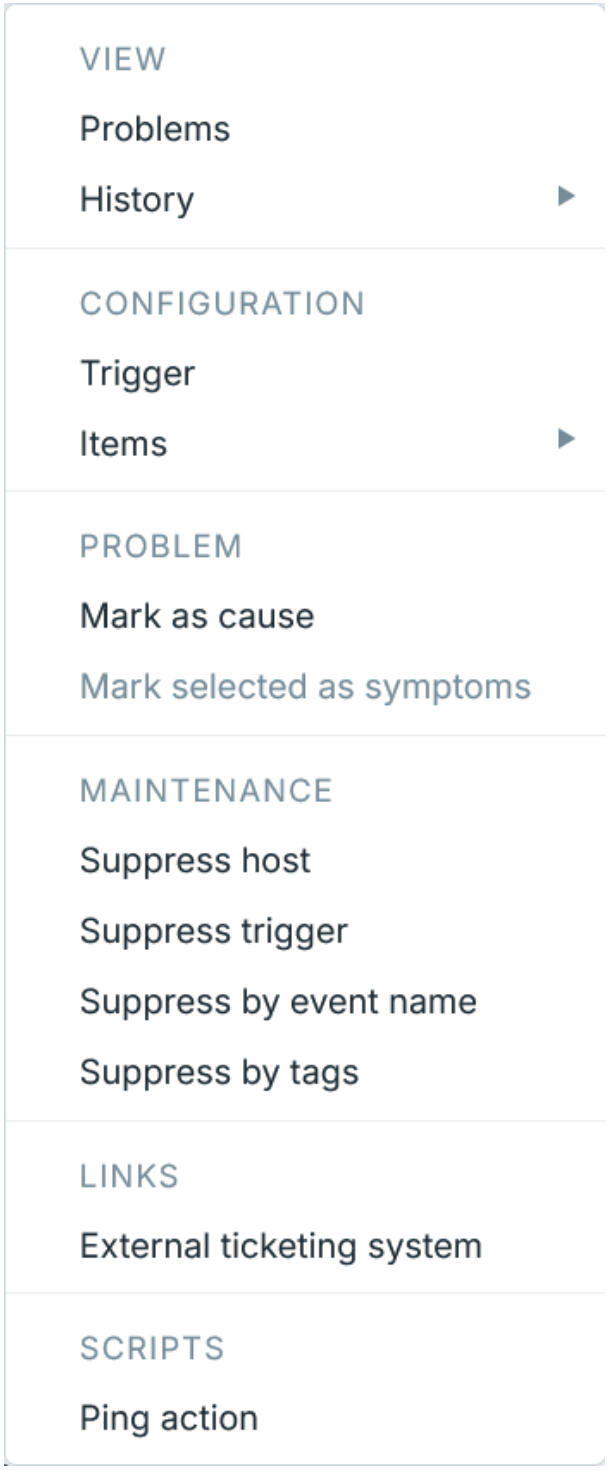
Se puede acceder a los menús contextuales haciendo clic en el nombre del host, item o problema/trigger en las ubicaciones compatibles.

Menú de eventos

Descripción general

El menú de eventos contiene accesos directos a acciones o secciones del frontend que se requieren con frecuencia para un evento.

El menú de eventos se puede abrir haciendo clic en el nombre del evento.



Contenido

El menú contextual del evento tiene varias secciones: *Ver*, *Acciones*, *Configuración*, *Problema*, *Mantenimiento*, *Enlaces* y *Scripts*. Para las entidades que no están configuradas, los enlaces están deshabilitados y se muestran en gris. Las secciones *Scripts* y *Enlaces* se muestran si sus entidades están configuradas.

La sección *Ver* contiene enlaces a:

- **Problemas**: abre la lista de problemas sin resolver del trigger subyacente;
- **Historial**: lleva al gráfico/historial del item en *Últimos datos* correspondiente al item subyacente. Si un trigger utiliza varios items, habrá enlaces disponibles para cada uno de ellos.

La sección *Acciones* solo está disponible en los widgets *Resumen de triggers*. Contiene un enlace a:

- **Actualizar problema**: abre la pantalla de actualización del problema.

La sección *Configuración* contiene enlaces a la configuración de:

- **Trigger** que generó el problema;
- **Items** utilizados en la expresión del trigger.

Note:
Tenga en cuenta que la sección de configuración solo está disponible para usuarios Administrador y Superadministrador.

La sección *Problema* contiene las siguientes opciones:

- **Marcar como causa**: marca el problema como causa;
- **Marcar los seleccionados como síntomas**: marca los problemas seleccionados como síntomas de este problema.

La sección *Mantenimiento* contiene opciones para iniciar un mantenimiento ad hoc:

- **Suprimir host**: inicia un período de mantenimiento para el host;
- **Suprimir trigger**: inicia un período de mantenimiento para el trigger;
- **Suprimir por nombre del evento**: inicia un período de mantenimiento para el nombre del evento;
- **Suprimir por etiquetas**: inicia un período de mantenimiento para los eventos con las etiquetas correspondientes.

Al seleccionar la opción correspondiente, se abrirá el formulario de configuración del período de mantenimiento.

Se preseleccionará un período de mantenimiento de una hora, de una sola vez (configurable en los ajustes y el perfil del usuario), que comenzará a partir de este momento. También se preseleccionarán el host, el trigger, el nombre del evento o las etiquetas correspondientes. El nombre del mantenimiento se completará previamente con el formato `Ad-hoc: <event name>`.

La sección *Enlaces* contiene enlaces para:

- acceder a una URL del trigger configurada;
- acceder a enlaces personalizados configurados en Scripts globales (con el alcance «Acción manual del evento» y el tipo «URL»);
- acceder a un ticket externo configurado para el problema (consulte la opción *Incluir entrada del menú del evento* al configurar webhooks).

La sección *Scripts* contiene enlaces para ejecutar un script global (con el alcance *Acción manual del evento*). Esta función puede resultar útil para ejecutar scripts utilizados para gestionar tickets de problemas en sistemas externos.

Ubicaciones compatibles

El menú contextual del evento está accesible al hacer clic en un nombre de problema o evento en varias secciones del frontend, por ejemplo:

- Dashboards widgets, como el widget *Problems*, el widget *Trigger overview*, etc.
- *Monitoring* > *Problems*
- *Monitoring* > *Problems* > Detalles del evento
- *Reports* > *Top 100 triggers* (los scripts globales y el acceso a tickets externos no son compatibles en esta ubicación)

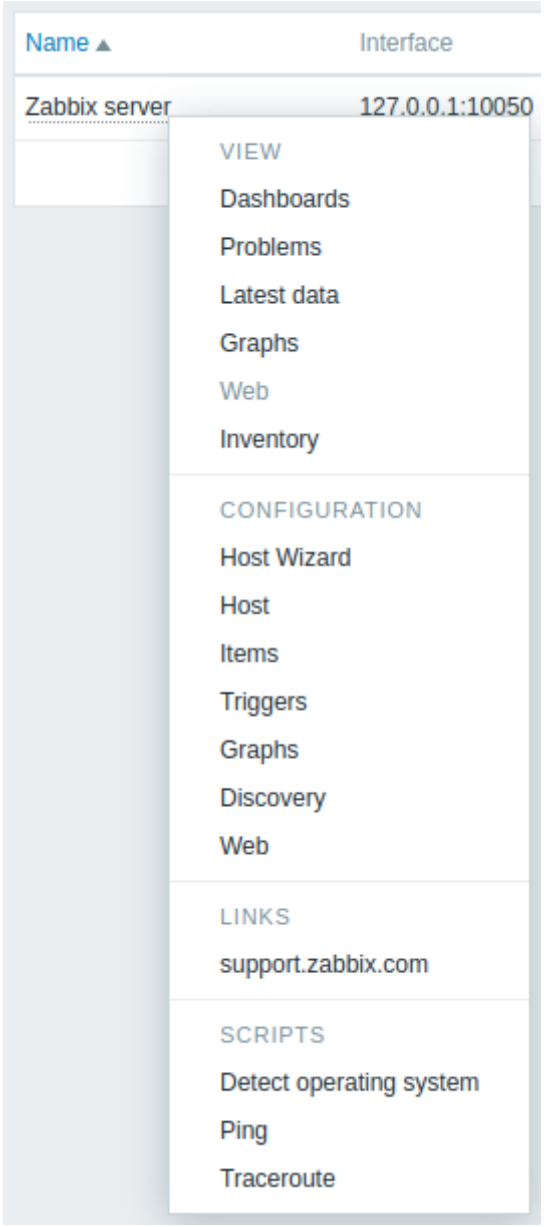
Menú del host

Descripción general

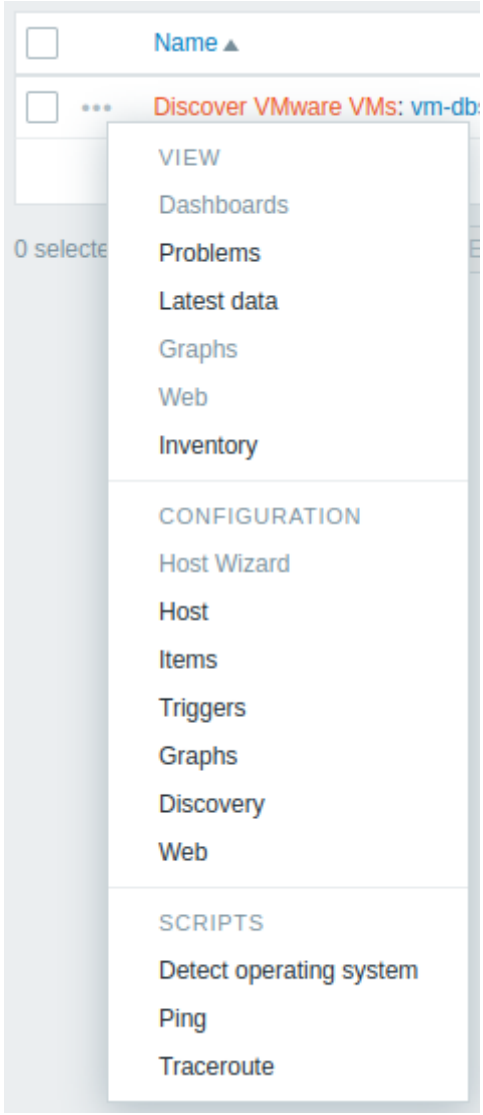
El menú del host contiene accesos directos a acciones o secciones del frontend que se requieren con frecuencia para un host.

El menú del host se puede abrir haciendo clic en el nombre del host o en el icono de tres puntos, según la sección del frontend, por ejemplo:

Monitoring > *Latest data*



Data collection > *Hosts*



Contenido

El menú contextual del host tiene cuatro secciones: *Vista*, *Configuración*, *Enlaces* y *Scripts*. Para las entidades que no están configuradas, los enlaces están deshabilitados y se muestran en gris. Las secciones *Scripts* y *Enlaces* se muestran si sus entidades están configuradas.

La sección *Vista* contiene enlaces a:

- **Dashboards**: abre los widgets y gráficos.
- **Problemas**: abre la sección *Problemas* con la lista de problemas sin resolver del trigger subyacente.
- **Últimos datos**: abre la sección *Últimos datos* con la lista de todos los datos más recientes del host actual.
- **Gráficos**: abre los gráficos simples del host actual.
- **Web**: abre el enlace a los escenarios web configurados.
- **Inventario**: abre el enlace al inventario del host actual.

La sección *Configuración* contiene enlaces a:

- **Asistente de hosts**: abre el Asistente de hosts para el host actual (deshabilitado para los hosts descubiertos).
- **Host**: formulario de configuración del host actual.
- **Items**: lista de los items del host actual.
- **Triggers**: lista de los triggers del host actual.
- **Gráficos**: gráficos simples del host actual.
- **Descubrimiento**: lista de las reglas de descubrimiento de bajo nivel del host actual.

- **Web**: lista de los escenarios web del host actual.

> **Note:**
> Tenga en cuenta que la sección de configuración solo está disponible para usuarios administradores y superadministradores.

La sección *Enlaces* contiene enlaces para:

- acceder a una URL del trigger configurada.
- acceder a enlaces personalizados configurados en Scripts globales (con el alcance *Acción manual del host* y el tipo «URL»).

La sección *Scripts* permite ejecutar scripts globales configurados para el host actual. Estos scripts deben tener definido el alcance *Acción manual del host* para estar disponibles en el menú del host.

Ubicaciones admitidas

El menú del host es accesible al hacer clic en el nombre de un host en varias secciones del frontend, por ejemplo:

- Paneles widgets, como *Problems*, *Top items*, *Trigger overview*, etc.
- *Monitoring* > *Problems*
- *Monitoring* > *Problems* > Detalles del evento
- *Monitoring* > *Hosts*
- *Monitoring* > *Hosts* > *Web Monitoring*
- *Monitoring* > *Latest data*
- *Monitoring* > *Maps*
- *Inventory* > *Hosts*
- *Reports* > *Top 100 triggers*

El menú del host es accesible al hacer clic en el icono de tres puntos en Data collection > Hosts.

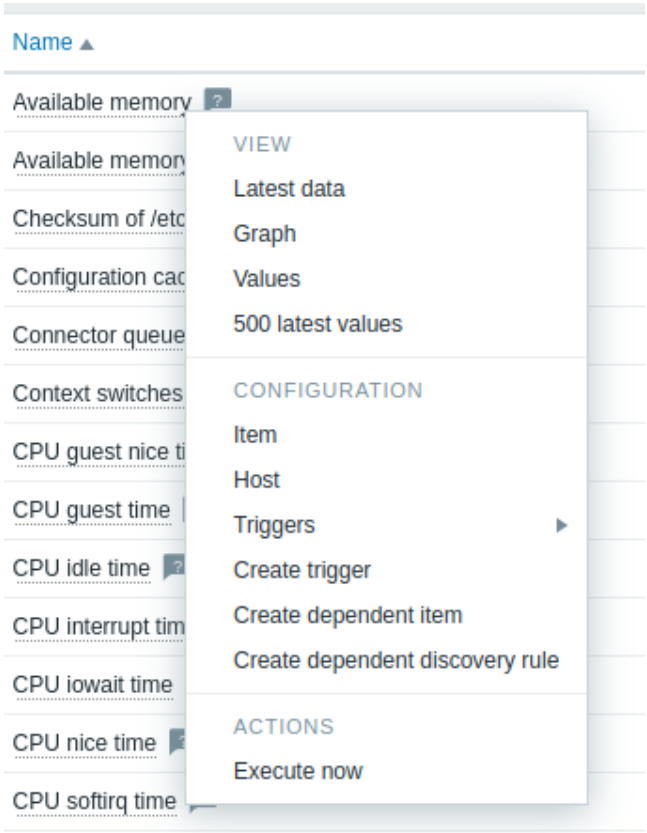
Menú de item

Descripción general

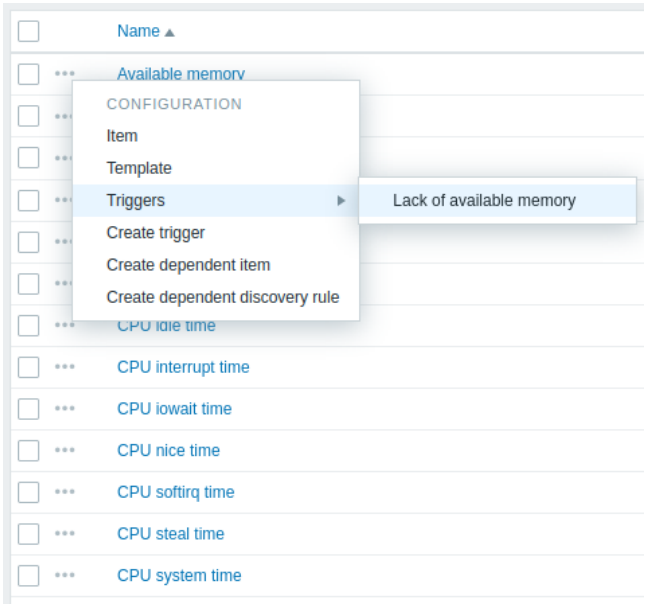
El menú del item contiene accesos directos a acciones o secciones del frontend que se requieren con frecuencia para un item.

El menú del item se puede abrir haciendo clic en el nombre del item o en el icono de tres puntos, según la sección del frontend, por ejemplo:

Monitoring > *Latest data*



Data Collection > *Templates* > *Items*



Contenido

El menú del item tiene tres secciones: *Vista*, *Configuración* y *Acciones*.

La sección *Vista* contiene las siguientes opciones:

- **Últimos datos**: abre la sección *Últimos datos* filtrada por el host y el item actuales;
- **Gráfico**: abre un gráfico simple del item actual;
- **Valores**: abre la lista de todos los valores recibidos para el item actual durante los últimos 60 minutos;
- **500 valores más recientes**: abre la lista de los 500 valores más recientes del item actual.

La sección *Configuración* (disponible únicamente para usuarios de tipo *Admin* y *Super admin*) contiene las siguientes opciones:

- **Item**: abre el formulario de configuración del item actual;
- **Template**: abre el formulario de configuración del template al que pertenece el item (disponible únicamente cuando se accede al menú del item desde *Recopilación de datos* > *Templates* > *Items*);
- **Host**: abre el formulario de configuración del host al que pertenece el item;
- **Triggers**: al pasar el cursor por encima, abre una lista de los triggers del item, si los hay; al hacer clic en un trigger, se abre su formulario de configuración del trigger;
- **Crear trigger**: abre el formulario de configuración del trigger para crear un trigger para este item;
- **Crear item dependiente**: abre el formulario de configuración del item para crear un item dependiente, con el item actual como item maestro;
- **Crear regla de descubrimiento dependiente**: abre el formulario de configuración de la regla de descubrimiento para crear una regla de descubrimiento dependiente, con el item actual como item maestro.

La sección *Acciones* contiene la siguiente opción:

- **Ejecutar ahora**: ejecuta una comprobación para obtener inmediatamente un nuevo valor del item.

Ubicaciones compatibles

El menú de item es accesible al hacer clic en el nombre de un item en varias secciones del frontend, por ejemplo:

- *Monitoring* > *Latest data*
- *Data collection* > *Hosts* > *Items*
- *Data collection* > *Hosts* > *Discovery rules* > *Item prototypes*

El menú de item es accesible al hacer clic en el valor de un item en el widget del panel *Top items*.

Secciones del frontend

Estructura del menú El menú del frontend de Zabbix tiene la siguiente estructura:

- Paneles
- Monitorización
  - Problemas
  - Hosts
  - Datos más recientes
  - Mapas
  - Detección
- APM
  - Trazas
  - Métricas
  - Registros
- Servicios
  - Servicios
  - SLA
  - Informe de SLA
- Inventario
  - Resumen
  - Hosts
- Informes
  - Información del sistema
  - Informes programados
  - Informe de disponibilidad
  - 100 triggers principales
  - Registro de auditoría
  - Registro de acciones
  - Notificaciones
- Recopilación de datos
  - Grupos de templates
  - Grupos de hosts
  - Templates
  - Hosts
  - Mantenimiento
  - Correlación de eventos
  - Detección
- Alertas
  - Acciones
    - Acciones de trigger
    - Acciones de servicio
    - Acciones de detección
    - Acciones de autoregistro
    - Acciones internas
  - Tipos de medios
  - Scripts
- Usuarios
  - Grupos de usuarios
  - Roles de usuario
  - Usuarios
  - Tokens de API
  - Autenticación
  - Dispositivos
- Administración
  - General
    - GUI
    - Autoregistro
    - Tiempos de espera
    - Imágenes
    - Asignación de iconos

    - Expresiones regulares
    - Opciones de visualización de triggers
    - Mapas geográficos
    - Módulos
    - Conectores
    - Otros
- Registro de auditoría
- Mantenimiento interno
- Grupos de proxies
- Proxies
- Macros
- Cola
    - Resumen de la cola
    - Resumen de la cola por proxy
    - Detalles de la cola

Paneles de control

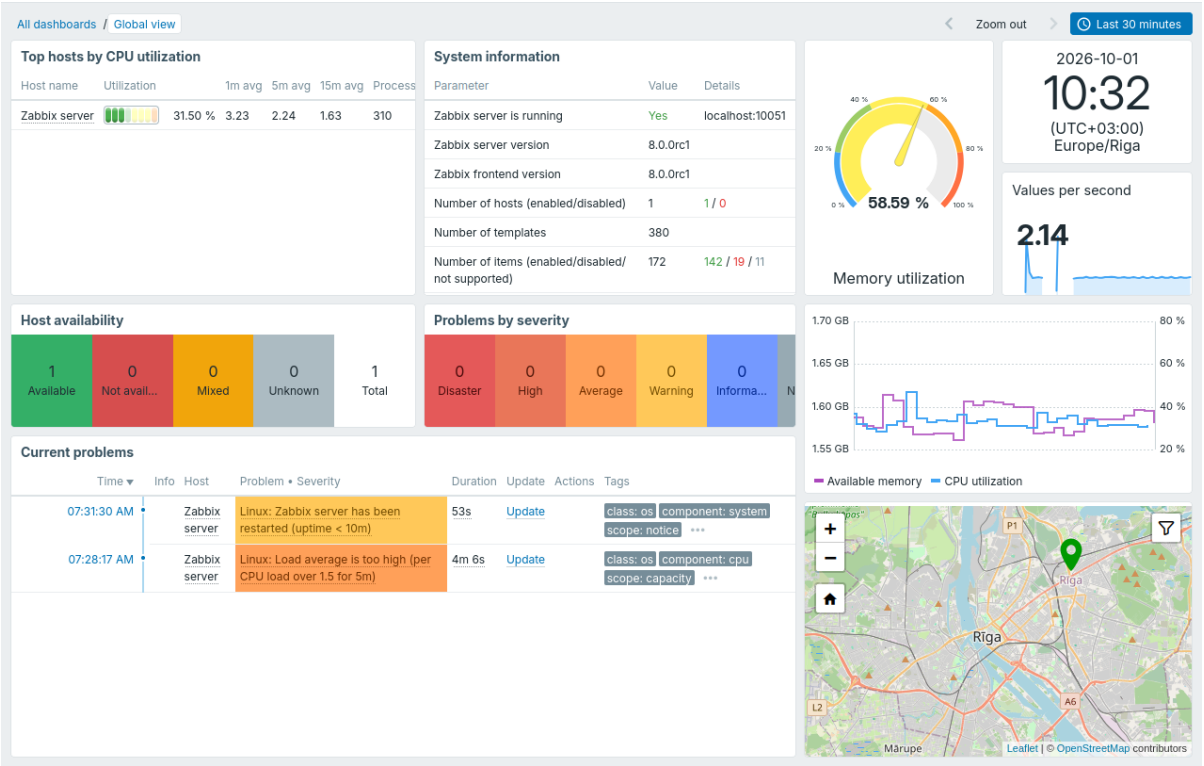
Los dashboards le ofrecen una visión general de su entorno monitorizado. Puede crearlos y personalizarlos para adaptarlos a sus necesidades.

Los elementos básicos son los widgets: un gráfico, un mapa, una lista de problemas recientes, etc., cada uno diseñado para mostrar un tipo de información específico.

Los widgets se pueden organizar en varias páginas del dashboard. Estas páginas se pueden visualizar como una presentación de diapositivas. También puede navegar a cualquier página haciendo clic en su pestaña.

Puede crear varios dashboards, aunque solo se muestra uno a la vez.

Zabbix incluye algunos dashboards predeterminados. Uno de ellos es el dashboard *Global view*, que muestra el estado y la salud de sus hosts monitorizados, incluido el server de Zabbix:

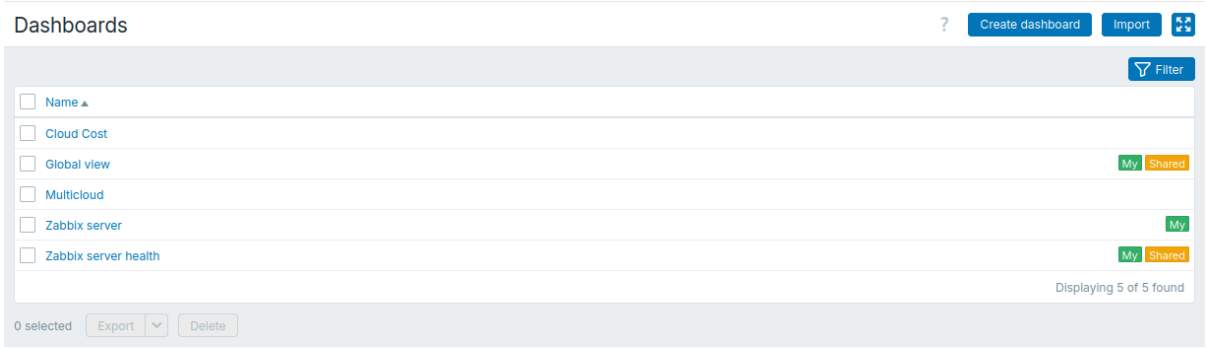


Puede encontrar más dashboards en el repositorio de Dashboards. Estos dashboards están diseñados para funcionar con los templates existentes de Zabbix y se pueden descargar e importar en su entorno de Zabbix. En el archivo `README` del dashboard encontrará una descripción detallada de cada dashboard, incluidos sus requisitos previos.

Lista de dashboards

Para abrir la lista de dashboards, puede:

- Hacer clic en *Dashboards* en el menú.
- Hacer clic en *Todos los dashboards* mientras visualiza un dashboard para abrir la lista de dashboards.



El acceso al menú *Dashboards* depende del rol de usuario y del permiso *Dashboard* en *Acceso a elementos de la interfaz de usuario*.

Desde aquí, puede: - Crear dashboards. - Eliminar dashboards. Puede eliminar un dashboard seleccionando su casilla de verificación y haciendo clic en el botón *Eliminar*. - Importar dashboards. - Exportar dashboards.

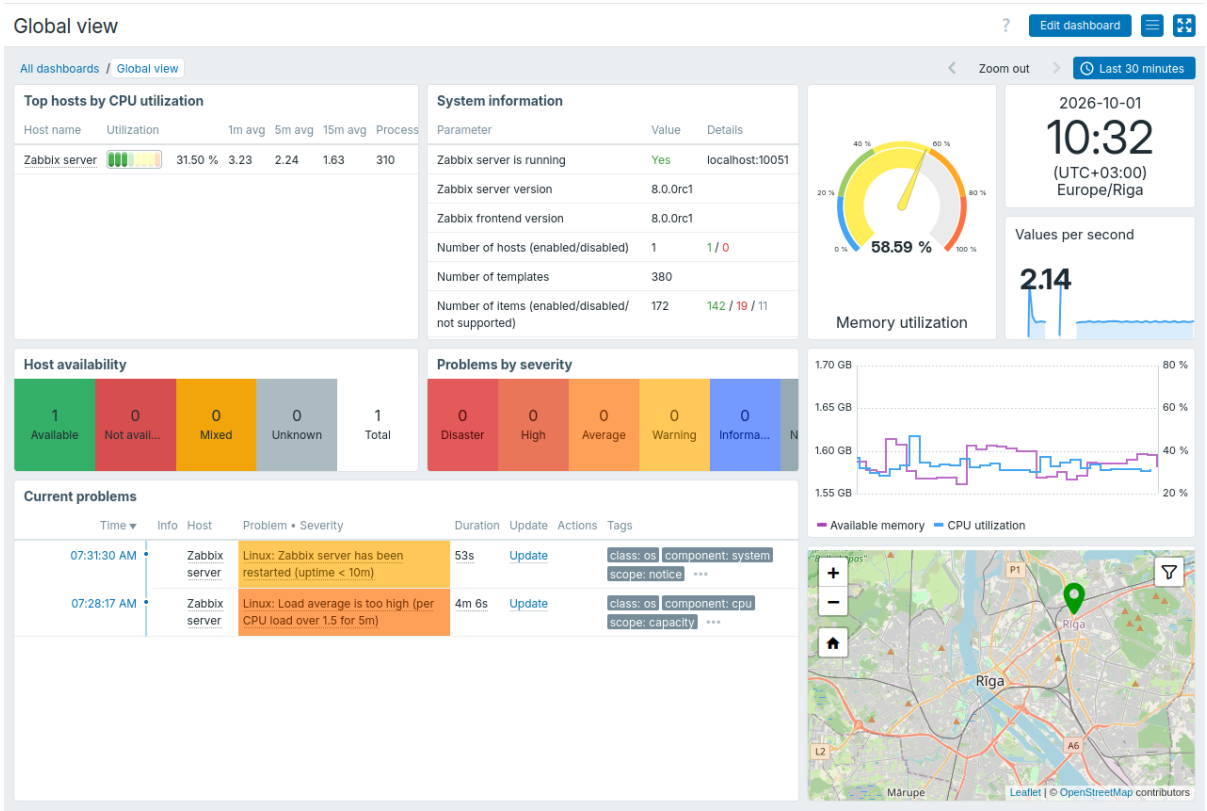
Los dashboards tienen etiquetas que indican su propiedad y configuración de uso compartido:

- *Mío*: el dashboard es suyo y solo usted puede verlo.
- *Mío + Compartido*: el dashboard es suyo, pero es público o se comparte con usuarios o grupos de usuarios específicos.
- Los dashboards sin etiquetas pertenecen a otros usuarios, pero son públicos o se comparten con usted.

Puede filtrar la lista por nombre y propietario del dashboard.

Ver el dashboard

Haga clic en el nombre de un dashboard en la lista de dashboards para ver el dashboard:



La interfaz de usuario de Zabbix recuerda el último dashboard abierto para la siguiente sesión.

Note:
Los datos que puede ver en un dashboard dependen de su acceso a los hosts.

Para los widgets que muestran gráficos, el selector de período de tiempo situado encima de los widgets establece el intervalo de tiempo. Muestra el intervalo de tiempo actual como una etiqueta. Haga clic en la etiqueta para abrir o cerrar el selector.

Desde aquí, puede administrar los dashboards de la siguiente manera:

- Editar el dashboard.
- Realizar las siguientes acciones desde el menú *Actions* del dashboard :
  - *Sharing*: editar la configuración de compartición del dashboard.

  - *Create new*: crear un nuevo dashboard.
  - *Clone*: crear un nuevo dashboard clonando el actual. Puede editar las propiedades del dashboard antes de guardarlo.
  - *Delete*: eliminar el dashboard.
  - *Export*: exportar el dashboard.
  - *Create new report*: crear un informe programado. Esta opción está deshabilitada si no tiene permiso para administrar informes programados.
  - *View related reports*: ver una lista de los informes existentes para el dashboard actual. Esta opción está deshabilitada si no hay informes relacionados o si no tiene permiso para administrar informes programados.
- Cambiar al modo kiosco.
- Iniciar o detener la presentación de diapositivas del dashboard.

En el modo de visualización del dashboard, están disponibles las siguientes opciones para los widgets:

- Editar el widget.
- Realizar las siguientes acciones desde el menú *Actions* del widget ⋮:
  - *Copy*: copiar el widget.
  - *Download image*: descargar el widget como una imagen PNG. Esta opción está disponible para los widgets que admiten esta función (consulte las páginas individuales de los widgets).
  - *Refresh interval*: seleccionar la frecuencia de actualización del contenido del widget.

Para los dashboards con varias páginas, las siguientes acciones están disponibles en el menú *Actions* de la página  en el modo de visualización del dashboard:

- *Copy*: copiar la página.
- *Properties*: personalizar las propiedades de la página, incluido el nombre y el intervalo de la presentación de diapositivas.

Presentación de diapositivas

Si la casilla *Iniciar presentación de diapositivas* está seleccionada en las propiedades del dashboard, la presentación de diapositivas se inicia automáticamente cuando el dashboard tiene más de una página. De lo contrario, puede iniciar la presentación de diapositivas haciendo clic en el botón *Iniciar presentación de diapositivas*.



Las páginas cambian según los intervalos especificados en las propiedades del dashboard y de cada página. Si la propiedad *Intervalo de presentación de diapositivas* está definida para una página individual, se aplica el intervalo especificado para esa página. De lo contrario, se aplica el *Intervalo de presentación de diapositivas predeterminado* del dashboard.

Para detener la presentación de diapositivas, haga clic en *Detener presentación de diapositivas*.

En el modo kiosco, utilice los siguientes botones:

-  para iniciar la presentación de diapositivas;
-  para detener la presentación de diapositivas;
-  para navegar a la página anterior;
-  para navegar a la página siguiente.

Compartir el dashboard

Los dashboards pueden ser públicos o privados.

Los dashboards públicos son visibles para todos los usuarios.
Los dashboards privados solo son visibles para sus propietarios y para los usuarios o grupos de usuarios con los que se comparten.

Para cambiar la configuración de uso compartido de un dashboard, haga clic en *Compartir* en el menú *Acciones* del dashboard mientras visualiza el dashboard:

Parámetro	Descripción
Tipo	Seleccione el tipo de dashboard: **Privado**: el dashboard solo es visible para los grupos de usuarios y usuarios seleccionados. **Público**: el dashboard es visible para todos los usuarios.
Lista de grupos de usuarios compartidos	Seleccione los grupos de usuarios que pueden acceder al dashboard. Puede conceder acceso de solo lectura o de lectura y escritura.

Parámetro	Descripción
Lista de usuarios compartidos	Seleccione los usuarios que pueden acceder al dashboard. Puede conceder acceso de solo lectura o de lectura y escritura.

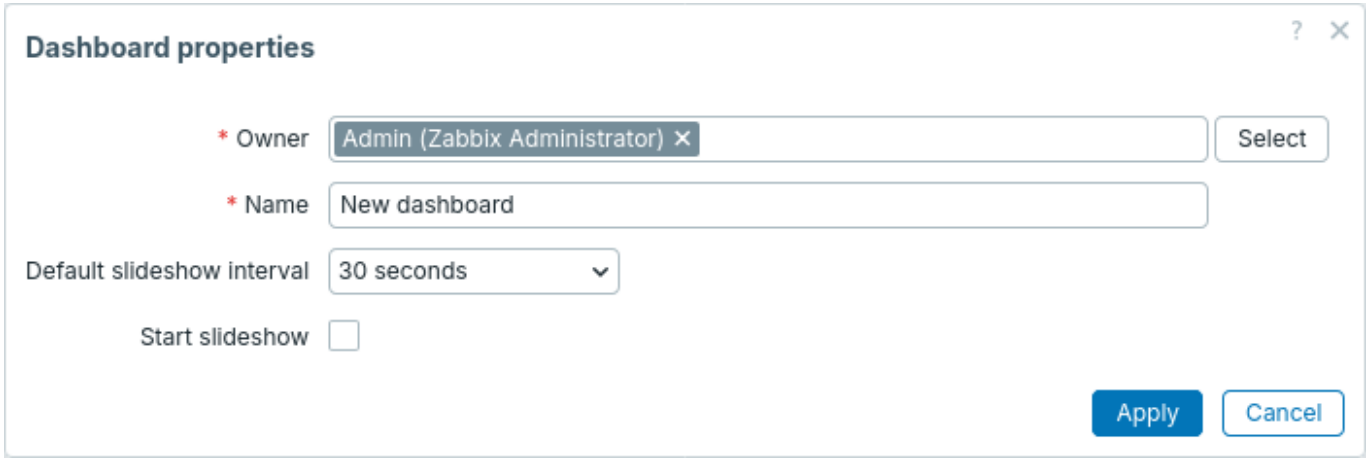
Consulte Permisos para dashboards para saber cómo las opciones de uso compartido limitan las acciones disponibles para los usuarios.

Crear un dashboard

Puede crear un dashboard de dos formas:

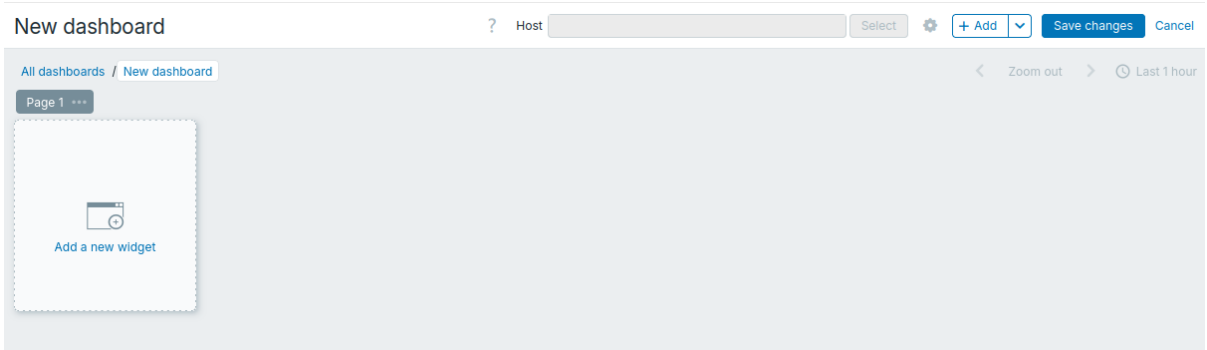
- En la lista de dashboards, haga clic en el botón *Crear dashboard*.
- En el modo de visualización del dashboard, haga clic en *Crear nuevo* en el menú *Acciones* del dashboard.

A continuación, especifique las propiedades del dashboard:



Parámetro	Descripción
Propietario	Seleccione el usuario que será el propietario del dashboard.
Nombre	Introduzca el nombre del dashboard.
Intervalo predeterminado de la presentación de diapositivas	Seleccione durante cuánto tiempo se muestra cada página del dashboard antes de que la presentación de diapositivas pase a la página siguiente.
Iniciar presentación de diapositivas	Marque esta casilla para iniciar automáticamente la presentación de diapositivas cuando el dashboard tenga más de una página.

A continuación, haga clic en *Aplicar*. El nuevo dashboard se abre en modo de edición:



Desde aquí, puede añadir widgets y páginas al dashboard.

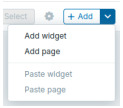
Tenga en cuenta que el ancho mínimo de un dashboard es de 1200 píxeles; en ventanas más estrechas, aparece una barra de desplazamiento horizontal. Un dashboard más ancho siempre ocupa toda la ventana del navegador; sus widgets no pueden extenderse más allá de los bordes de la ventana.

Haga clic en el botón *Guardar cambios* para guardar el dashboard o en *Cancelar* para volver a la pantalla anterior.

Añadir widgets

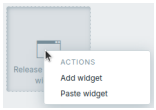
En el modo de edición del dashboard, puede añadir widgets de dos formas:

- Haga clic en el botón *Añadir*. Como alternativa, haga clic en la flecha desplegable del botón *Añadir* y seleccione *Añadir widget*.



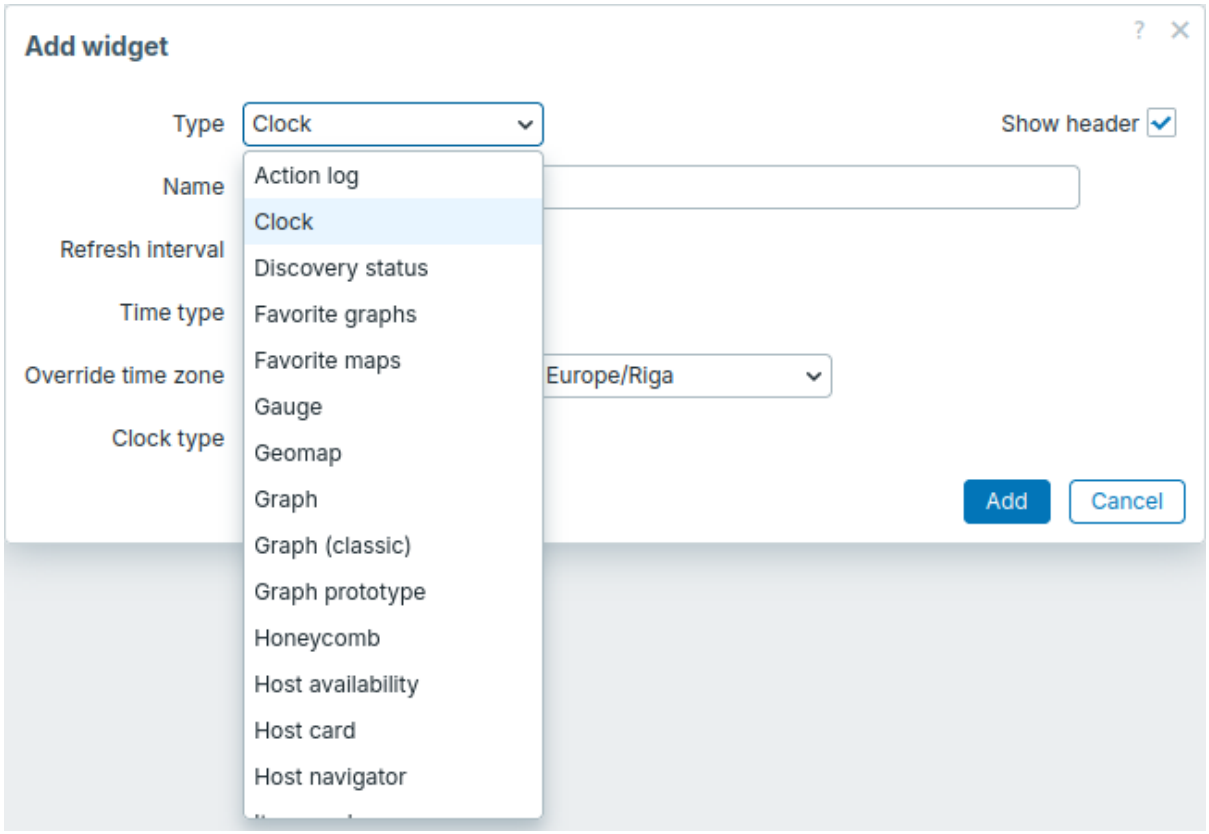
El marcador de posición del widget se añade con su tamaño predeterminado en el primer espacio vacío disponible del dashboard y se abre el formulario de configuración del widget.

- Mueva el puntero del ratón sobre un área vacía del dashboard. A continuación, haga clic en el marcador de posición del widget para abrir el formulario de configuración del widget. Como alternativa, haga clic en el marcador de posición y arrástrelo para cambiar su tamaño; después, suelte el botón del ratón para abrir el formulario de configuración del widget. Si se ha copiado un widget al portapapeles, primero se le pedirá que elija entre *Añadir widget* y *Pegar widget*.



A continuación, siga estos pasos:

1. En el formulario de configuración del widget, seleccione el tipo de widget.
2. Configure los parámetros del widget.
3. Haga clic en *Añadir* para añadir el widget al dashboard.



El widget se añade con su tamaño predeterminado o se cambia de tamaño para ajustarse al espacio disponible. Puede cambiar el tamaño y la posición de los widgets tal como se describe en la sección *Editar widgets*.

También puede copiar y pegar widgets para crear nuevos widgets con la misma configuración, ya sea dentro del mismo dashboard o entre dashboards abiertos en distintas pestañas del navegador.

Copiar/pegar widgets

Para copiar y pegar un widget:

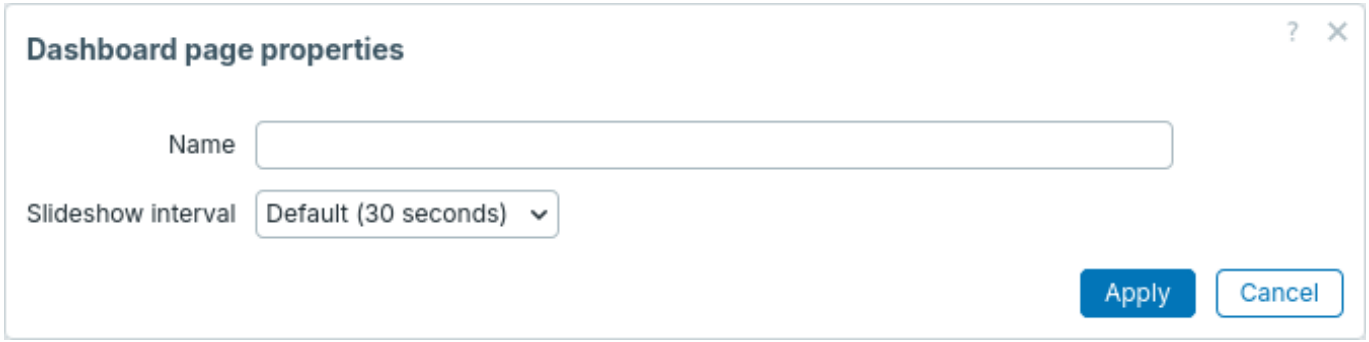
1. En el menú *Acciones* del widget ⋯, seleccione *Copiar*.
2. En el modo de edición del dashboard, haga clic en la flecha desplegable del botón *Añadir* y seleccione *Pegar widget*. Como alternativa, haga clic en un área vacía del dashboard y seleccione *Pegar widget*.

También puede pegar un widget copiado sobre uno existente mediante la opción *Pegar* de su menú *Acciones*.

Añadir páginas

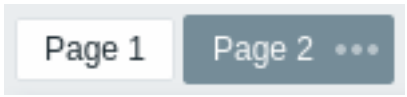
Para añadir una nueva página a un dashboard:

1. En el modo de edición del dashboard, haga clic en la flecha desplegable del botón *Añadir* y seleccione *Añadir página*.
2. Complete las propiedades de la página:
    - *Nombre*: introduzca el nombre de la página. Si deja el nombre vacío, la página se añadirá con el nombre *Página N*, donde «N» es el número incremental de la página.
    - *Intervalo de presentación*: seleccione durante cuánto tiempo se muestra la página antes de que la presentación pase a la siguiente página.



3. Haga clic en *Aplicar*.

Se añadirá una nueva pestaña (por ejemplo, *Página 2*) para la nueva página.



También puede copiar y pegar páginas para crear nuevas páginas con las mismas propiedades, ya sea dentro del mismo dashboard o entre dashboards abiertos en pestañas diferentes.

Copiar/pegar páginas

Para copiar y pegar una página:

1. En el menú *Acciones* de la página , seleccione *Copiar*.
2. En el modo de edición del dashboard, haga clic en la flecha desplegable del botón *Añadir* y seleccione *Pegar página*.

Editar el dashboard

Para editar un dashboard, haga clic en el botón *Editar dashboard* situado en la esquina superior derecha. Como alternativa, haga clic en el botón *Editar* de un widget. El dashboard se abre en modo de edición.

Desde aquí, puede administrar los dashboards de la siguiente manera:

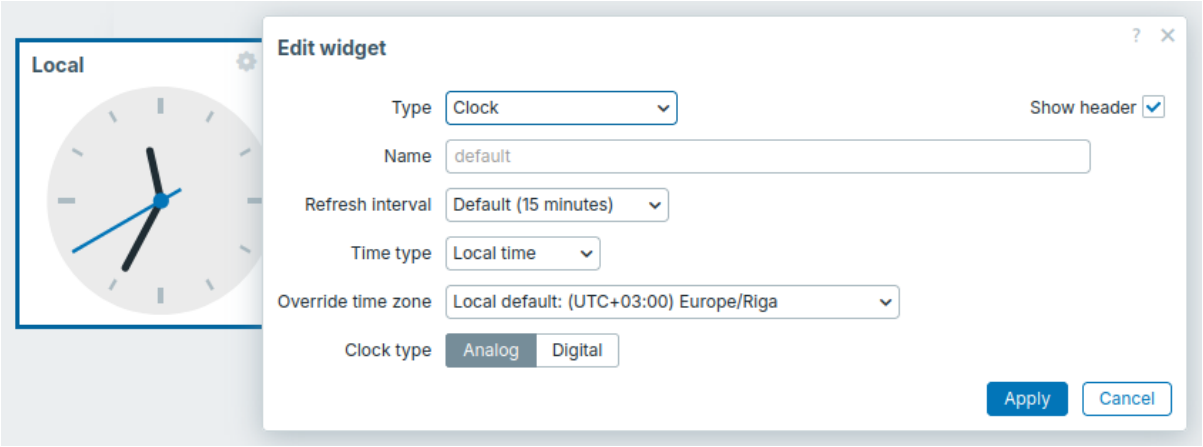
- Edite las propiedades del dashboard haciendo clic en el botón de engranaje.
- Edite los widgets del dashboard.
- Acceda a las siguientes acciones del dashboard haciendo clic en la flecha desplegable del botón *Añadir*:
    - *Añadir widget*: añada un widget al dashboard.
    - *Añadir página*: añada una página al dashboard.
    - *Pegar widget*: pegue un widget copiado. Esta opción aparece atenuada si no se ha copiado ningún widget. Solo se puede copiar un widget a la vez.
    - *Pegar página*: pegue una página copiada. Esta opción aparece atenuada si no se ha copiado ninguna página. Solo se puede copiar una página a la vez.

Después de editar el dashboard, haga clic en el botón *Guardar cambios* para guardar el dashboard o en *Cancelar* para volver a la pantalla anterior.

Editar widgets

Un widget se puede editar en el modo de edición del dashboard haciendo clic en su botón *Editar*. Como alternativa, al ver un dashboard, puede pasar el cursor sobre un widget y hacer clic en el botón *Editar*.

Mientras se edita un widget, este aparece resaltado y los cambios se previsualizan en tiempo real.



Puede cambiar la posición del widget y de su formulario de configuración arrastrando sus barras de título, y cambiar el tamaño del widget arrastrando sus controladores de cambio de tamaño.



Las siguientes acciones están disponibles en el menú *Acciones* del widget ⋯ al editar un widget:

- *Copiar*: copiar el widget.
- *Pegar*: pegar un widget copiado sobre este widget. Esta opción aparece atenuada si no se ha copiado ningún widget.
- *Eliminar*: eliminar el widget.

Si no se configuran los parámetros obligatorios del widget, este pasa al estado *El widget no está completamente configurado*.

Después de editar, haga clic en *Aplicar* en el formulario de configuración del widget y, a continuación, en *Guardar cambios* en la esquina superior derecha del dashboard para aplicar los cambios.

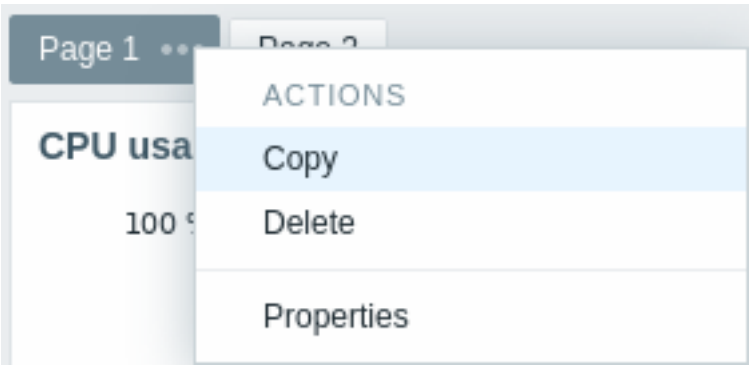
Editar páginas

Las páginas se pueden editar en el modo de edición del dashboard.

Las páginas se pueden reordenar arrastrando y soltando las pestañas de las páginas. Reordenar las páginas no cambia sus nombres, excepto en el caso de las páginas con nombres asignados automáticamente (*Page N*), que se vuelven a numerar.

Las siguientes acciones están disponibles en el menú *Actions* de la página ••• al editar una página:

- *Copy*: copiar la página.
- *Delete*: eliminar la página.
- *Properties*: personalizar las propiedades de la página.



Permisos para los dashboards

Los permisos para los dashboards de los usuarios normales y de los usuarios de tipo *Admin* están limitados como se describe a continuación, siempre que el acceso a los dashboards y la opción *Crear y editar dashboards* estén habilitados para su rol de usuario asignado:

- Pueden ver y clonar un dashboard público incluso cuando no se hayan añadido usuarios ni grupos de usuarios a las preferencias de uso compartido del dashboard.
- Pueden ver y clonar un dashboard privado si tienen al menos permisos de *Lectura*, según se especifica en las preferencias de uso compartido.

- Pueden editar y eliminar un dashboard solo si tienen permisos de *Lectura y escritura*, según se especifica en las preferencias de uso compartido.
- No pueden cambiar el propietario del dashboard.

Widgets del dashboard

Descripción general

Zabbix proporciona una variedad de widgets de panel que visualizan y resumen los datos de monitoreo, brindándole una visión clara del rendimiento y la disponibilidad del sistema.

A continuación se muestra una descripción general de todos los widgets, agrupados según su propósito principal.

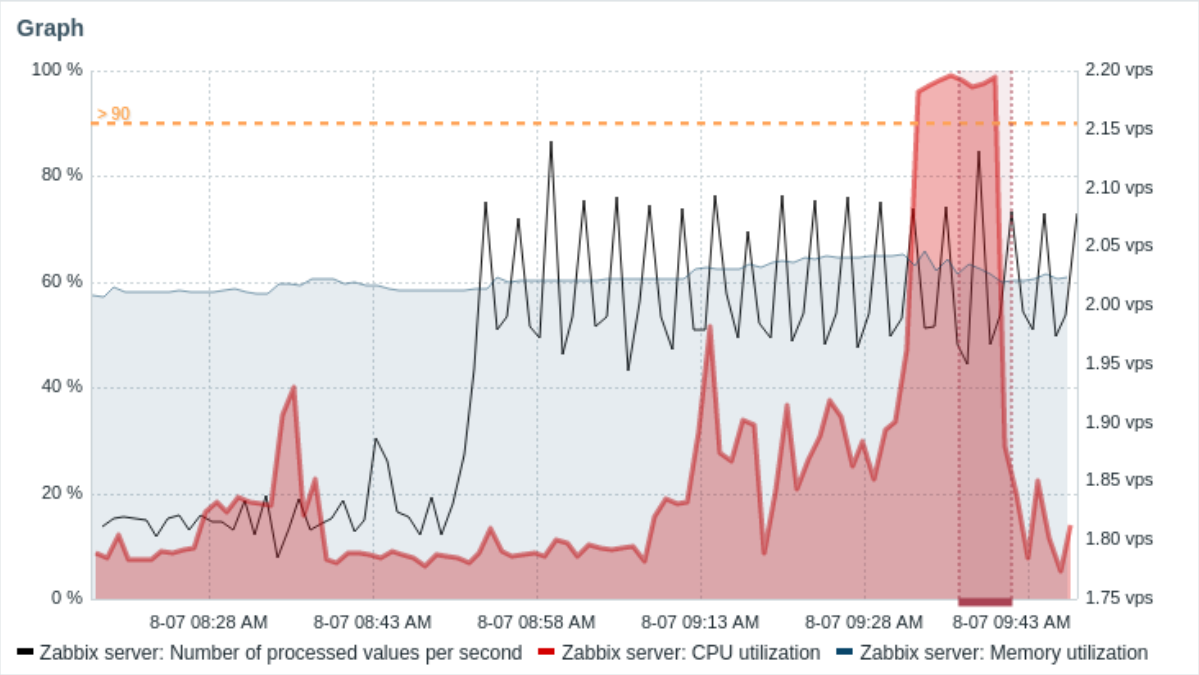
Para obtener detalles sobre la configuración de los widgets, consulte Parámetros de los widgets del panel.

Visualización de datos

Estos widgets presentan datos en varios formatos visuales, ayudándole a supervisar datos y problemas de diferentes maneras.

Gráfico

El widget Gráfico muestra los datos numéricos de un item como un gráfico vectorial. Puede ayudarle a realizar un seguimiento de las métricas, detectar problemas y comparar valores a lo largo del tiempo y entre hosts.



Consulte también:

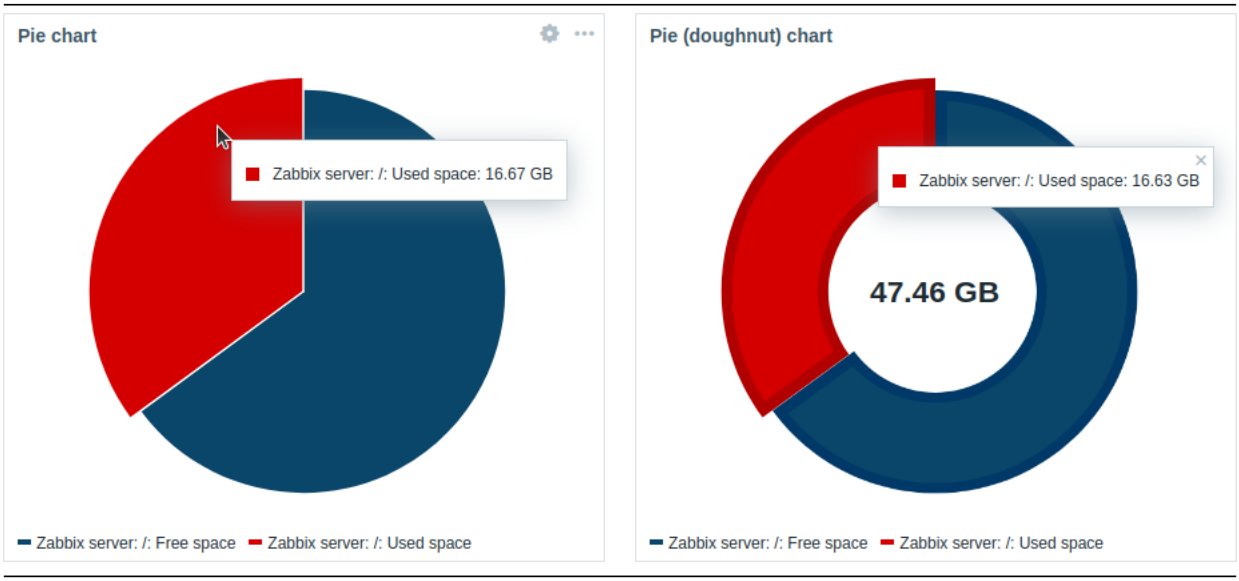
- Gráfico (clásico): el widget Gráfico heredado que muestra los datos numéricos de un item como un gráfico basado en imágenes.
- Prototipo de gráfico: muestra una cuadrícula de gráficos personalizados creados automáticamente basados en prototipos de gráficos de reglas de bajo nivel de descubrimiento o prototipos de item.
- Gráficos favoritos: muestra una lista alfabética de accesos directos a los gráficos marcados como favoritos por el usuario actual.

Gráfico de dispersión

El widget Gráfico de dispersión muestra la relación entre dos métricas trazando puntos de datos individuales a lo largo de un eje X e Y. Esto ayuda a revelar patrones, agrupaciones, correlaciones y valores atípicos en el conjunto de datos.

Gráfico circular

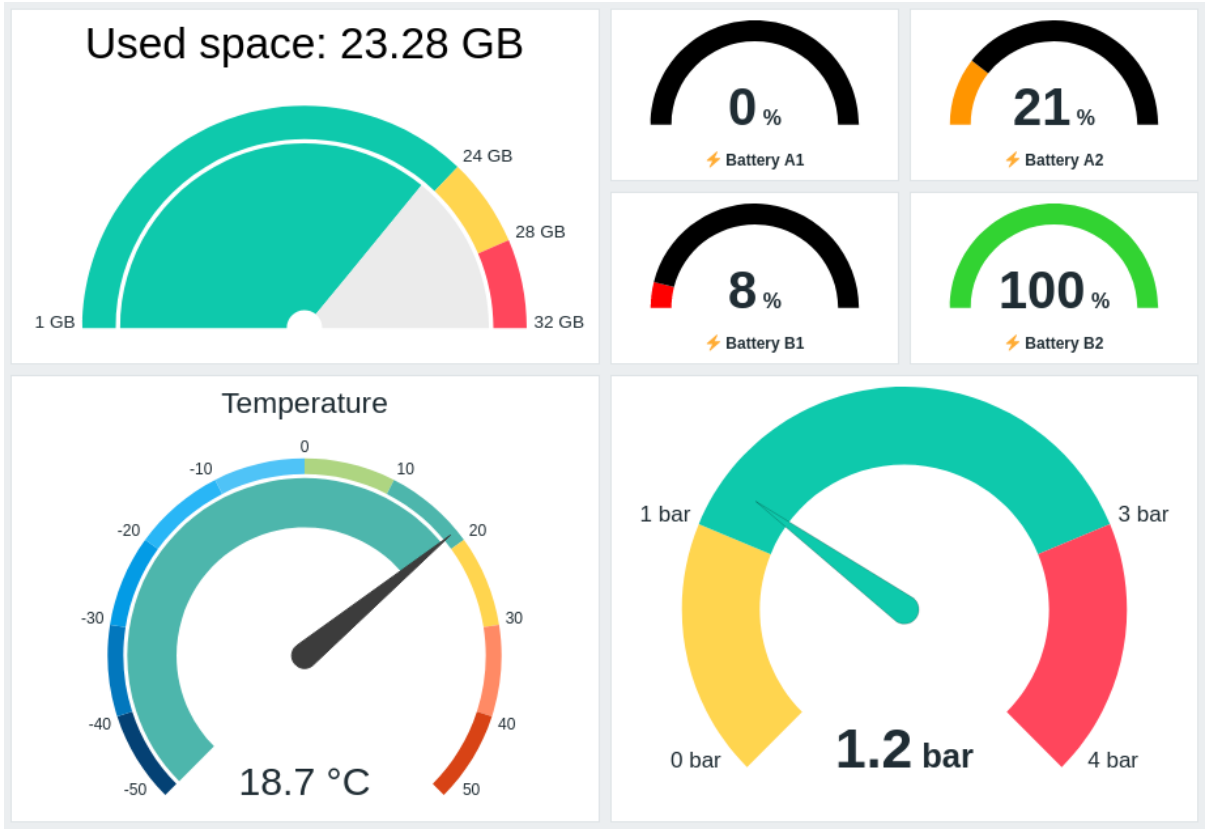
El widget Gráfico circular muestra datos numéricos de elementos como un gráfico circular o de anillo basado en vectores. Este widget puede ayudarle a visualizar cómo los elementos o los hosts contribuyen al conjunto de datos general.



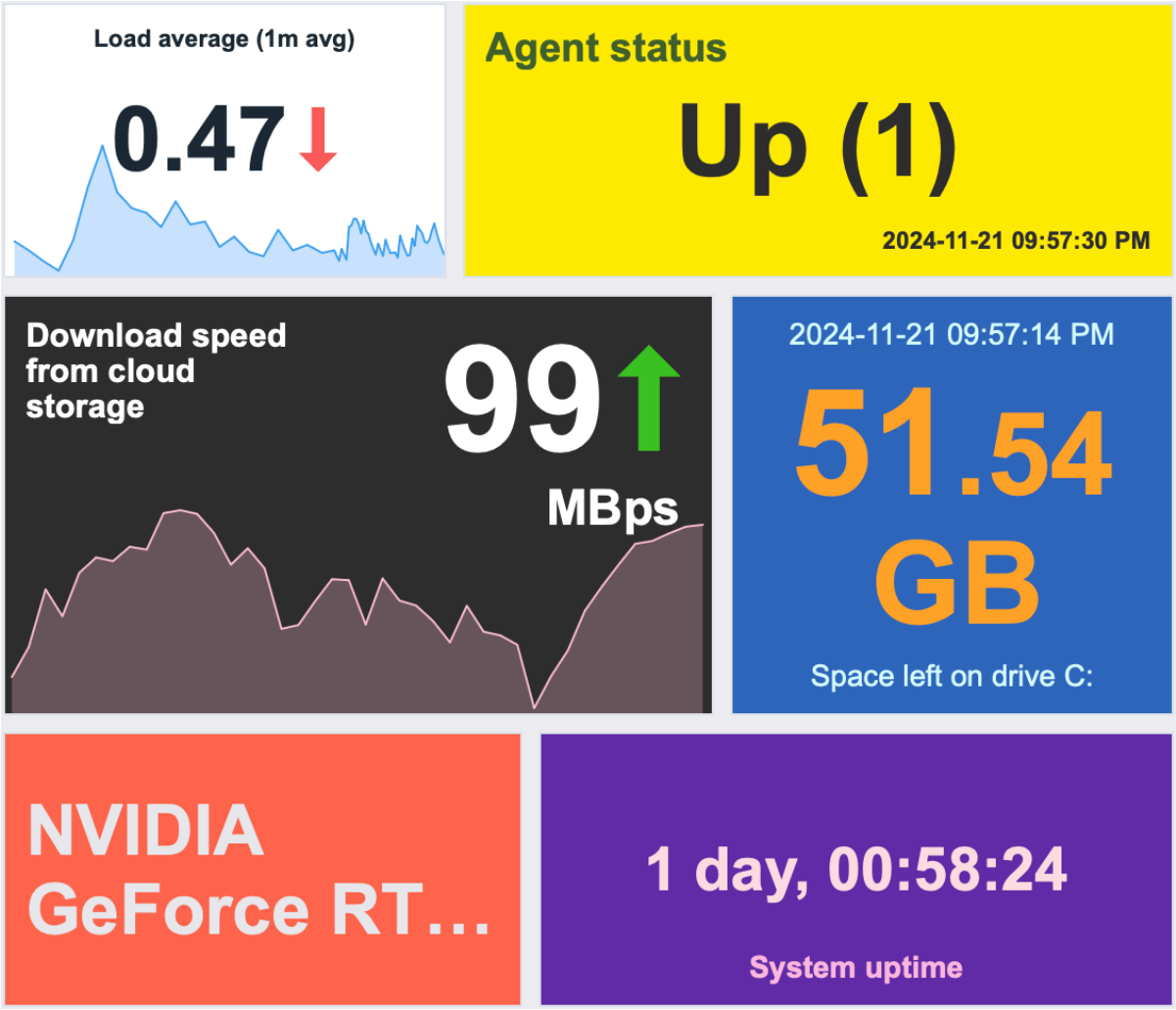
Medidor/Valor de elemento

Los widgets de Medidor/Valor de elemento muestran el valor de un único elemento, lo que resulta útil para vigilar métricas importantes, visualizar umbrales y detectar cambios repentinos en los datos.

Widget de medidor:

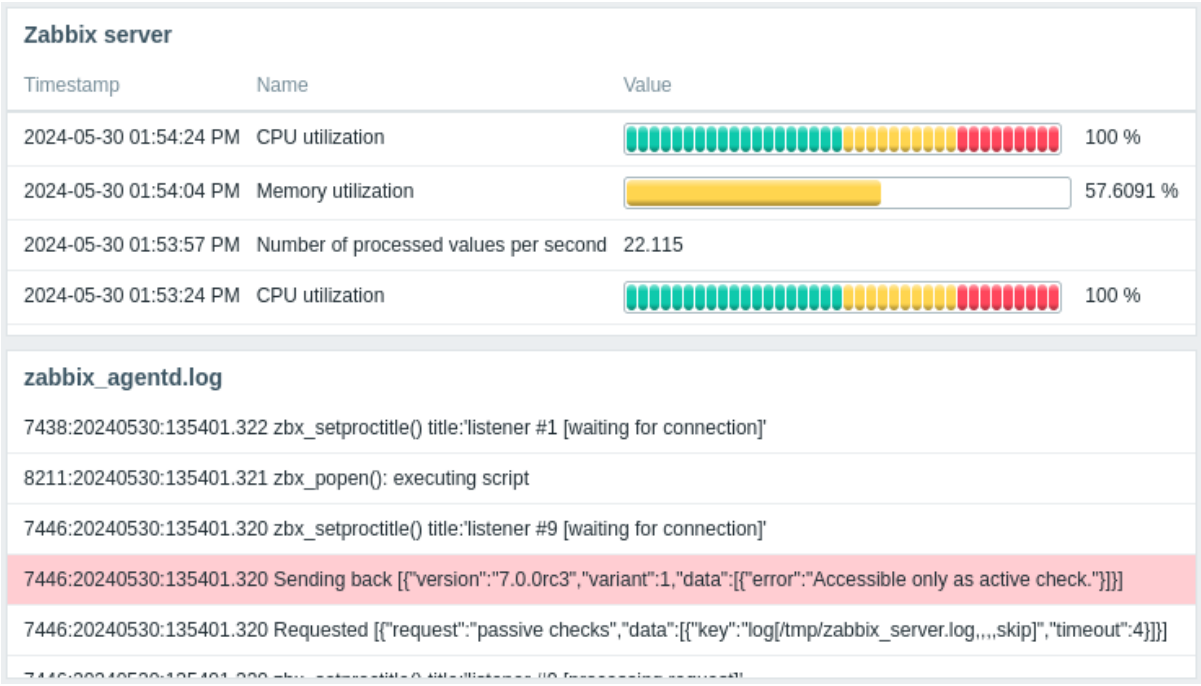


Widget de valor de elemento:



Historial de ítem

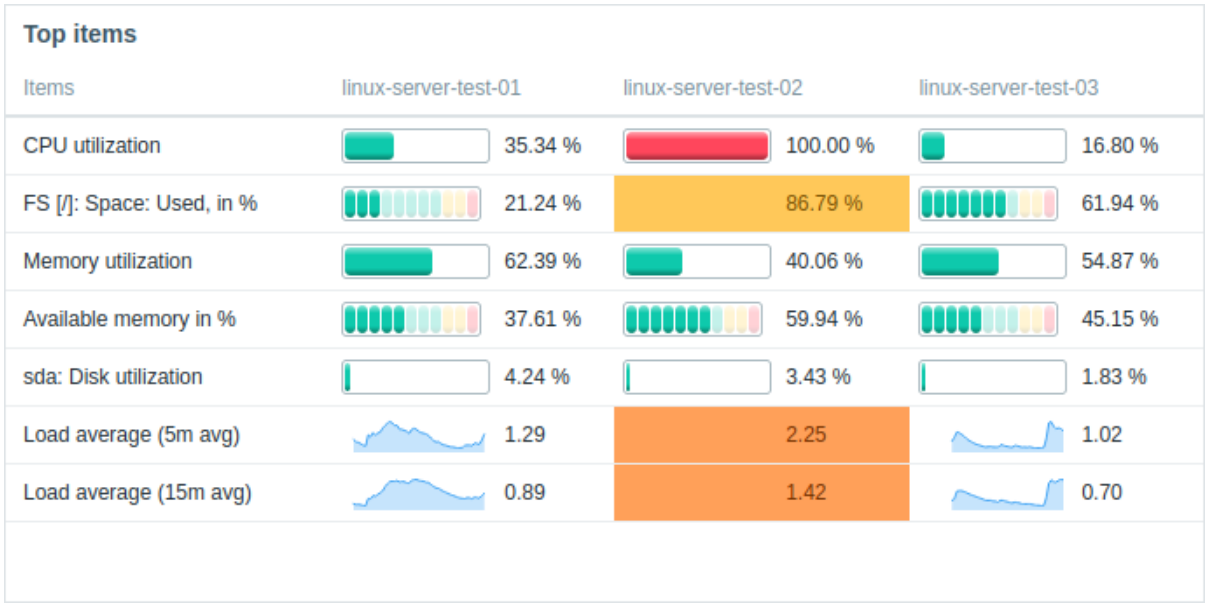
El widget Historial de ítem muestra los datos más recientes de todos los tipos de ítems (numéricos, texto, etc.) en formato de tabla. Puede mostrar barras de progreso e imágenes (útil para ítems de navegador) y puede resaltar valores (útil para la monitorización de archivos de registro).



Principales items

El widget Principales items muestra los valores más altos/bajos para los items seleccionados, proporcionando una visión rápida de

su rendimiento.



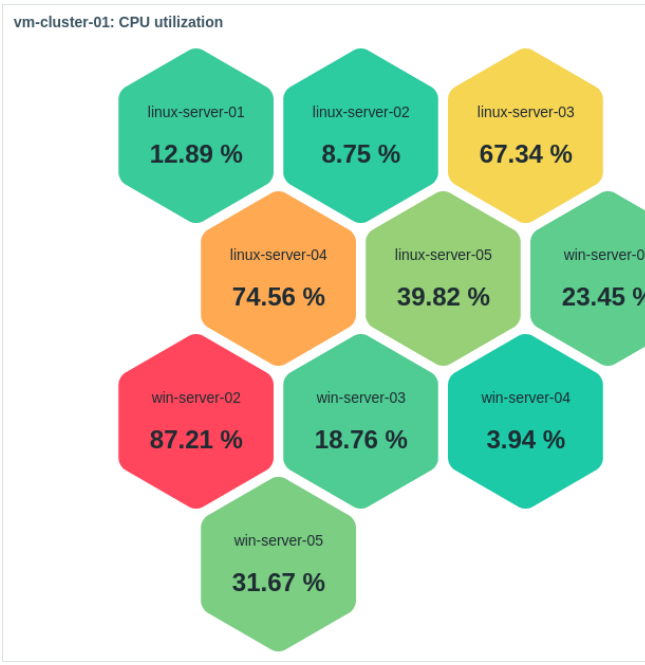
Ver también:

- Principales hosts: muestra los valores más altos/bajos para los items seleccionados de varios hosts.
- Principales triggers: muestra los triggers con el mayor número de problemas.

Panal de abeja

El widget Panal de abeja muestra equipos o elementos como celdas hexagonales en una cuadrícula. Este diseño facilita la detección de equipos/elementos problemáticos, la identificación de agrupaciones de problemas, la comparación de grupos de un vistazo y mucho más.



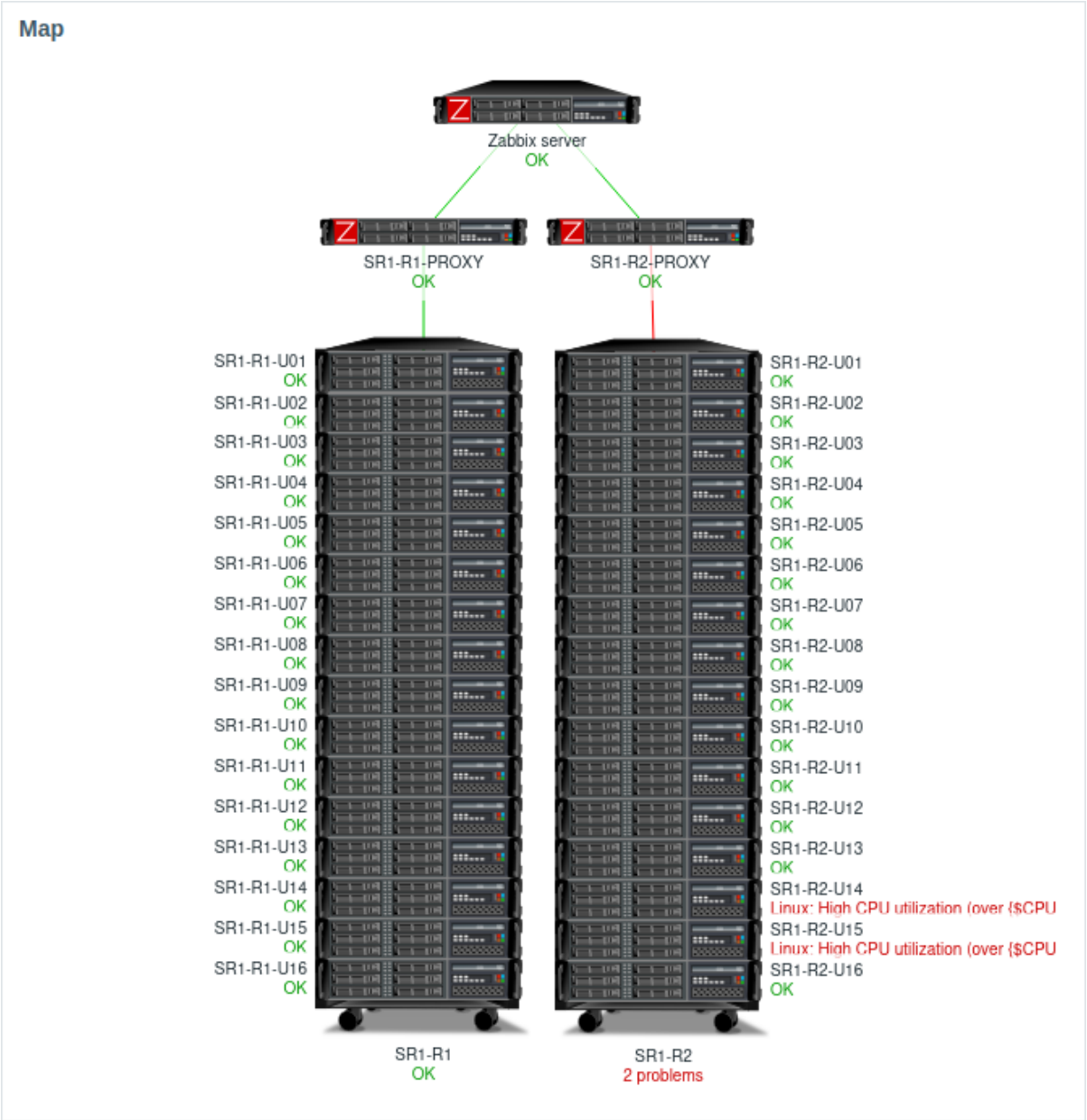


Mapas

Los widgets de mapas visualizan la topología de la red o la ubicación de los equipos, ayudando a evaluar el estado general y específico por ubicación.

Mapa

El widget Mapa puede mostrar un mapa de red o una visualización similar, proporcionándole una visión dinámica de su red.



Véase también:

- Árbol de navegación de mapas: muestra una jerarquía de mapas existentes y el recuento de problemas para cada mapa y grupo de mapas.
- Mapas favoritos: muestra una lista de accesos directos a los mapas marcados como favoritos por el usuario actual, ordenados alfabéticamente.

Geomapa

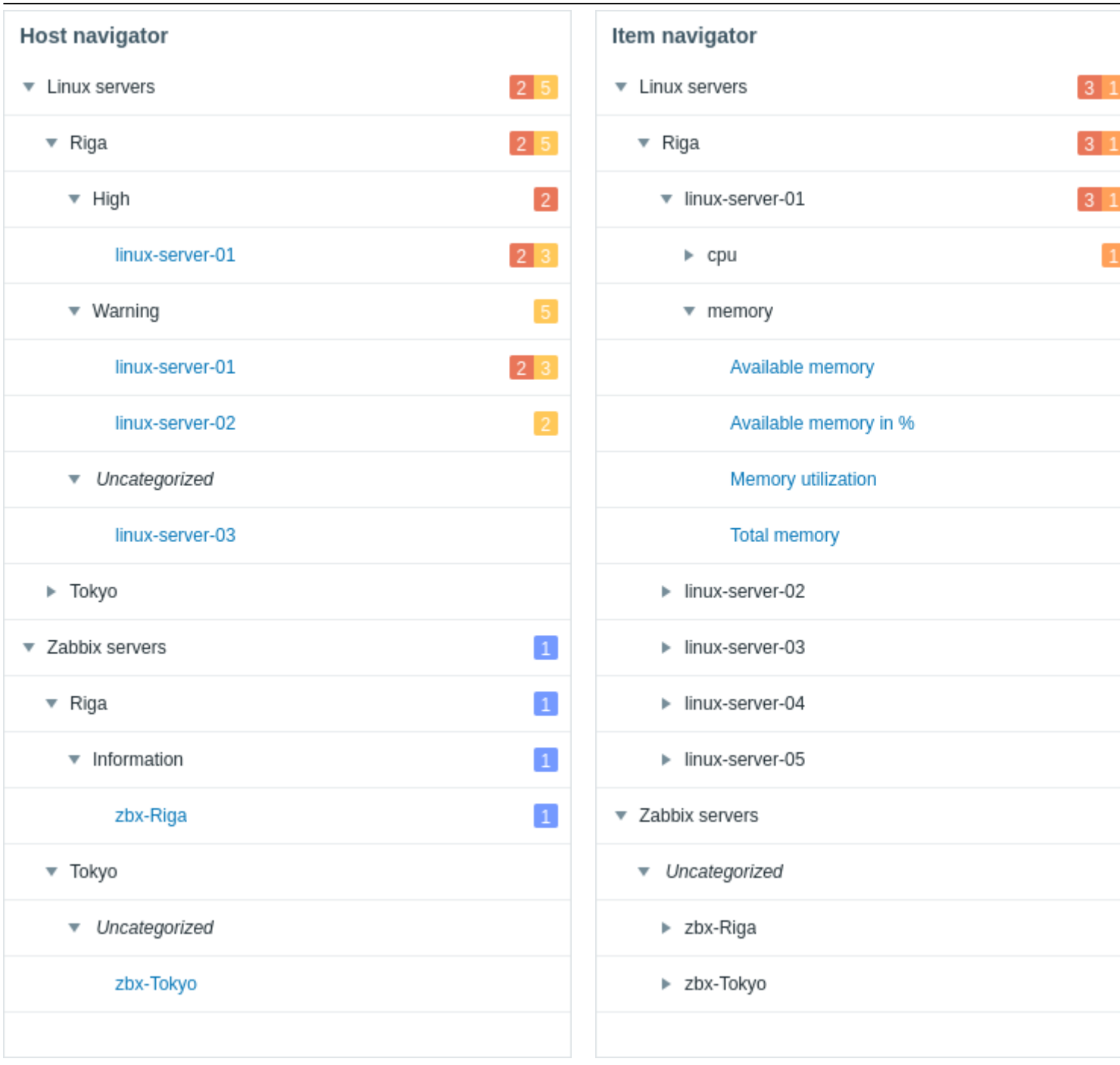
El widget Geomapa muestra los equipos como marcadores en un mapa interactivo basado en Leaflet, con colores de marcador que indican el problema más grave de cada equipo.

Detalles de Host/Elemento

Estos widgets proporcionan listas y vistas detalladas de hosts o elementos.

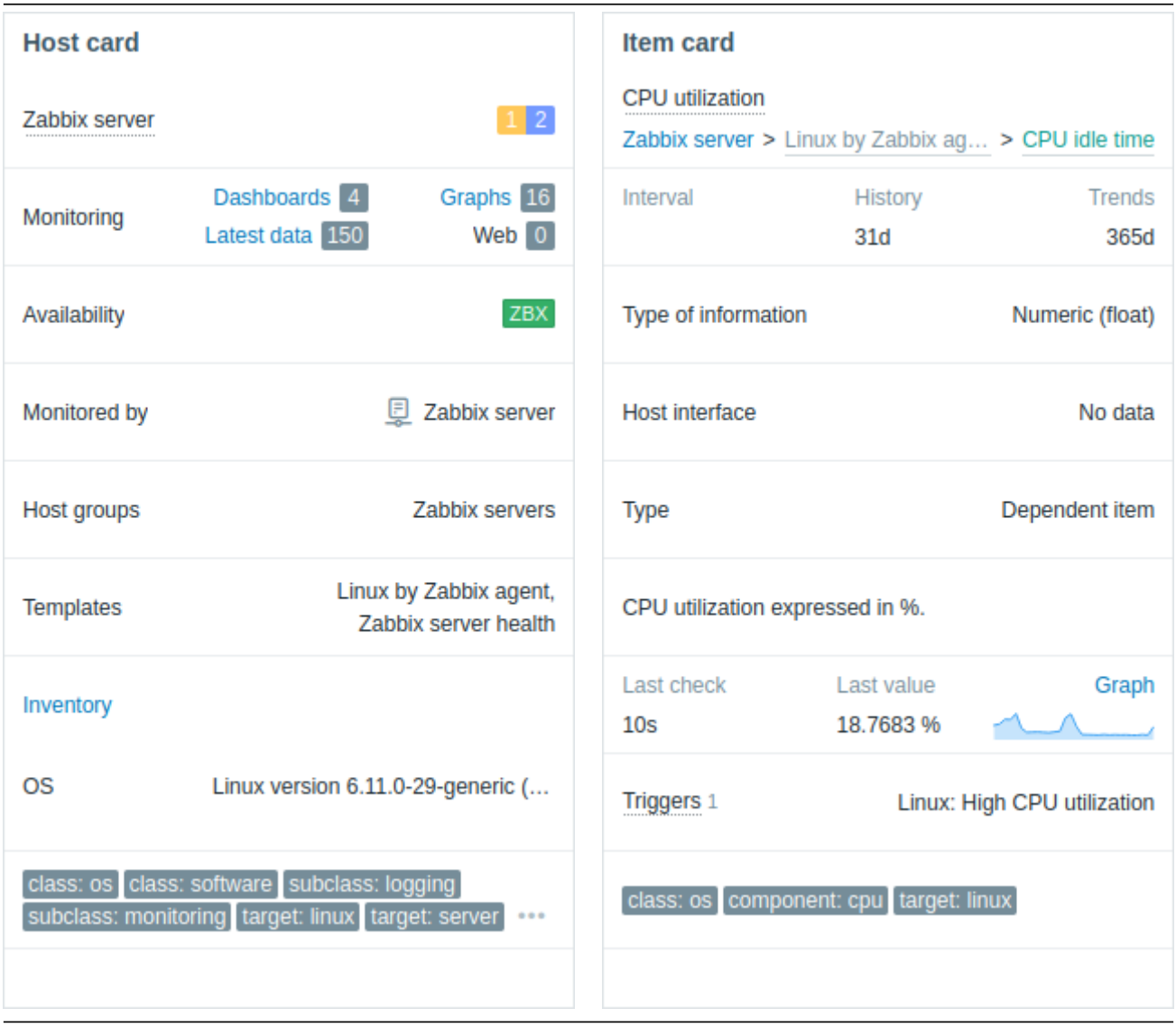
Navegador de equipos/elementos

Los widgets Navegador de equipos/Navegador de elementos muestran una lista de equipos/elementos basada en varias opciones de filtrado y agrupamiento. Estos widgets son más útiles para controlar lo que muestran otros widgets en función del equipo/elemento seleccionado.



Tarjeta de Equipo/Elemento

Los widgets de Tarjeta de equipo/Tarjeta de elemento muestran detalles de un único equipo/elemento, lo que le permite evaluar rápidamente el estado y la configuración de un equipo o elemento (disponibilidad del equipo, problemas, inventario, últimos datos del elemento, errores, descripción, etc.).

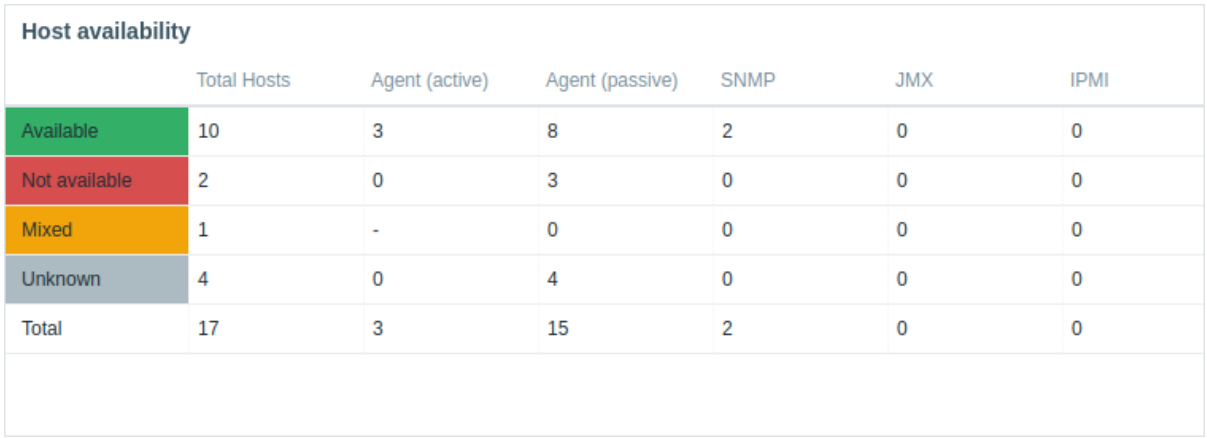


Estado e informes

Estos widgets muestran datos de estado (por ejemplo, disponibilidad, disparadores, problemas) así como informes.

Disponibilidad de equipos

El widget Disponibilidad de equipos muestra la disponibilidad de los equipos en los grupos seleccionados, lo que le permite supervisar el número de equipos que están activos o inactivos.

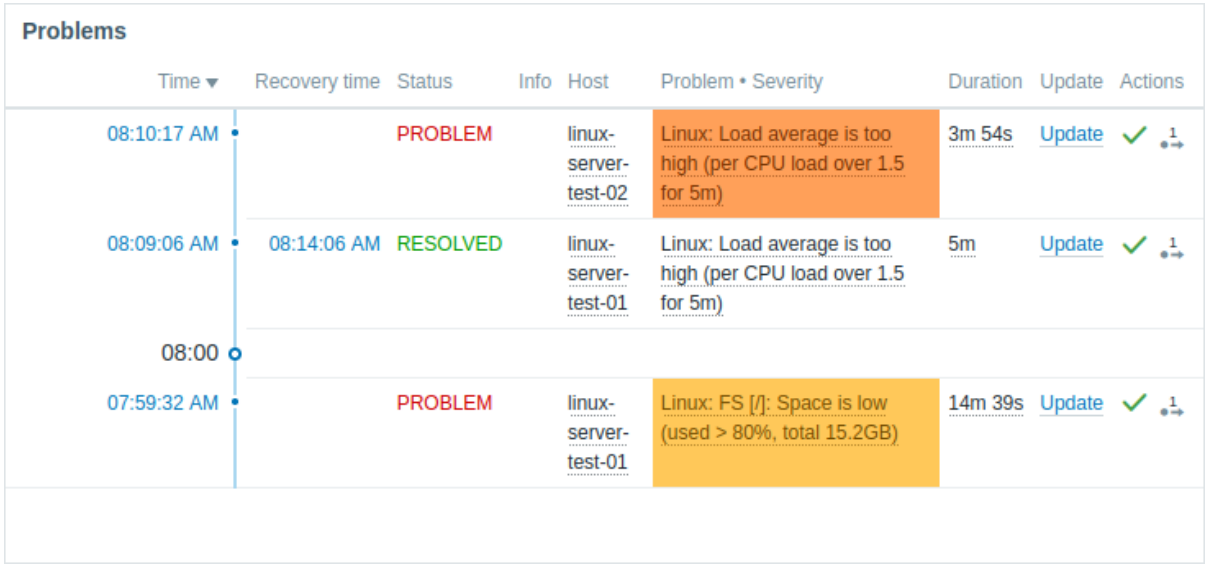


Host availability

	Total Hosts	Agent (active)	Agent (passive)	SNMP	JMX	IPMI
Available	10	3	8	2	0	0
Not available	2	0	3	0	0	0
Mixed	1	-	0	0	0	0
Unknown	4	0	4	0	0	0
Total	17	3	15	2	0	0

Problemas

El widget Problemas muestra los problemas actuales filtrados por diferentes parámetros (grupos de hosts, hosts, nombres de problemas, etc.), dándole una visión clara de lo que necesita atención.



Vea también:

- Problemas por severidad: muestra el número de problemas raíz agrupados por severidad.
- Hosts con problemas: muestra el número de problemas raíz por grupo de hosts y muestra la mayor severidad de problema en cada grupo.

Resumen de disparadores

El widget Resumen de disparadores muestra los estados actuales de los disparadores de los equipos como una tabla de bloques de colores. Proporciona una visión rápida y visual del estado y la actividad de los disparadores en diferentes equipos.



Registro de acciones

El widget Registro de acciones muestra detalles de las operaciones (notificaciones, comandos remotos) ejecutadas dentro de una acción. El registro detallado ayuda con la auditoría, la resolución de problemas y el monitoreo de la ejecución de cada acción.

Action log

Time ▾	Action	Media type	Recipient	Message	Status	Info
2025-08-15 07:58:30 AM	Report problems to Zabbix administrators	Email	Admin (Zabbix Administrator) *admin@example.com*	**Resolved in 1m 59s: Linux: Load average is too high (per CPU load over 1.5 for 5m)** Problem has been resolved at 10:58:30 on 2025.08.15 Problem name: Linux: Load average is too high (per CPU load over 1.5 for 5m) Problem duration: 1m 59s Host: Linux server Severity: Average Original problem ID: 45	Sent	
2025-08-15 07:56:30 AM	Report problems to Zabbix administrators	Email	Admin (Zabbix Administrator) *admin@example.com*	**Problem: Linux: Load average is too high (per CPU load over 1.5 for 5m)** Problem started at 10:56:30 on 2025.08.15 Problem name: Linux: Load average is too high (per CPU load over 1.5 for 5m) Host: Linux server Severity: Average Operational data: Load averages(1m 5m 15m): (6.256836 4.808105 3.398438), # of CPUs: 2 Original problem ID: 45	Sent	
2025-08-15 07:42:30 AM	Report problems to Zabbix administrators		Admin (Zabbix Administrator)		Failed	i

Informe de SLA

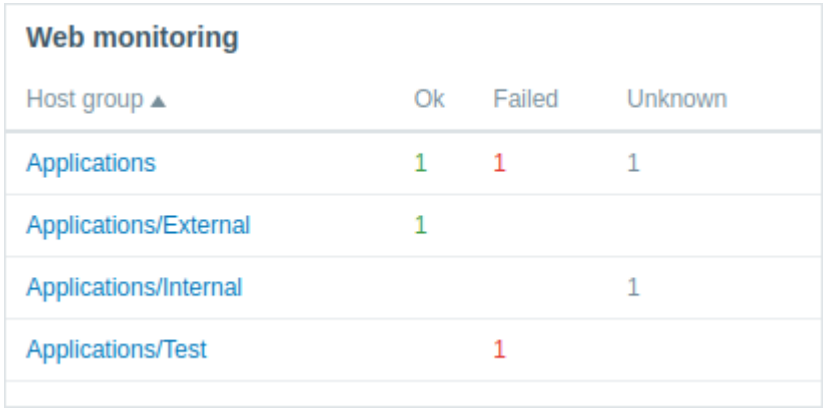
El widget de informe de SLA muestra informes de SLA, ayudándole a supervisar el rendimiento del servicio en comparación con los objetivos definidos.

SLA report

Day	SLO	SLI	Uptime	Downtime	Error budget	Excluded downtimes
2025-09-04	99.9%	99.8437	5h 40m 50s	28s	-8s	
2025-09-03	99.9%	99.9479	23h 59m 15s	45s	41s	
2025-09-02	99.9%	99.9190	23h 58m 50s	1m 10s	16s	
2025-09-01	99.9%	99.9251	22h 58m 58s	1h 1m 2s	21s	2025-09-01 01:00 AM Maintenance: 1h
2025-08-31	99.9%	99.6863	23h 55m 29s	4m 31s	-3m 5s	
2025-08-30	99.9%	99.9190	23h 58m 50s	1m 10s	16s	
2025-08-29	99.9%	99.7095	23h 55m 49s	4m 11s	-2m 45s	
2025-08-28	99.9%	99.9421	23h 59m 10s	50s	36s	
2025-08-27	99.9%	99.7188	23h 55m 57s	3m 15s	-2m 37s	
2025-08-26	99.9%	99.7755	23h 56m 46s	3m 14s	-1m 48s	
2025-08-25	99.9%	99.9094	22h 58m 45s	1h 1m 15s	8s	2025-08-25 01:00 AM Maintenance: 1h

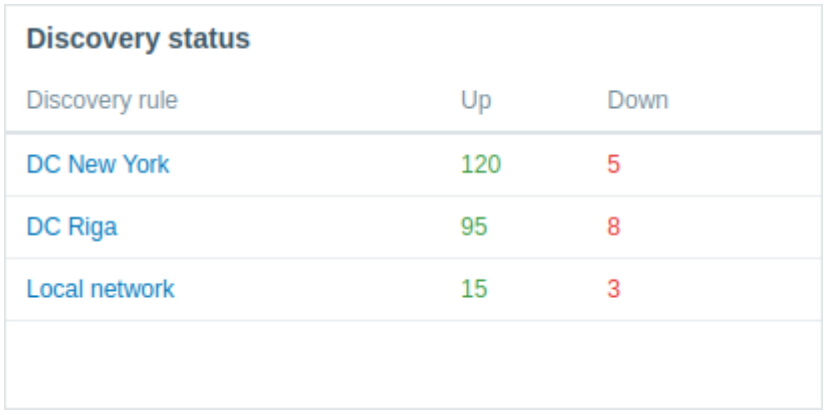
Supervisión web

El widget de Supervisión web muestra el estado de los escenarios web activos.



Estado de descubrimiento

El widget Estado de descubrimiento muestra el estado de los dispositivos descubiertos por las reglas de descubrimiento de red habilitadas. Le ayuda a supervisar si el descubrimiento de red funciona correctamente y el número de dispositivos que están activos o inactivos.



Utilidades del panel e información del sistema

Estos widgets proporcionan información y herramientas complementarias para mejorar sus paneles.

Información del sistema

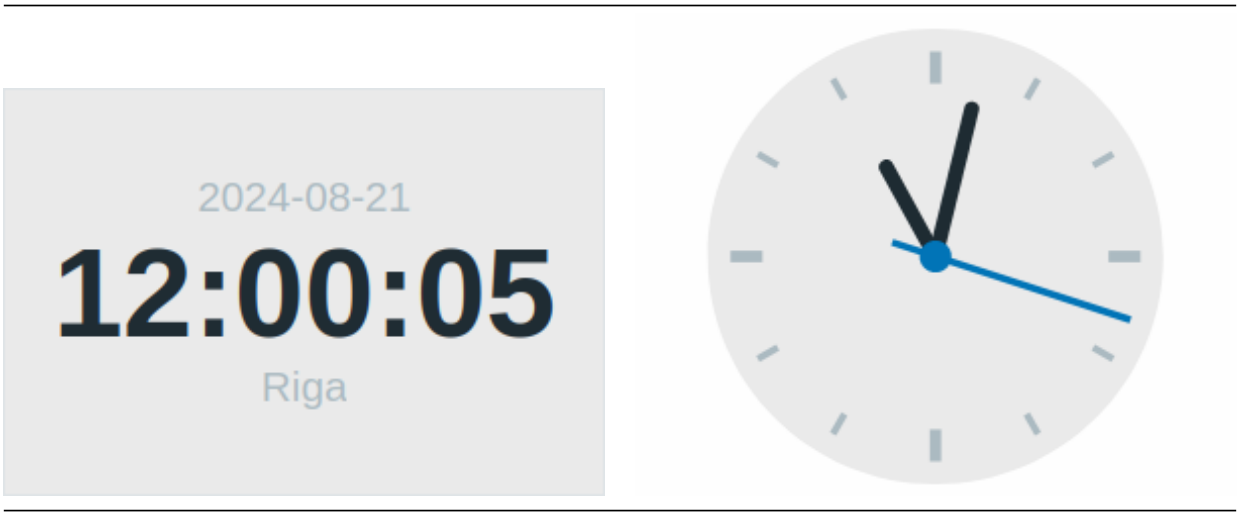
El widget Información del sistema muestra un resumen de los datos clave del servidor Zabbix y del sistema o detalles de los nodos de alta disponibilidad.

System information

Parameter	Value	Details
Zabbix server is running	Yes	localhost:10051
Zabbix server version	8.0.0	Up to date
Zabbix frontend version	8.0.0	Up to date
Software update last checked	2025-08-30	
Latest release	8.0.0	Release notes
Number of hosts (enabled/disabled)	2	2 / 0
Number of templates	312	
Number of items (enabled/disabled/not supported)	176	165 / 0 / 11
Number of triggers (enabled/disabled [problem/ok])	89	89 / 0 [4 / 85]
Number of users (online)	2	1
Required server performance, new values per second	2.68	
Global scripts on Zabbix server	Disabled	
High availability cluster	Disabled	

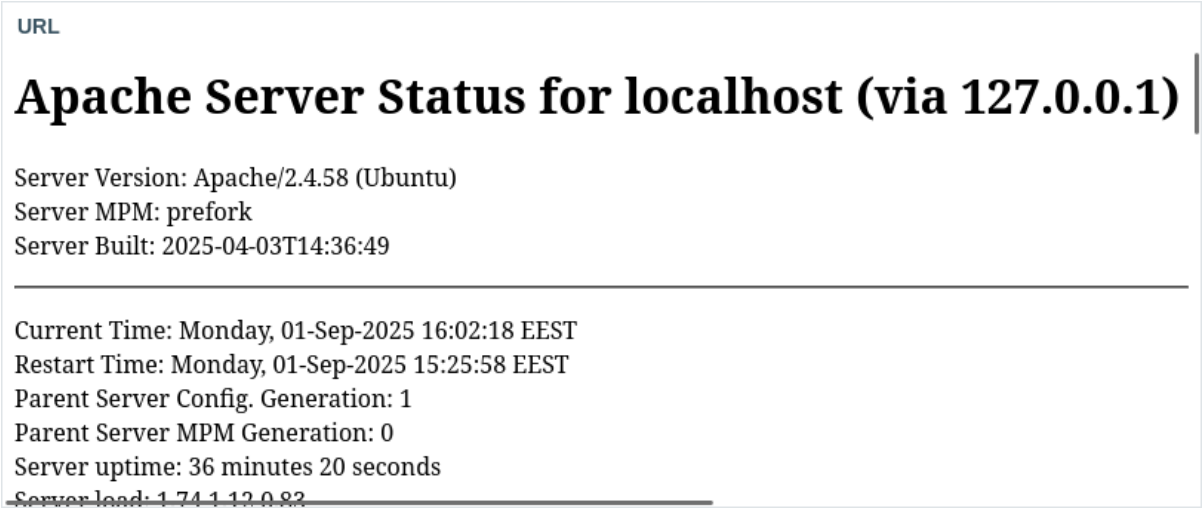
Reloj

El widget Reloj muestra la hora local, del servidor o del equipo en formato analógico o digital.



URL

El widget URL muestra el contenido recuperado de una URL.



Parámetros del widget del dashboard

Descripción general

Esta página describe los parámetros que son comunes a todos los widgets del panel, así como los parámetros dinámicos que permiten que varios widgets intercambien datos de configuración entre sí o con el panel.

Para ver los parámetros específicos de cada widget, utilice la barra lateral para navegar a la página correspondiente del widget.

Parámetros comunes

Los siguientes parámetros son comunes para todos los widgets:

Nombre	Introduzca un nombre para el widget.
Intervalo de actualización	Configure el intervalo de actualización predeterminado. Los intervalos de actualización predeterminados para los widgets van desde *Sin actualización* hasta *15 minutos*, según el tipo de widget. Por ejemplo: - *Sin actualización* para el widget *URL*; - *1 minuto* para el widget *Registro de acciones*; - *15 minutos* para el widget *Reloj*. Los intervalos de actualización se pueden establecer en un valor predeterminado para todos los usuarios. Cambie el dashboard al modo de edición, haga clic en el botón editar un widget y seleccione el intervalo de actualización deseado en la lista desplegable. Cada usuario también puede establecer su propio intervalo de actualización del widget. En el modo de visualización del dashboard, haga clic en el botón de tres puntos de un widget y seleccione el intervalo de actualización deseado en la lista desplegable. Tenga en cuenta que el intervalo de actualización personalizado de un usuario tiene prioridad sobre la configuración del widget y se conserva incluso cuando se modifica la configuración del widget.
Mostrar encabezado	Marque la casilla para mostrar permanentemente el encabezado del widget. Si no está marcada, el encabezado se oculta para ahorrar espacio y solo se hace visible al pasar el ratón sobre el widget (tanto en el modo de visualización como en el modo de edición). El encabezado también se muestra parcialmente al arrastrar un widget a una nueva ubicación.

Parámetros dinámicos

Varios parámetros de widgets (por ejemplo, *Equipos*, *Anular equipo*, *Período de tiempo*) admiten fuentes de datos dinámicas. En lugar de configurar valores estáticos, puede vincular estos parámetros a otros widgets o al propio panel. Los widgets se actualizan automáticamente en función de las selecciones o transmisiones desde otras partes del panel.

Compatibilidad de widgets

Algunos widgets pueden transmitir datos de configuración a otros widgets, algunos pueden escuchar datos y algunos pueden hacer ambas cosas. Por ejemplo:

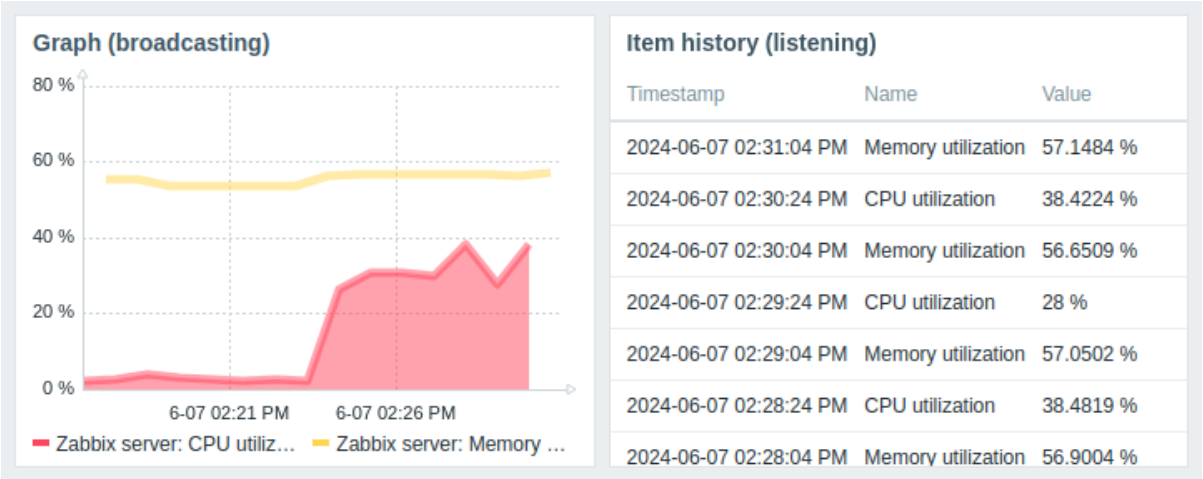
- El widget *Registro de acciones* solo puede recuperar datos de periodo de tiempo de los widgets *Gráfico*, *Gráfico (clásico)* y *Prototipo de gráfico*.
- El widget *Geomap* puede transmitir datos de host a widgets que los escuchen (*Honeycomb*, *Top items*, etc.) y también puede escuchar datos de grupo de hosts y de host de widgets que los transmitan (*Honeycomb*, *Problem hosts*, etc.).
- El widget *Reloj* no puede transmitir ni escuchar ningún dato.

La siguiente tabla describe las capacidades de transmisión y escucha de cada widget.

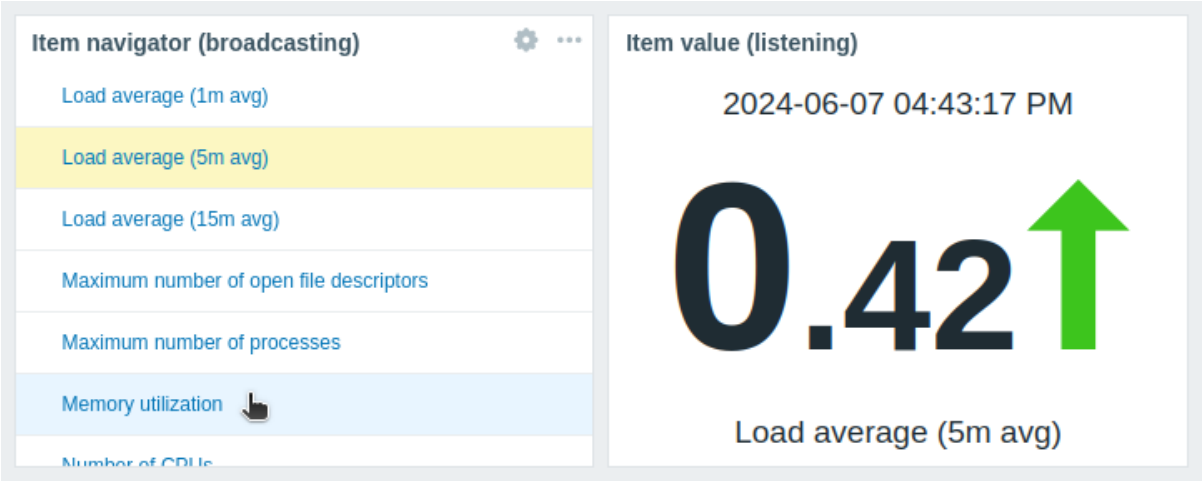
Widget	Transmite	Escucha
Registro de acciones	-	Periodo de tiempo
Reloj	-	-
Estado de descubrimiento	-	-
Gráficos favoritos	-	-
Mapas favoritos	-	-
Medidor	-	Hosts, Items
Geomap	Hosts	Grupos de hosts, Hosts
Gráfico	Periodo de tiempo, Items	Hosts, Items, Periodo de tiempo
Gráfico (clásico)	Periodo de tiempo	Hosts, Items, Gráficos, Periodo de tiempo
Prototipo de gráfico	Periodo de tiempo	Hosts, Periodo de tiempo
Honeycomb	Hosts, Items	Grupos de hosts, Hosts
Disponibilidad de host	-	Grupos de hosts
Tarjeta de host	-	Hosts
Navegador de hosts	Hosts	Grupos de hosts
Tarjeta de item	-	Hosts, Items, Periodo de tiempo
Histórico de item	Items	Hosts, Periodo de tiempo
Navegador de items	Items	Grupos de hosts, Hosts
Valor de item	-	Hosts, Items, Periodo de tiempo
Mapa	Grupos de hosts, Hosts	Mapas
Árbol de navegación de mapas	Mapas	-
Gráfico circular	-	Items, Periodo de tiempo
Problem hosts	Grupos de hosts	Grupos de hosts, Hosts
Problemas	Eventos	Grupos de hosts, Hosts
Problemas por severidad	Grupos de hosts	Grupos de hosts, Hosts
Gráfico de dispersión	Periodo de tiempo, Items	Hosts, Items, Periodo de tiempo
Informe SLA	-	-
Información del sistema	-	-
Top hosts	Hosts	Grupos de hosts, Hosts
Top items	-	Grupos de hosts, Hosts
Top triggers	-	Periodo de tiempo
Resumen de triggers	-	Grupos de hosts, Hosts
URL	-	Hosts
Supervisión web	Grupos de hosts	Grupos de hosts, Hosts

Los widgets difieren en cómo **transmiten** datos a otros widgets.

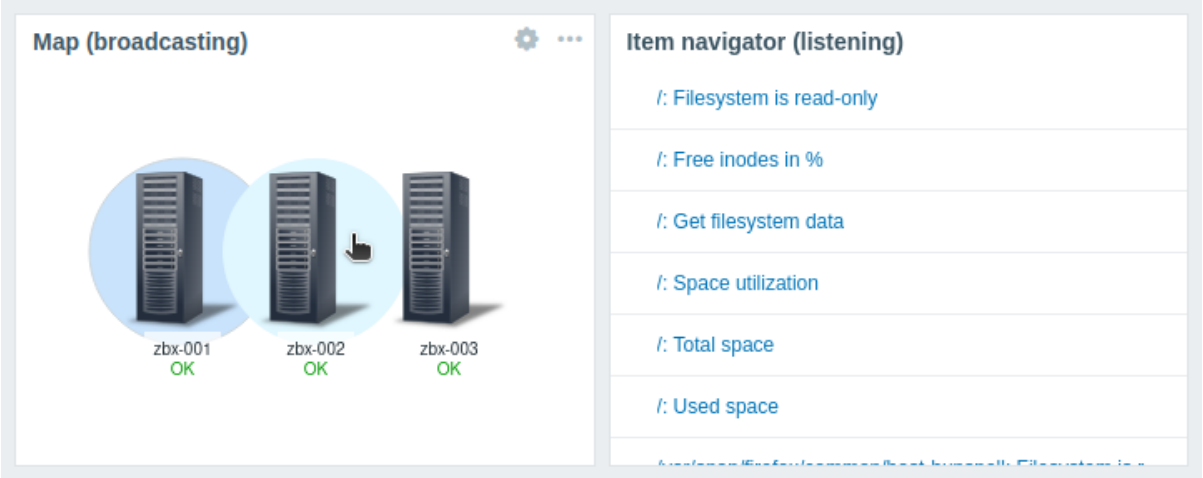
Los widgets que admiten la transmisión comienzan a hacerlo automáticamente cuando se crean. Por ejemplo, el widget *Gráfico* transmite inmediatamente los datos del período de tiempo a los widgets que escuchan.



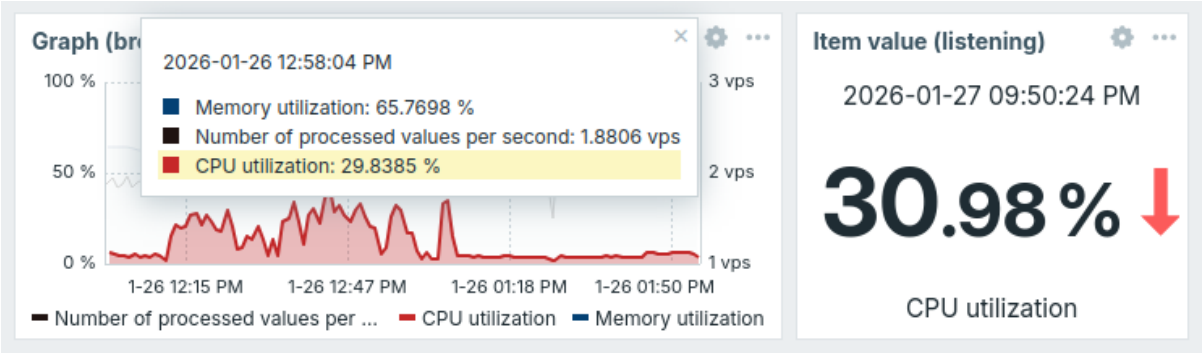
Los widgets que admiten la selección de entidades transmiten automáticamente los datos de la primera entidad disponible. Por ejemplo, el widget *Navegador de items* transmite los datos del primer item de su lista de items. Cuando se selecciona un item diferente, transmite los datos de ese item. Al pasar el ratón por encima, el item se resalta en azul claro; al seleccionarlo, se resalta en amarillo.



El widget *Mapa* se comporta de manera similar, transmitiendo los datos del primer elemento más cercano a la esquina superior izquierda del área visible del widget (para el widget *Geomapa*, el más cercano al centro). Cuando se selecciona un elemento diferente, transmite los datos de ese elemento. Al pasar el ratón por encima, el elemento se resalta en azul claro; al seleccionarlo, se resalta en azul oscuro.

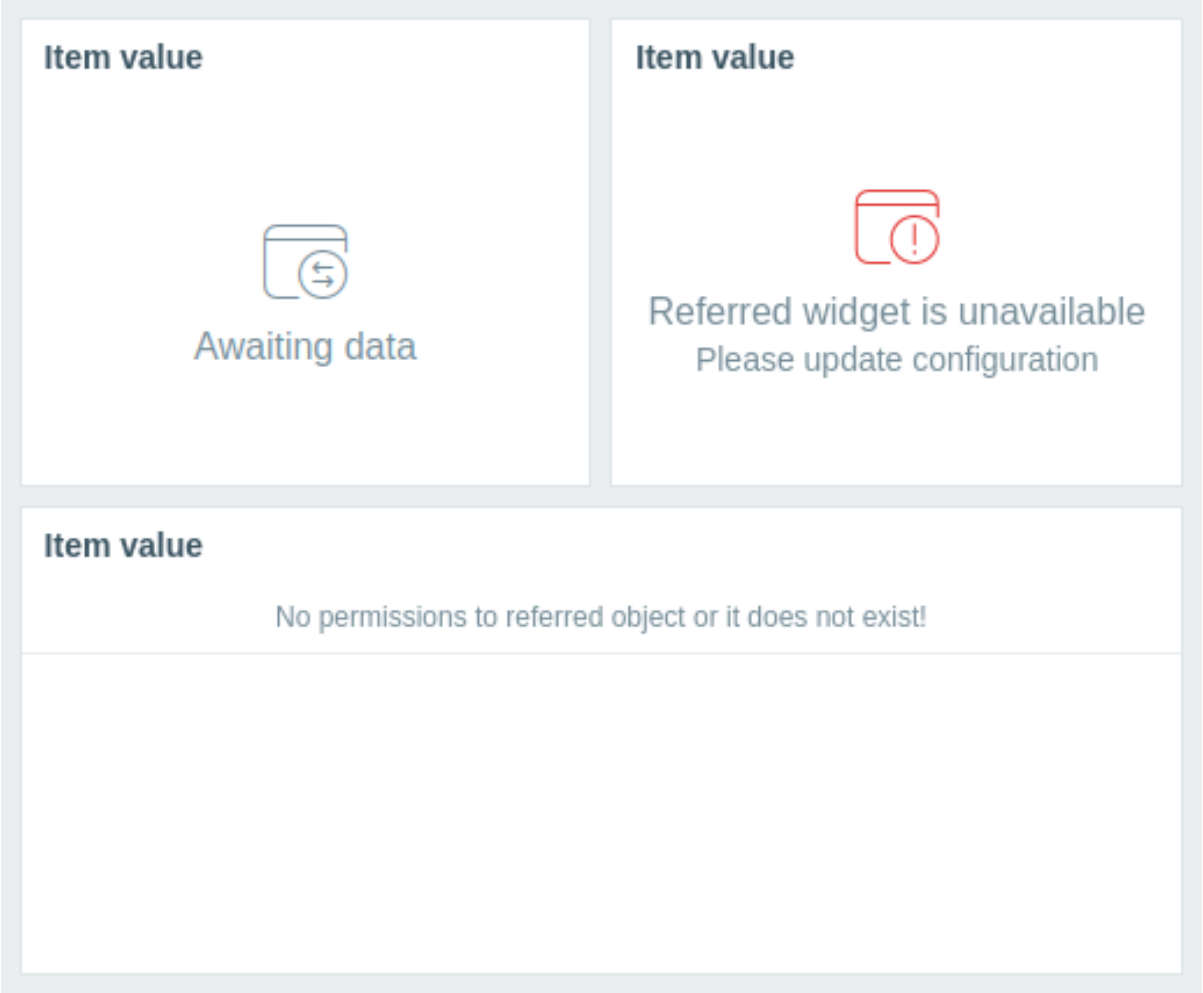


Los widgets como *Gráfico* y *Diagrama de dispersión* pueden transmitir datos de items cuando se selecciona un item en sus tooltips:



Los widgets también tienen comportamientos específicos al **escuchar** datos de otros widgets:

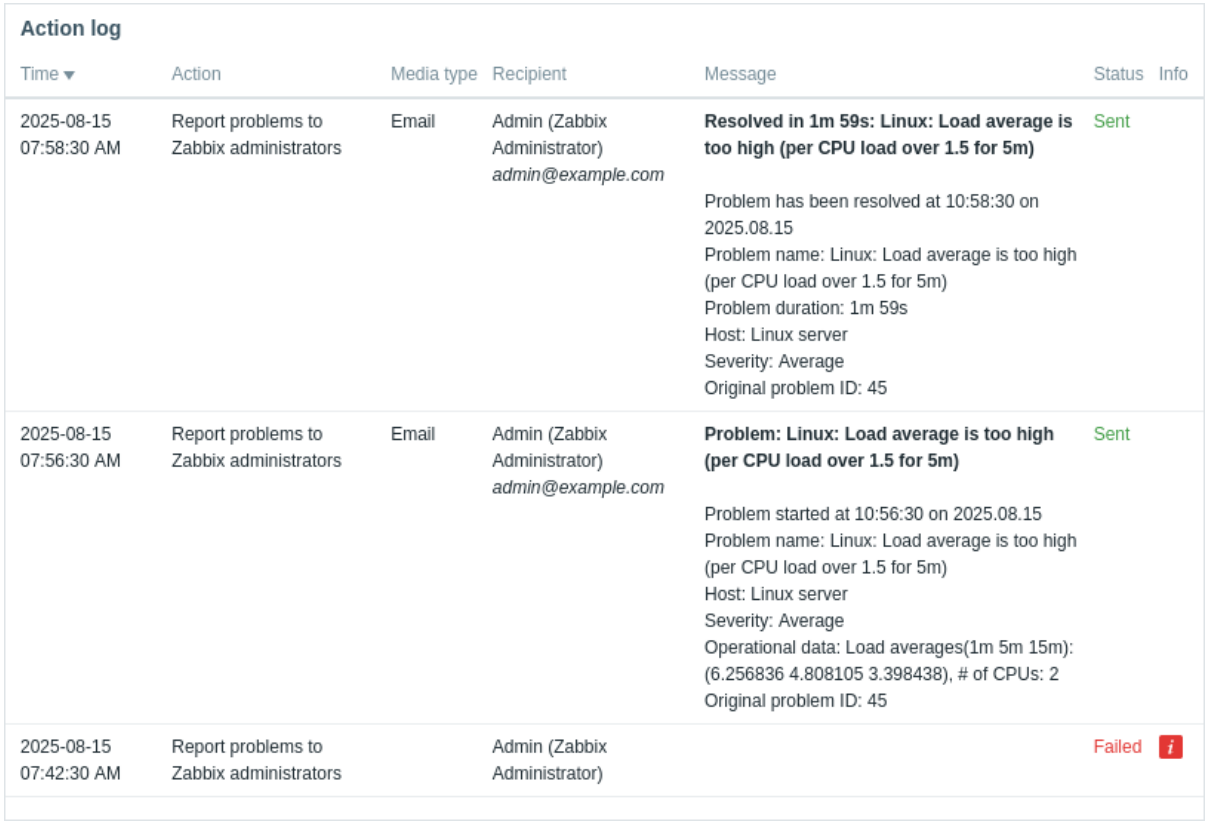
- Si el widget fuente de datos no está transmitiendo datos, el widget que escucha entra en estado *Esperando datos*.
- Si el widget fuente de datos ha sido eliminado, reemplazado por un widget incompatible o movido a otra página del dashboard, el widget que escucha entra en estado *El widget referido no está disponible*.
- Si el host especificado en la fuente de datos (widget o dashboard) no tiene la entidad configurada en el widget que escucha (item, gráfico, mapa, etc.) o si el usuario no tiene permisos para acceder al host, el widget que escucha muestra el siguiente mensaje: *"¡Sin permisos para el objeto referido o no existe!"*



Registro de acciones

Descripción general

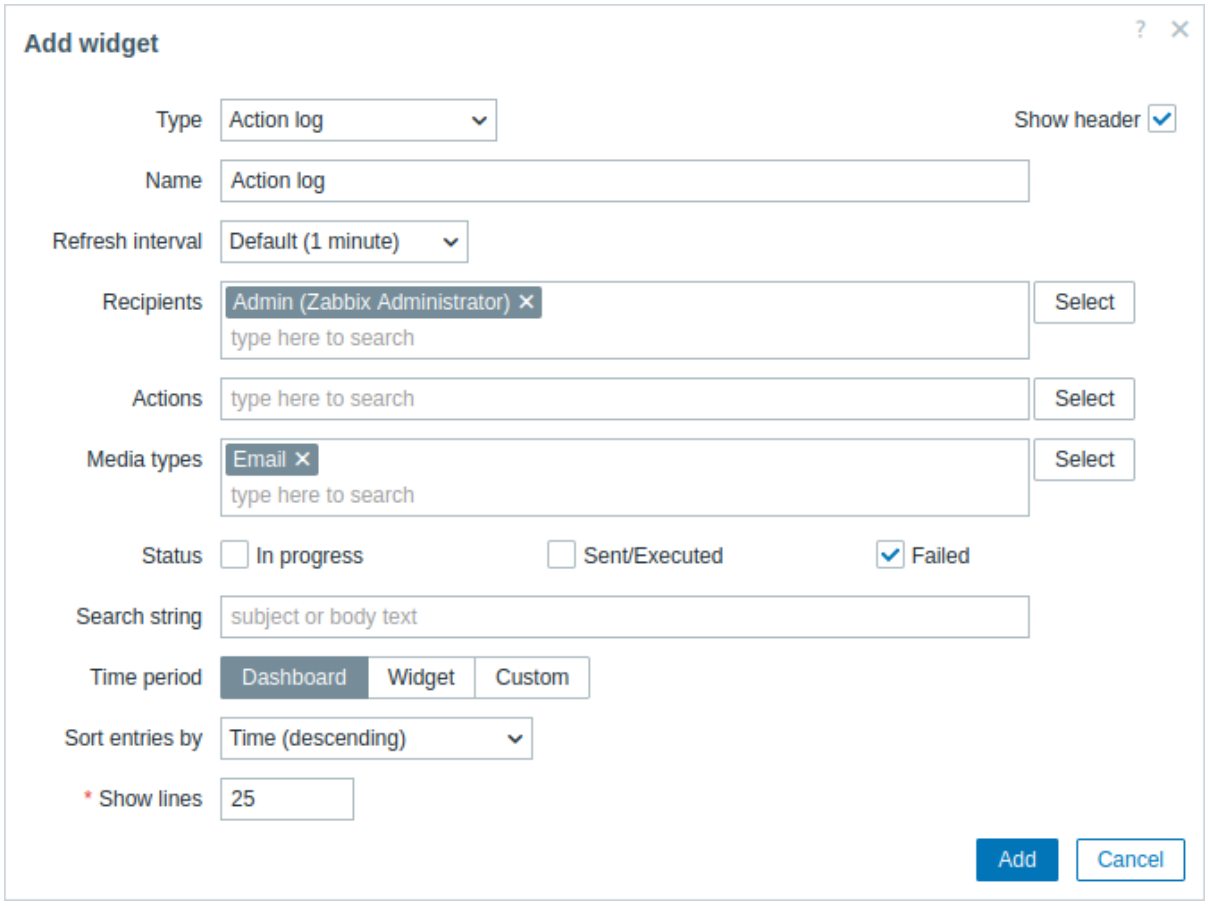
El widget *Registro de acciones* muestra detalles de las operaciones (notificaciones, comandos remotos) ejecutadas dentro de una acción. El registro detallado ayuda con la auditoría, la resolución de problemas y el monitoreo de la ejecución de cada acción.



Muestra los mismos datos que *Informes* > *Registro de acciones* y puede mostrar hasta 1000 registros.

Configuración

Para configurar, seleccione *Registro de acciones* como tipo:



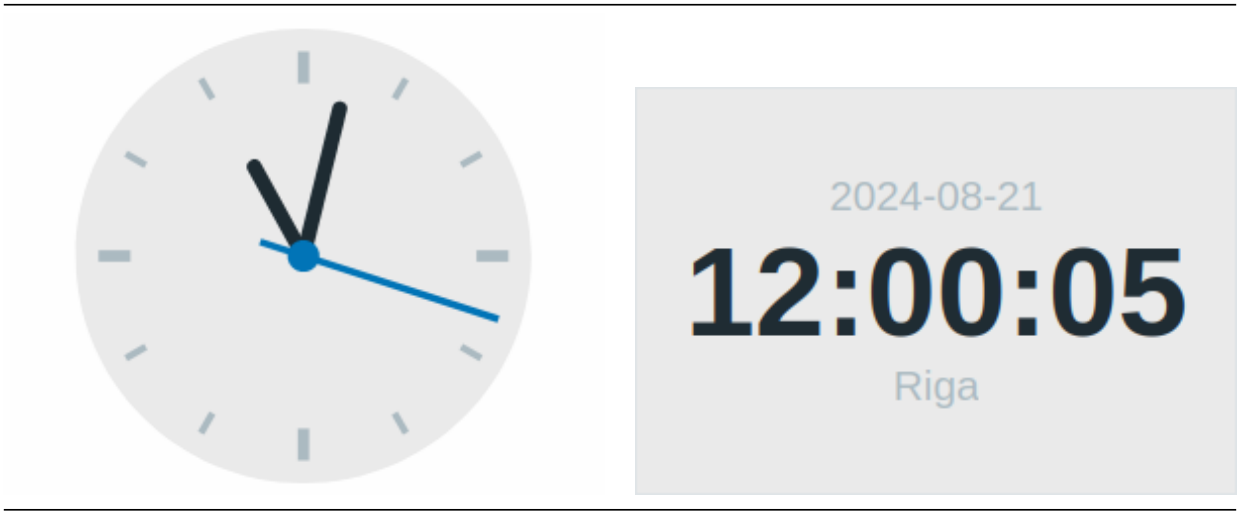
Además de los parámetros que son comunes para todos los widgets, puede establecer las siguientes opciones específicas:

Destinatarios	Filtrar entradas por destinatarios. Este campo es autocompletable, por lo que al comenzar a escribir el nombre de un destinatario se mostrará un menú desplegable con los destinatarios coincidentes. Si no se seleccionan destinatarios, se mostrarán los detalles de las operaciones de acción para todos los destinatarios.
Acciones	Filtrar entradas por acciones. Este campo es autocompletable, por lo que al comenzar a escribir el nombre de una acción se mostrará un menú desplegable con las acciones coincidentes. Si no se seleccionan acciones, se mostrarán los detalles de las operaciones de acción para todas las acciones.
Tipos de medio	Filtrar entradas por tipos de medio. Este campo es autocompletable, por lo que al comenzar a escribir el nombre de un tipo de medio se mostrará un menú desplegable con los tipos de medio coincidentes. Si no se seleccionan tipos de medio, se mostrarán los detalles de las operaciones de acción para todos los tipos de medio.
Estado	Marque la casilla para filtrar las entradas por el estado correspondiente: **En progreso**: se muestran las operaciones de acción que están en progreso; **Enviado/Ejecutado**: se muestran las operaciones de acción que han enviado una notificación o han sido ejecutadas; **Fallido**: se muestran las operaciones de acción que han fallado.
Cadena de búsqueda	Filtrar entradas por el contenido del mensaje/comando remoto. Si introduce una cadena aquí, solo se mostrarán aquellas operaciones de acción cuyo mensaje/comando remoto contenga la cadena introducida. Las macros no se resuelven.
Período de tiempo	Filtrar entradas por período de tiempo. Seleccione la fuente de datos para el período de tiempo: **Panel**: usar el selector de período de tiempo del panel; **Widget**: usar un widget compatible (establecido en el parámetro *Widget*); **Personalizado**: usar un período de tiempo personalizado establecido en los parámetros *Desde* y *Hasta*; si se establece, se mostrará un icono de reloj en la esquina superior derecha del widget, indicando el tiempo establecido al pasar el ratón. Tenga en cuenta que, independientemente de la configuración de *Período de tiempo* del widget, los widgets compatibles aún pueden usarlo como fuente de datos para el período de tiempo.
Widget	Introduzca o seleccione un widget compatible como fuente de datos para el período de tiempo. Este parámetro está disponible si *Período de tiempo* está establecido en "Widget".
Desde	Introduzca o seleccione el inicio del período de tiempo. Se admite la sintaxis de tiempo relativo (`now`, `now/d`, `now/w-1w`, etc.). Este parámetro está disponible si *Período de tiempo* está establecido en "Personalizado".
Hasta	Introduzca o seleccione el final del período de tiempo. Se admite la sintaxis de tiempo relativo (`now`, `now/d`, `now/w-1w`, etc.). Este parámetro está disponible si *Período de tiempo* está establecido en "Personalizado".
Ordenar entradas por	Ordenar entradas por: **Hora** (descendente o ascendente); **Tipo** (descendente o ascendente); **Estado** (descendente o ascendente); **Destinatario** (descendente o ascendente).
Mostrar líneas	Establezca cuántas líneas del registro de acciones se mostrarán en el widget.

Reloj

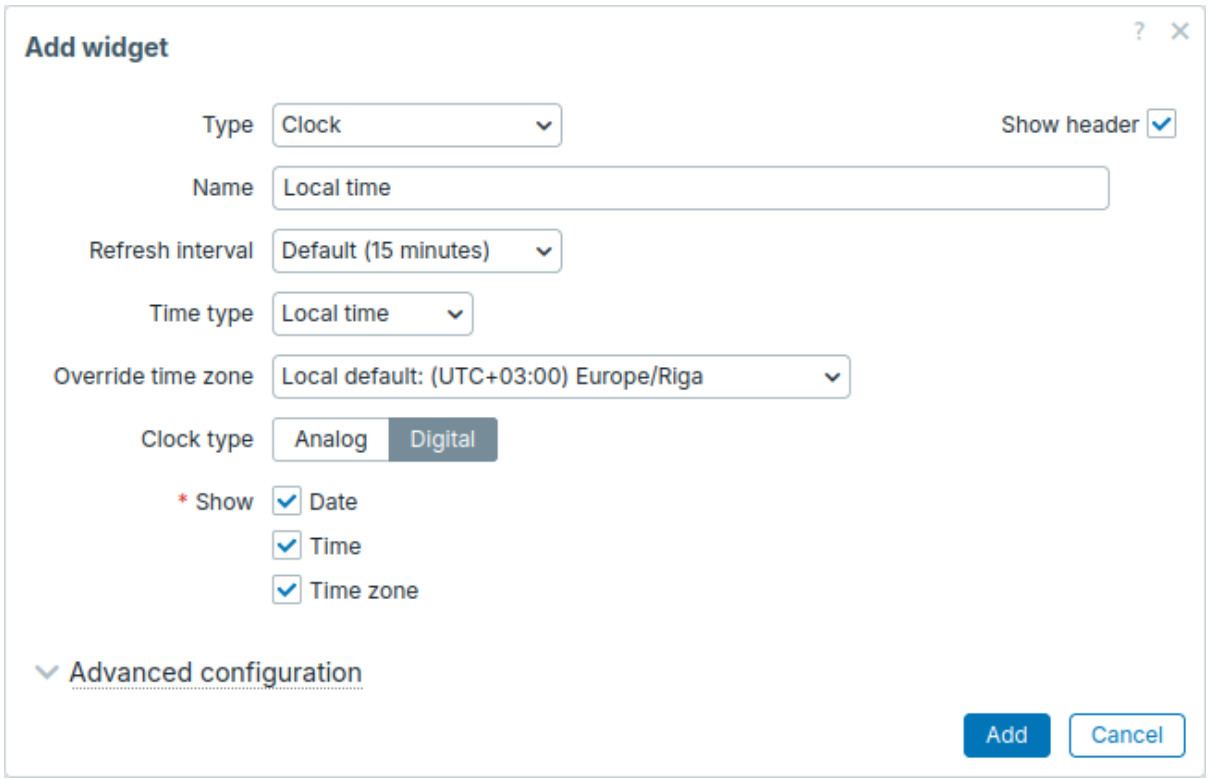
Descripción general

El widget *Reloj* muestra la hora local, del servidor o del host en formato analógico o digital.



Configuración

Para configurar, seleccione *Reloj* como tipo:



Además de los parámetros comunes para todos los widgets, puede establecer las siguientes opciones específicas:

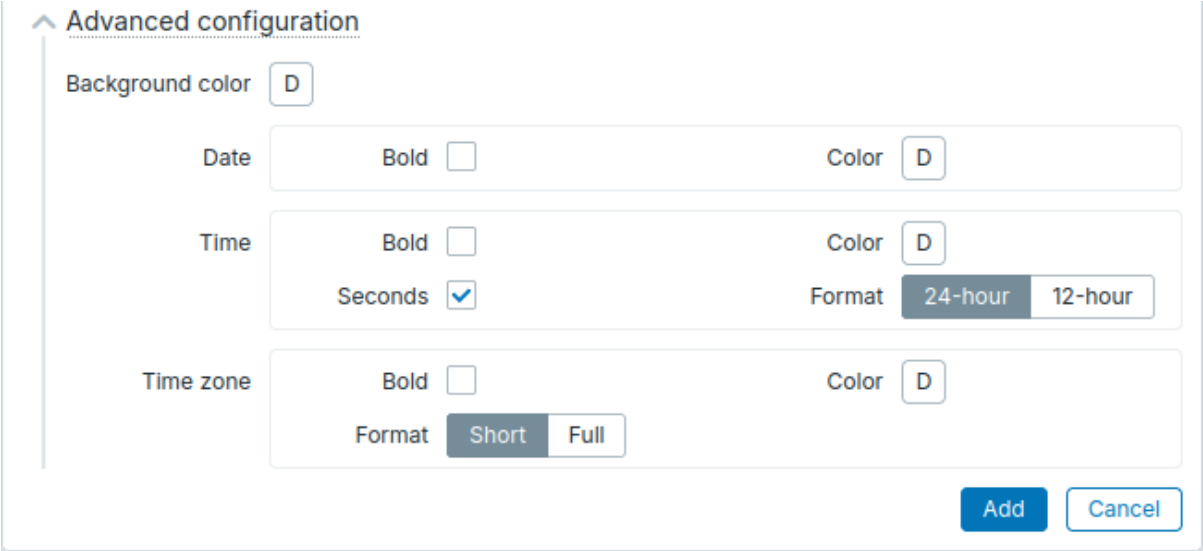
Tipo de hora	Seleccione la fuente de hora: **Hora local**: hora del sistema del dispositivo en el que se ejecuta el navegador; **Hora del server**: zona horaria establecida globalmente o para el usuario de Zabbix; **Hora del host**: hora obtenida de un item del host.
Item	Seleccione el item con la clave de item `system.localtime[local]` para mostrar la hora del host. Este parámetro está disponible si *Tipo de hora* está establecido en ”Hora del host”.
Tipo de reloj	Seleccione el tipo de reloj (analógico o digital).
Anular zona horaria	Seleccione la zona horaria. Este parámetro está disponible para los tipos de hora *Hora local* y *Hora del server*.
Mostrar	Seleccione una o más unidades de información (fecha, hora, zona horaria) para mostrar en el reloj digital. El tamaño de la unidad se adapta automáticamente a las unidades mostradas, al tamaño del widget y a la escala de visualización. Este parámetro está disponible si *Tipo de reloj* está establecido en ”Digital”.

Configuración avanzada	Haga clic en el encabezado *Configuración avanzada* para mostrar las opciones de configuración avanzada del reloj digital. Esta sección está disponible si *Tipo de reloj* está establecido en "Digital".

Configuración avanzada

Las opciones de configuración avanzada están disponibles en la sección contraíble *Configuración avanzada*, y solo para aquellos elementos seleccionados en el campo *Mostrar* (consulte arriba).

Además, la configuración avanzada permite cambiar el color de fondo de todo el widget.

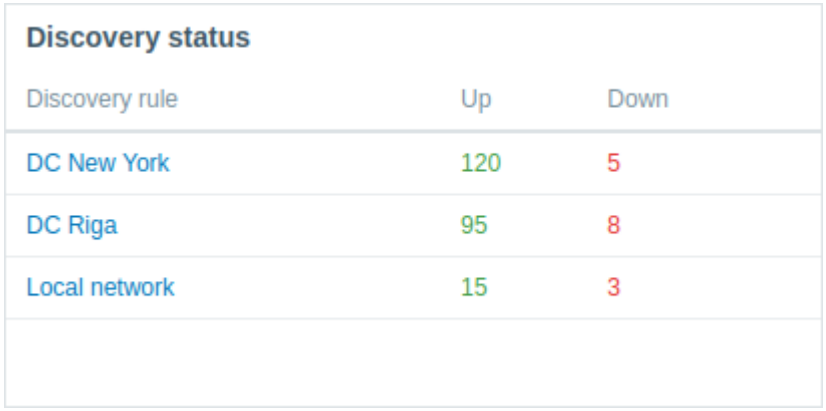


Color de fondo		Seleccione el color de fondo en el selector de colores. D representa el color predeterminado (depende del tema del frontend). Para volver al valor predeterminado, haga clic en el botón *Usar predeterminado* del selector de colores.
Fecha		Esta sección está disponible cuando la casilla *Fecha* está seleccionada para el parámetro *Mostrar*.
	Negrita	Marque la casilla para mostrar la fecha en negrita.
	Color	Seleccione el color de la fecha en el selector de colores. D representa el color predeterminado (depende del tema del frontend). Para volver al valor predeterminado, haga clic en el botón *Usar predeterminado* del selector de colores.
Hora		Esta sección está disponible cuando la casilla *Hora* está seleccionada para el parámetro *Mostrar*.
	Negrita	Marque la casilla para mostrar la hora en negrita.
	Color	Seleccione el color de la hora en el selector de colores. D representa el color predeterminado (depende del tema del frontend). Para volver al valor predeterminado, haga clic en el botón *Usar predeterminado* del selector de colores.
	Segundos	Marque la casilla para mostrar los segundos. De lo contrario, solo se mostrarán las horas y los minutos.
	Formato	Seleccione si desea mostrar la hora en formato de 24 o de 12 horas.
Zona horaria		Esta sección está disponible cuando la casilla *Zona horaria* está seleccionada para el parámetro *Mostrar*.
	Negrita	Marque la casilla para mostrar la zona horaria en negrita.
	Color	Seleccione el color de la zona horaria en el selector de colores. D representa el color predeterminado (depende del tema del frontend). Para volver al valor predeterminado, haga clic en el botón *Usar predeterminado* del selector de colores.
	Formato	Seleccione si desea mostrar la zona horaria en formato abreviado (por ejemplo, `New York`) o en formato completo (por ejemplo, `(UTC-04:00) America/New York`).

Estado de descubrimiento

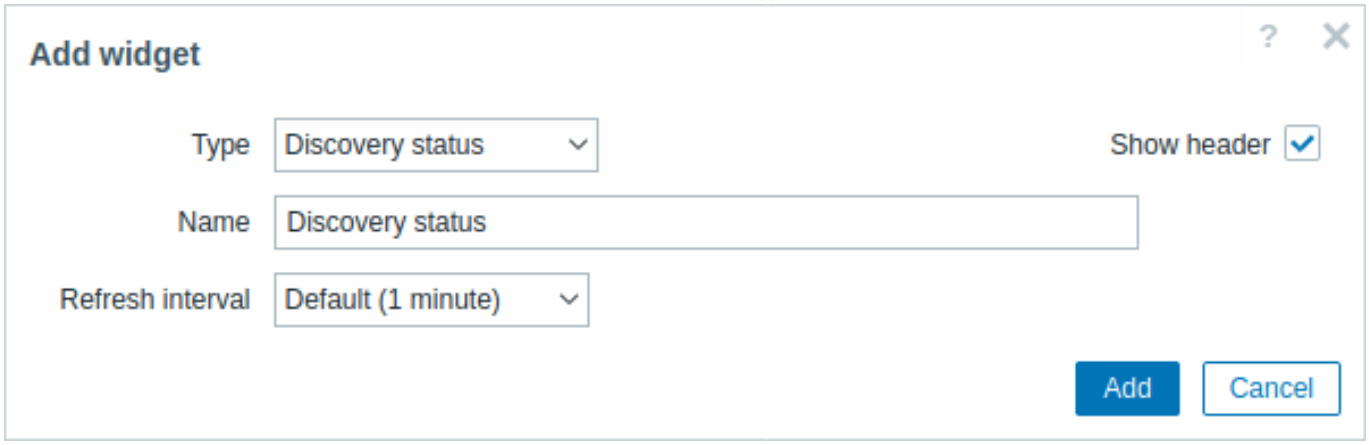
Descripción general

El widget *Estado de descubrimiento* muestra el estado de los dispositivos descubiertos por las reglas de descubrimiento de red habilitadas. Ayuda a supervisar si el descubrimiento de red está funcionando correctamente y el número de dispositivos que están activos o inactivos.



Configuración

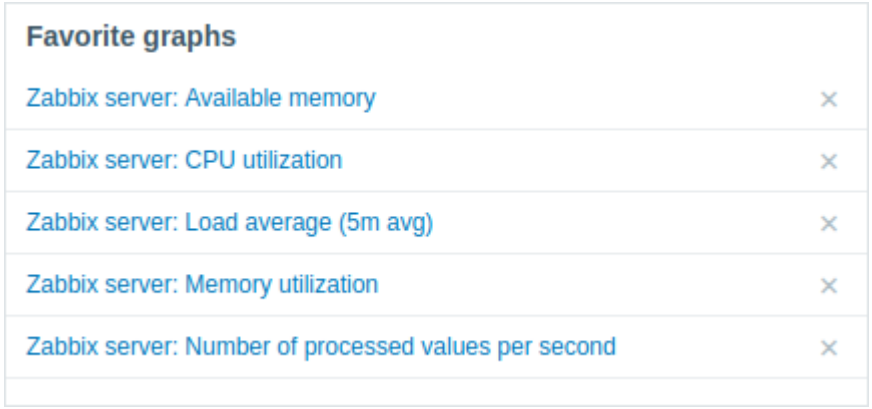
Todos los parámetros de configuración de este widget son comunes a todos los widgets.



Gráficos favoritos

Descripción general

El widget *Gráficas favoritas* muestra una lista alfabética de accesos directos a las gráficas marcadas como favoritas por el usuario actual.



Al hacer clic en x se elimina la gráfica de la lista y de favoritos.

Las gráficas se marcan como favoritas haciendo clic en el botón  *Añadir a favoritos* al visualizar gráficas de ítems en *Supervisión* > *Últimos datos*.



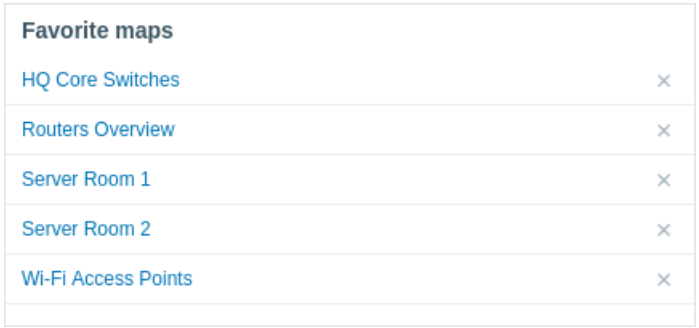
Configuración

Todos los parámetros de configuración de este widget son comunes a todos los widgets.

Mapas favoritos

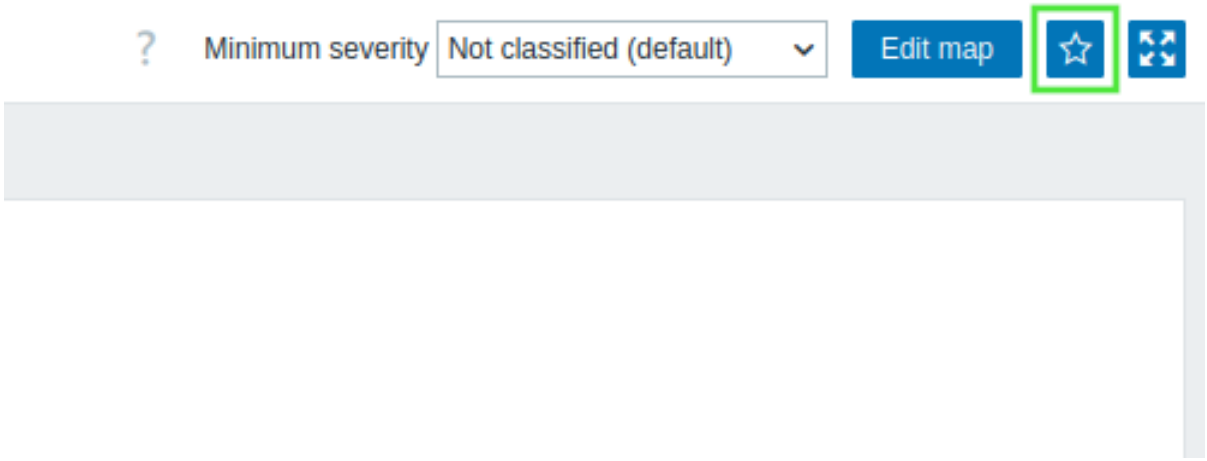
Descripción general

El widget *Mapas favoritos* muestra una lista de accesos directos a los mapas marcados como favoritos por el usuario actual, ordenados alfabéticamente.



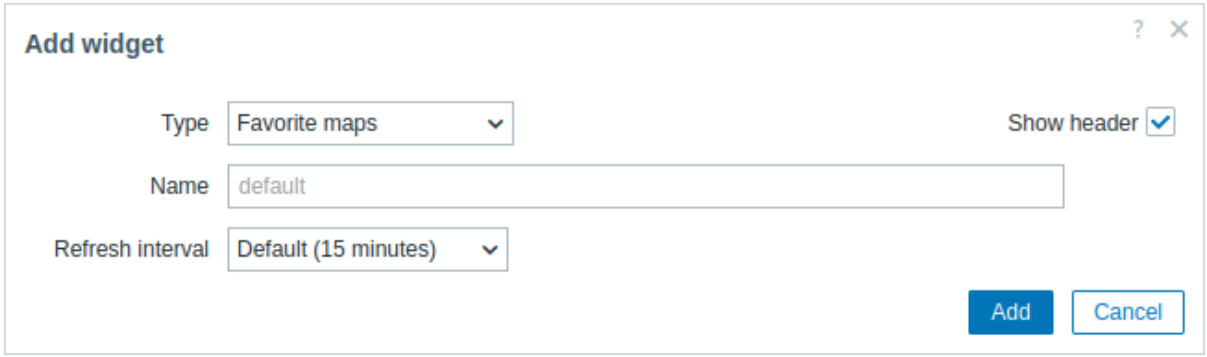
Al hacer clic en x se elimina el mapa de favoritos.

Los mapas se marcan como favoritos haciendo clic en el botón  *Agregar a favoritos* al visualizar mapas en *Supervisión* > *Mapas*.



Configuración

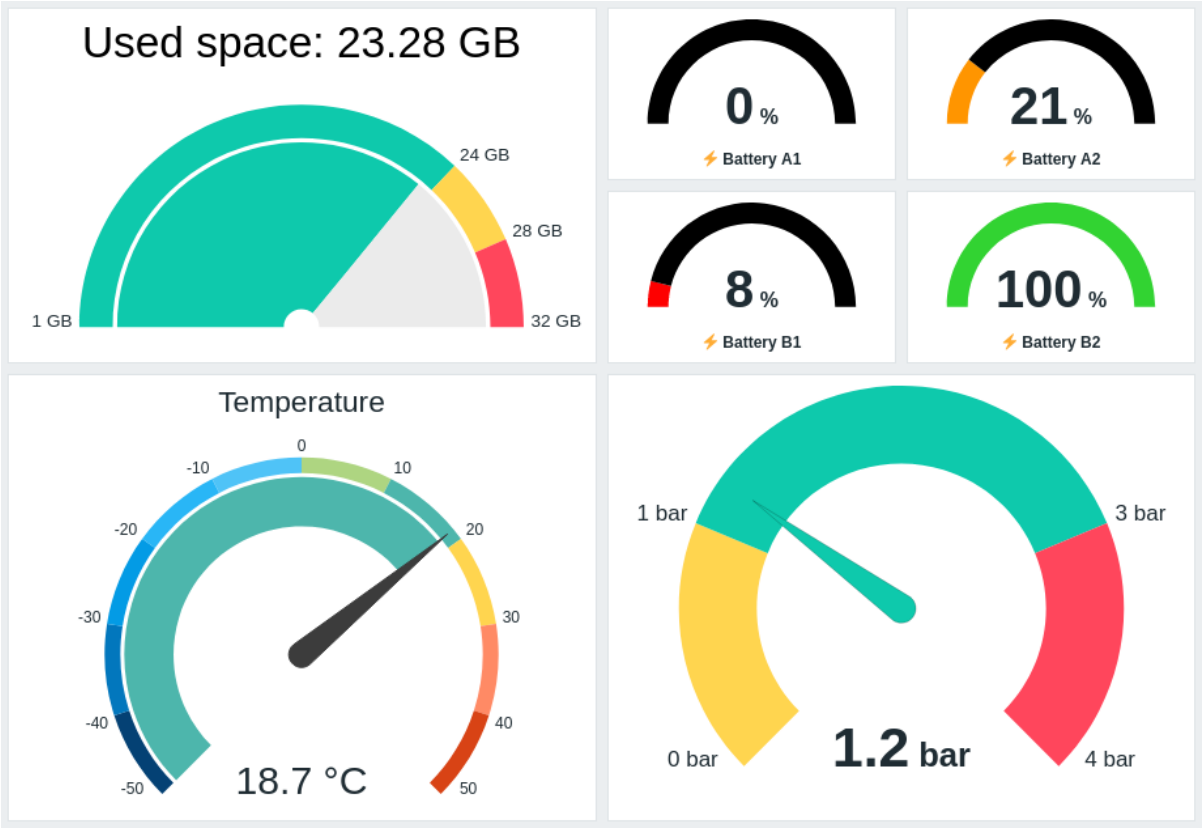
Todos los parámetros de configuración de este widget son comunes a todos los widgets.



Indicador

Descripción general

El widget *Gauge* muestra el valor de un único item numérico como un indicador. Es útil para supervisar métricas importantes, visualizar umbrales y detectar cambios repentinos.



Puede configurar el widget para que muestre lo siguiente:

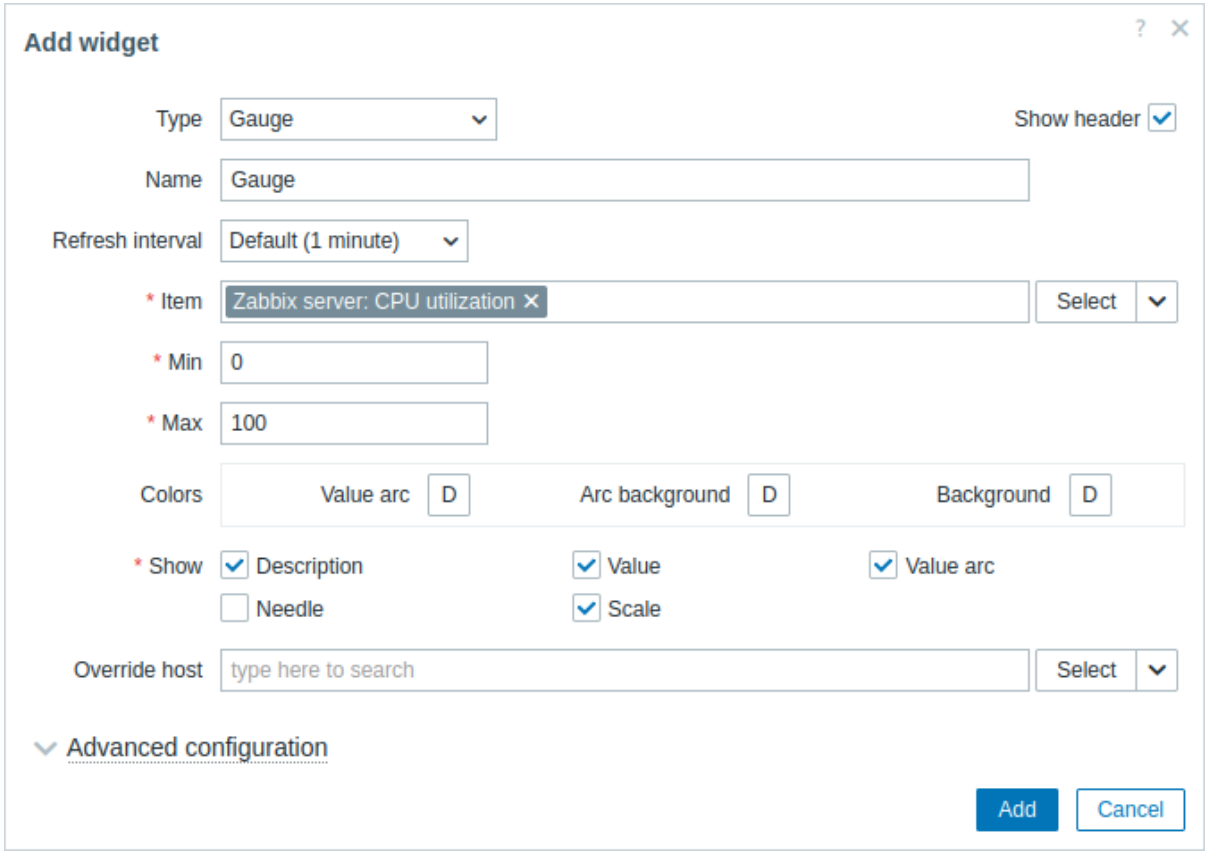
- Descripción del item (Espacio utilizado: {ITEM.LASTVALUE}, Temperatura)
- Valor y unidades del item (21 %, 18.7 °C, 1.2 bar)
- Escala del indicador (1GB/24GB/28GB/32GB, -50/-40/-30/etc.)
- Arco del indicador (arco del valor del indicador y arco de los umbrales del indicador)
- Aguja del indicador

Al hacer clic en el widget, se abre un gráfico simple para el item.

La información que se muestra en el widget *Gauge* se puede descargar como una imagen PNG seleccionando la opción *Descargar imagen* en el menú *Acciones* del widget ⋮, que aparece al pasar el cursor sobre el widget. Según la configuración del navegador, la imagen se guarda automáticamente en la carpeta `Downloads` o se abre un cuadro de diálogo *Guardar* que permite elegir la ubicación.

Configuración

Para configurar, seleccione *Gauge* como tipo:

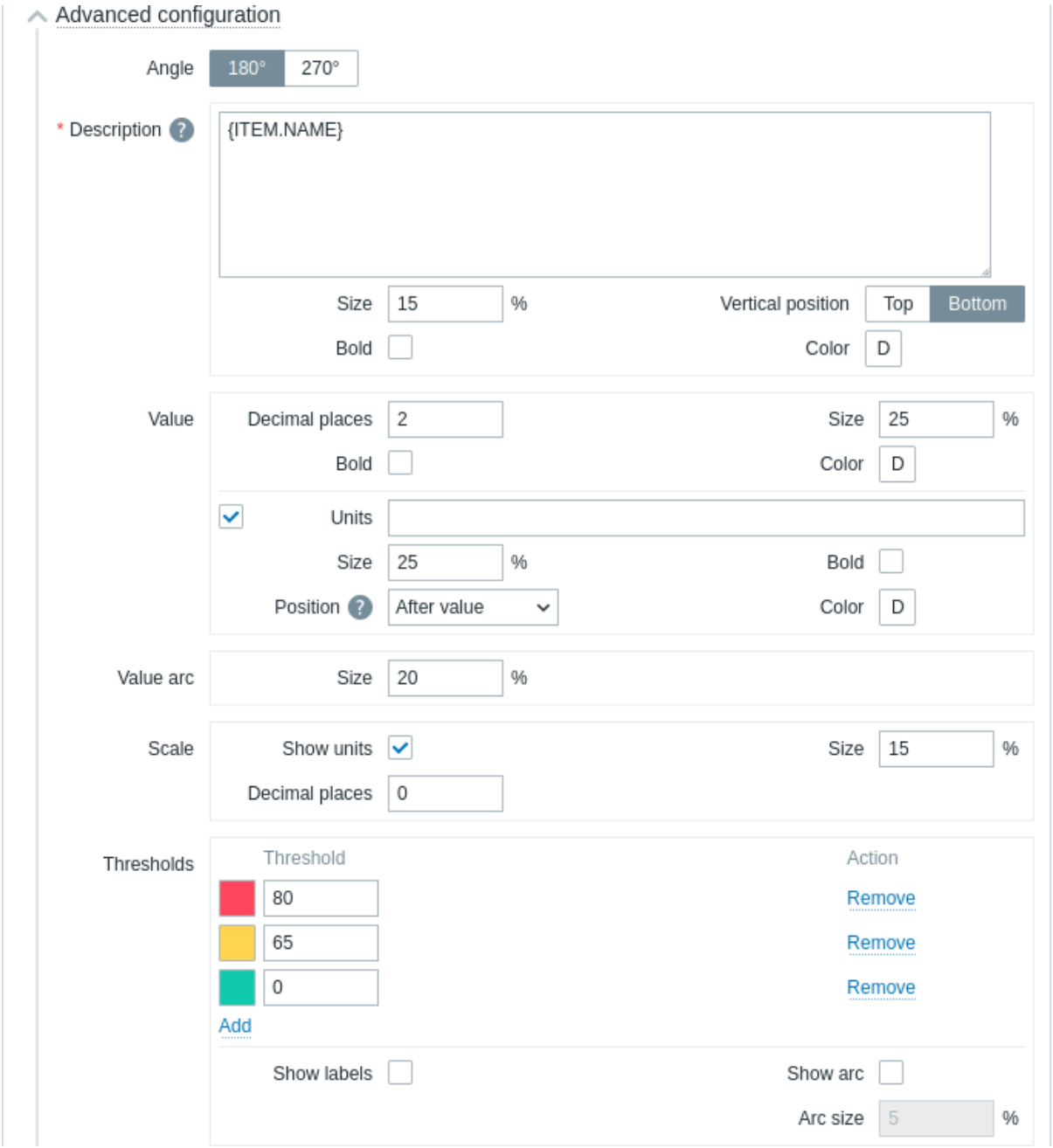


Además de los parámetros comunes para todos los widgets, puede establecer las siguientes opciones específicas:

Item	Seleccione el item. Como alternativa, seleccione un widget compatible como origen de datos para items. Este campo tiene autocompletado, por lo que al comenzar a escribir el nombre de un item se ofrecerá una lista desplegable con los items coincidentes. Tenga en cuenta que solo puede seleccionar items que devuelvan datos numéricos (excepto datos binarios).
Min	Introduzca el valor mínimo del gauge. Se admiten sufijos (por ejemplo, ”1d”, ”2w”, ”4K”, ”8G”). Se admiten mapeos de valores.
Max	Introduzca el valor máximo del gauge. Se admiten sufijos (por ejemplo, ”1d”, ”2w”, ”4K”, ”8G”). Se admiten mapeos de valores.
Colors	Seleccione el color en el selector de color: **Value arc** - seleccione el color del arco de valor del gauge; **Arc background** - seleccione el color de fondo del arco de valor del gauge y del arco de umbrales del gauge; **Background** - seleccione el color de fondo del widget. ”D” representa el color predeterminado, que depende del tema del frontend. Si se establecen *Thresholds*, el color predeterminado de **Value arc** depende del color del umbral. Para volver al color predeterminado, haga clic en el botón *Use default* del selector de color.
Show	Marque la casilla para mostrar el elemento correspondiente del gauge: descripción, valor, arco de valor, aguja, escala (el valor mínimo y máximo del gauge al principio y al final del arco del gauge). Desmárquela para ocultarlo. Debe seleccionarse al menos un elemento. Tenga en cuenta que la aguja y la escala del gauge pueden mostrarse si se muestra el arco de valor del gauge o el arco de umbrales del gauge (consulte las opciones de configuración avanzada). Tenga en cuenta también que, si se muestra la aguja del gauge, el valor se coloca debajo de la aguja; si la aguja está oculta, el valor se alinea con la parte inferior del arco del gauge.
Override host	Seleccione un widget compatible o el selector de host del dashboard como origen de datos para hosts. Este parámetro no está disponible al configurar el widget en un dashboard de template.
Advanced configuration	Haga clic en el encabezado *Advanced configuration* para mostrar las opciones de configuración avanzada. Aquí también podrá ajustar los elementos del gauge seleccionados en el campo *Show*.

Configuración avanzada

Las opciones de configuración avanzada están disponibles en la sección plegable *Configuración avanzada*:



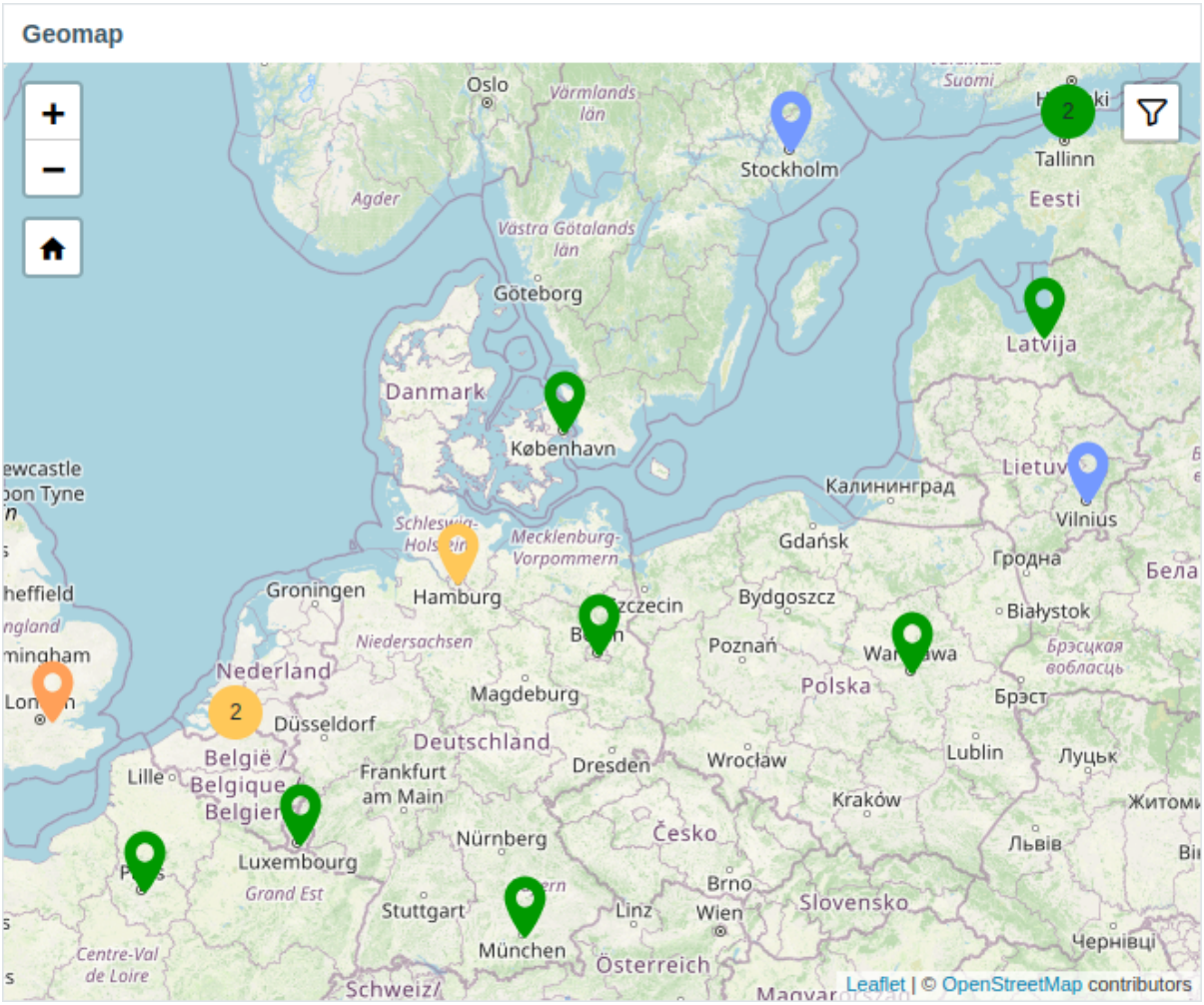
Ángulo	Seleccione el ángulo del indicador (180° o 270°).
Descripción	
Descripción	Introduzca la descripción de la métrica. Esta descripción puede sobrescribir el nombre por defecto de la métrica. Se admiten descripciones multilínea. Es posible una combinación de texto y macros soportadas. Se admiten {HOST.*}, {ITEM.*}, {INVENTORY.*} y macros de usuario.
Tamaño	Introduzca el tamaño de la fuente para la descripción de la métrica (en porcentaje, relativo a la altura total del widget).
Posición vertical	Seleccione la posición vertical de la descripción de la métrica (arriba o abajo, relativa al arco del indicador).
Negrita	Marque la casilla para mostrar la descripción de la métrica en negrita.
Color	Seleccione el color de la descripción de la métrica desde el selector de color. ”D” significa el color por defecto, que depende del tema de la interfaz. Para volver al color por defecto, haga clic en el botón *Usar por defecto* en el selector de color.
Valor	
Decimales	Introduzca el número de decimales a mostrar con el valor. Esta opción solo afecta a las métricas que devuelven datos numéricos (float).

Tamaño	Introduzca el tamaño de la fuente para el valor (en porcentaje, relativo a la altura del arco del indicador).
Negrita	Marque la casilla para mostrar el valor en negrita.
Color	Seleccione el color del valor desde el selector de color. ”D” significa el color por defecto, que depende del tema de la interfaz. Para volver al color por defecto, haga clic en el botón *Usar por defecto* en el selector de color.
Unidades	
Unidades	Marque la casilla para mostrar las unidades junto al valor de la métrica. Si introduce un nombre de unidad, sobrescribirá las unidades establecidas en la configuración de la métrica.
Tamaño	Introduzca el tamaño de la fuente para las unidades de la métrica (en porcentaje, relativo a la altura del arco del indicador).
Negrita	Marque la casilla para mostrar las unidades de la métrica en negrita.
Posición	Seleccione la posición de las unidades de la métrica (arriba, abajo, antes o después, relativo al valor de la métrica). Esta opción se ignora para las siguientes unidades relacionadas con el tiempo: unixtime, uptime, s.
Color	Seleccione el color de las unidades de la métrica desde el selector de color. ”D” significa el color por defecto, que depende del tema de la interfaz. Para volver al color por defecto, haga clic en el botón *Usar por defecto* en el selector de color.
Arco de valor	
Tamaño del arco	Introduzca la altura del arco de valor del indicador (en porcentaje, relativo al radio del arco del indicador).
Aguja	
Color	Seleccione el color de la aguja del indicador desde el selector de color. ”D” significa el color por defecto, que depende del tema de la interfaz. Si se establecen *Umbrales*, el color por defecto de la aguja depende del color del umbral. Para volver al color por defecto, haga clic en el botón *Usar por defecto* en el selector de color.
Escala	
Mostrar unidades	Marque la casilla para mostrar las unidades junto al valor mínimo y máximo del indicador.
Tamaño	Introduzca el tamaño de la fuente para el valor mínimo y máximo del indicador (en porcentaje, relativo a la altura del arco del indicador).
Decimales	Introduzca el número de decimales a mostrar con el valor mínimo y máximo del indicador. Esta opción solo afecta a las métricas que devuelven datos numéricos (float).
Umbrales	
Umbrales	Haga clic en *Añadir* para agregar un umbral, seleccione un color de umbral desde el selector de color y especifique un valor numérico. La lista de umbrales se ordenará de forma ascendente al guardar. Tenga en cuenta que los colores configurados como umbrales solo se mostrarán correctamente para métricas numéricas. Se admiten sufijos (por ejemplo, ”1d”, ”2w”, ”4K”, ”8G”). Se admiten mapeos de valores.
Mostrar etiquetas	Marque la casilla para mostrar los valores de umbral como etiquetas en la escala del indicador.
Mostrar arco	Marque la casilla para mostrar el arco de umbrales del indicador.
Tamaño del arco	Introduzca la altura del arco de umbrales del indicador (en porcentaje, relativo al radio del arco del indicador).

Geomap

Descripción general

El widget *Geomap* muestra los hosts como marcadores en un mapa interactivo basado en Leaflet, con colores de marcador que indican el problema más grave de cada host.

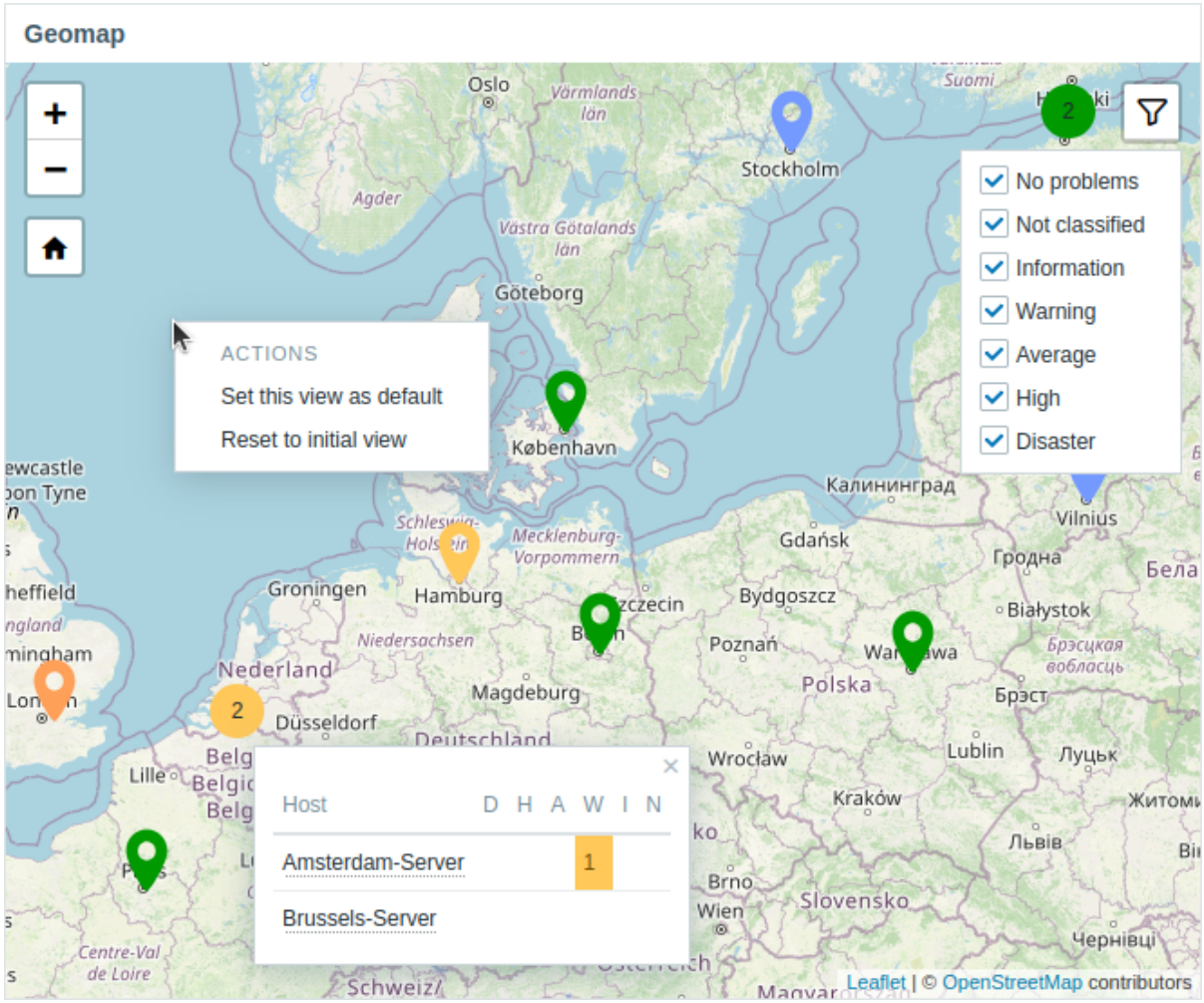


Note:
Zabbix ofrece varios proveedores de mapas predefinidos y permite añadir proveedores personalizados en *Administración > General > Mapas geográficos*.

Por defecto, el widget muestra todos los hosts habilitados que tienen coordenadas válidas (latitud -90 a 90, longitud -180 a 180) en su inventario.

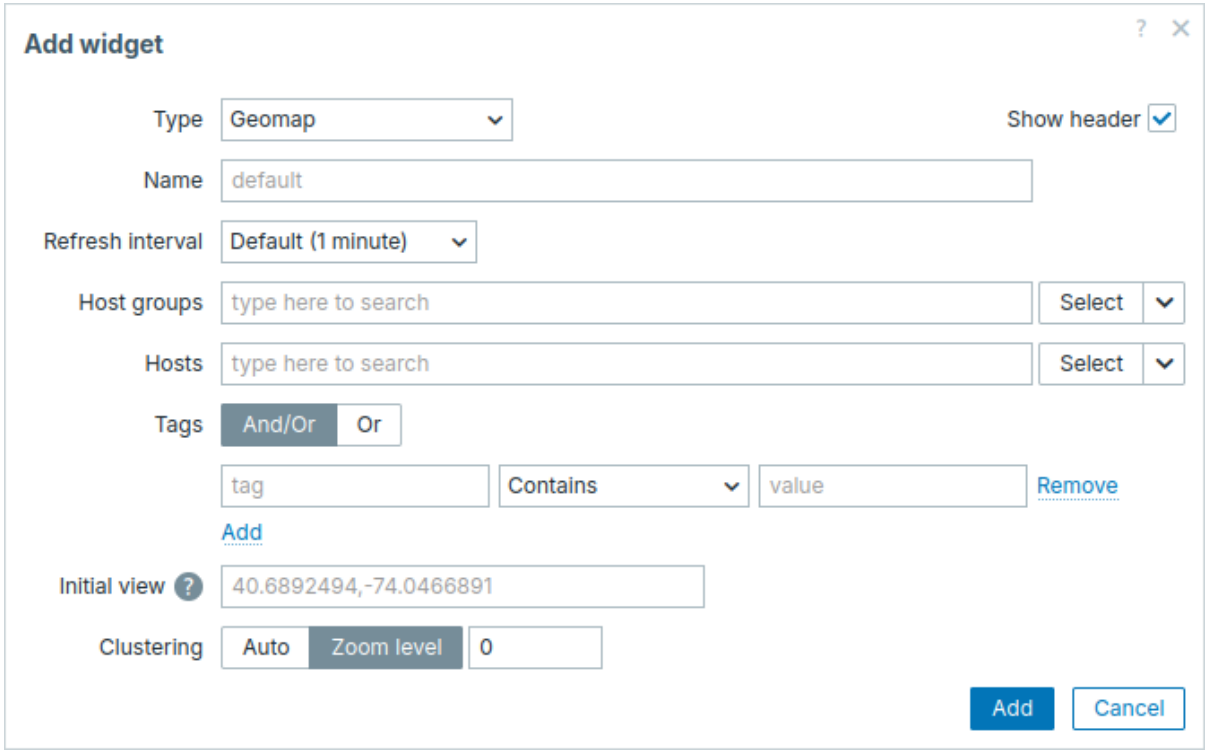
Utilice las siguientes opciones para interactuar con el widget:

- Al hacer clic en un marcador se muestra el nombre del host y los problemas no resueltos; al hacer clic en el nombre del host se abre el menú del host.
- Filtre los hosts por gravedad del problema utilizando el icono de filtro en la esquina superior derecha.
- Haga clic con el botón derecho en el mapa para establecer o restablecer la vista predeterminada; utilice el icono de inicio para volver a la vista predeterminada.



Configuración

Para añadir el widget, seleccione *Geomap* como tipo.



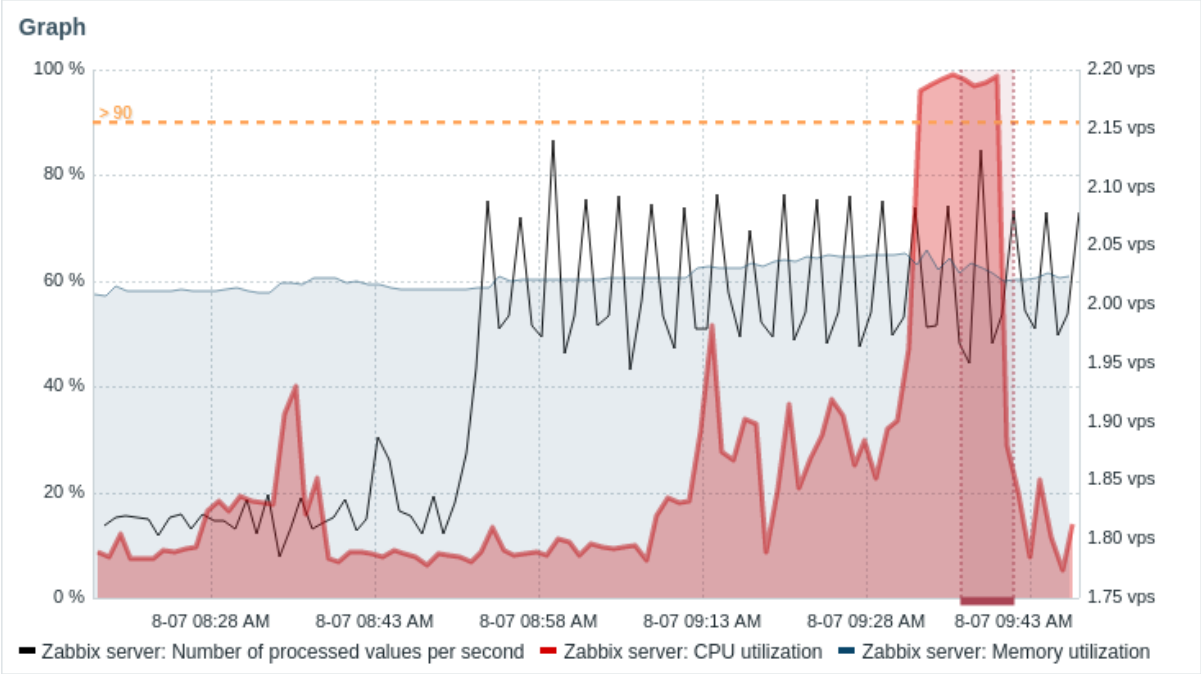
Además de los parámetros comunes para todos los widgets, puede establecer las siguientes opciones específicas:

Grupos de hosts	Seleccione los grupos de hosts que se mostrarán en el mapa. Como alternativa, seleccione un widget compatible como origen de datos para los grupos de hosts. Este campo tiene autocompletado, por lo que al comenzar a escribir el nombre de un grupo se ofrecerá una lista desplegable de grupos coincidentes. Si no se selecciona nada en los campos *Grupos de hosts* y *Hosts*, se mostrarán todos los hosts con coordenadas válidas. Este parámetro no está disponible al configurar el widget en un dashboard de template.
Hosts	Seleccione los hosts que se mostrarán en el mapa. Como alternativa, seleccione un widget compatible o el dashboard como origen de datos para los hosts. Este campo tiene autocompletado, por lo que al comenzar a escribir el nombre de un host se ofrecerá una lista desplegable de hosts coincidentes. Si no se selecciona nada en los campos *Grupos de hosts* y *Hosts*, se mostrarán todos los hosts con coordenadas válidas. Este parámetro no está disponible al configurar el widget en un dashboard de template.
Etiquetas	Especifique etiquetas para limitar el número de hosts mostrados en el widget. Es posible incluir y también excluir etiquetas y valores de etiqueta específicos. Se pueden establecer varias condiciones. La coincidencia del nombre de la etiqueta siempre distingue entre mayúsculas y minúsculas. Hay varios operadores disponibles para cada condición: **Existe** - incluir los nombres de etiqueta especificados; **Igual a** - incluir los nombres de etiqueta y valores especificados (distingue entre mayúsculas y minúsculas); **Contiene** - incluir los nombres de etiqueta especificados cuando los valores de etiqueta contengan la cadena introducida (coincidencia de subcadena, sin distinguir entre mayúsculas y minúsculas); **No existe** - excluir los nombres de etiqueta especificados; **No es igual a** - excluir los nombres de etiqueta y valores especificados (distingue entre mayúsculas y minúsculas); **No contiene** - excluir los nombres de etiqueta especificados cuando los valores de etiqueta contengan la cadena introducida (coincidencia de subcadena, sin distinguir entre mayúsculas y minúsculas). Hay dos tipos de cálculo para las condiciones: **And/Or** - deben cumplirse todas las condiciones; las condiciones que tengan el mismo nombre de etiqueta se agruparán mediante la condición *Or*; **Or** - basta con que se cumpla una condición. Este parámetro no está disponible al configurar el widget en un dashboard de template.
Vista inicial	Coordenadas del centro separadas por comas y un nivel de zoom opcional que se mostrará cuando el widget se cargue inicialmente, en el formato `<latitude>,<longitude>,<zoom>` Si se especifica el zoom inicial, el widget Geomap se carga con el nivel de zoom indicado. De lo contrario, el zoom inicial se calcula como la mitad del zoom máximo para el proveedor de teselas correspondiente. La vista inicial se ignora si está establecida la vista predeterminada (véase más abajo). Ejemplos: 40.6892494,-74.0466891 40.6892494,-122.0466891
Agrupación	Especifique cómo se combinan los marcadores de hosts cercanos en un único marcador con un contador: **Auto** - los marcadores se combinan automáticamente; **Nivel de zoom** - los marcadores se combinan en función del nivel de zoom especificado (0 - alejamiento máximo, 30 - acercamiento máximo). Por ejemplo, si el parámetro se establece en 0, los marcadores no se combinan en ningún nivel de zoom; si se establece en 15, los marcadores se combinan desde el alejamiento máximo hasta el nivel de zoom 15; si se establece en 30, los marcadores se combinan en todos los niveles de zoom. Tenga en cuenta que el nivel máximo de acercamiento lo determina el proveedor de teselas del mapa.

Gráfico

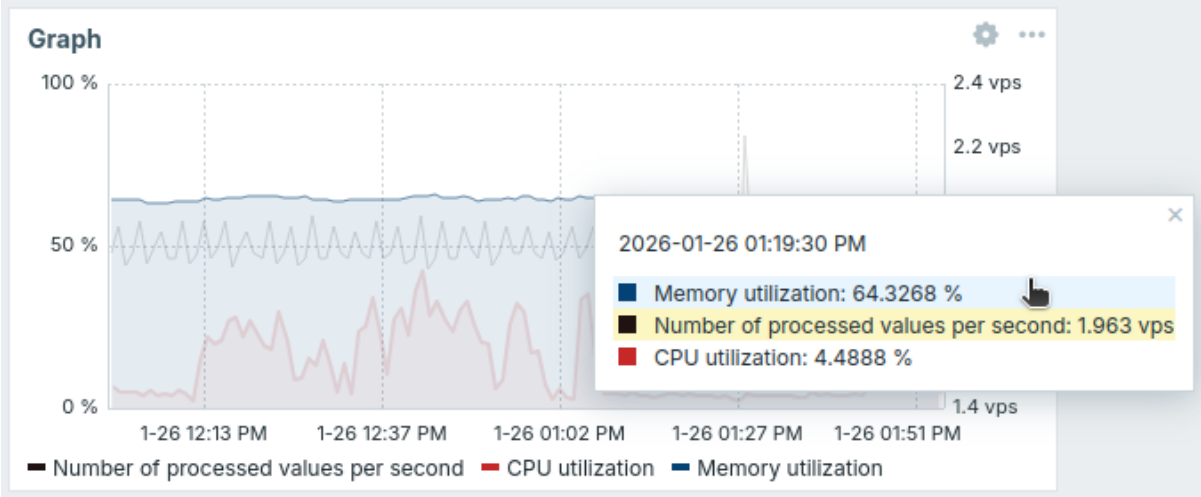
Descripción general

El widget *Graph* muestra los datos numéricos de los item en forma de gráfico basado en vectores. Puede ayudarle a realizar un seguimiento de las tendencias, detectar problemas y comparar valores a lo largo del tiempo y entre hosts.

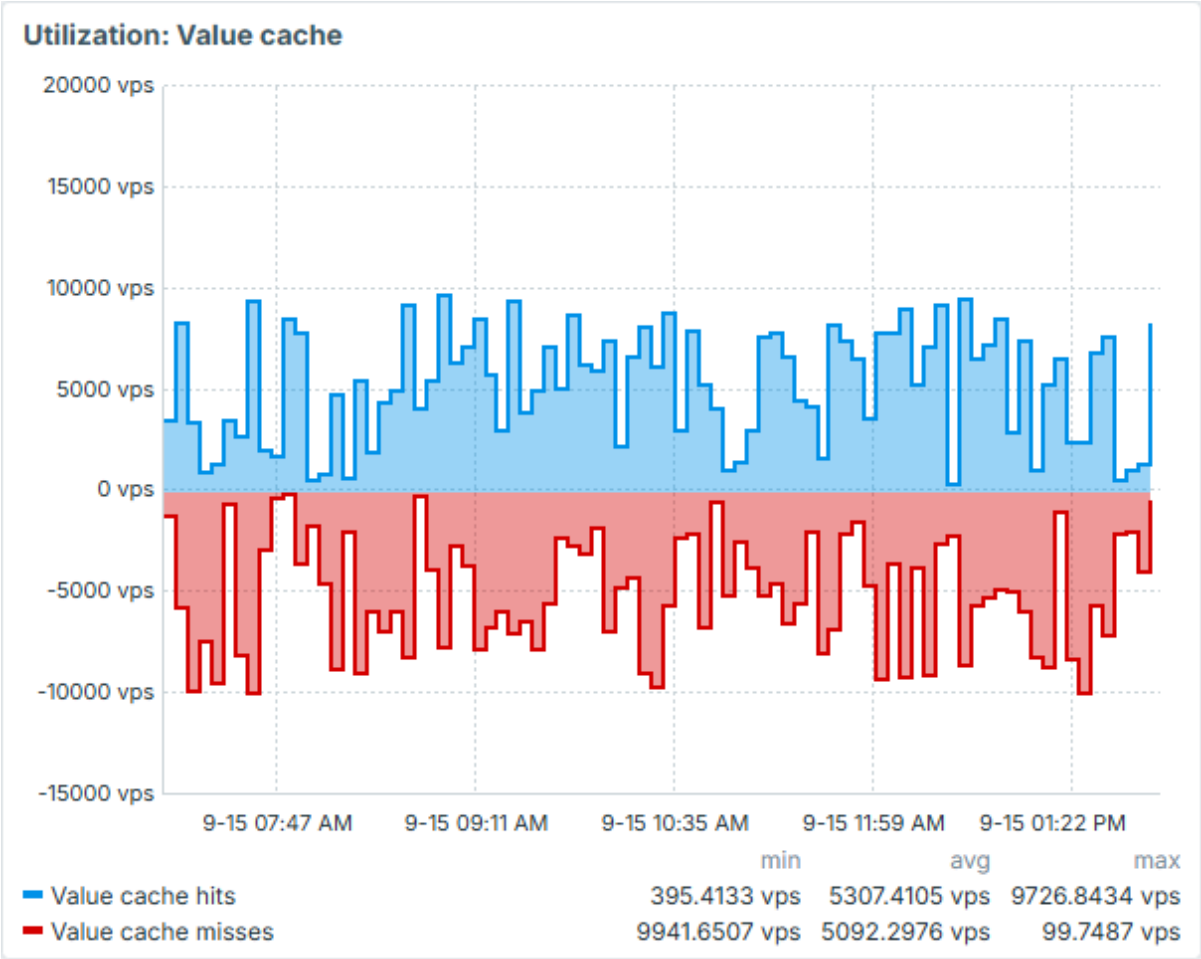


Al pasar el cursor sobre el gráfico, se muestra una información emergente con una lista de item ordenados por valor en orden descendente.

Al pasar el cursor sobre un item en la información emergente, se resalta su gráfico (los demás se atenúan); al seleccionar el item, sus datos se transmiten a otros widgets.



El gráfico se puede mostrar con los valores del eje Y invertidos; esto solo afecta a la representación visual y no modifica los valores originales.

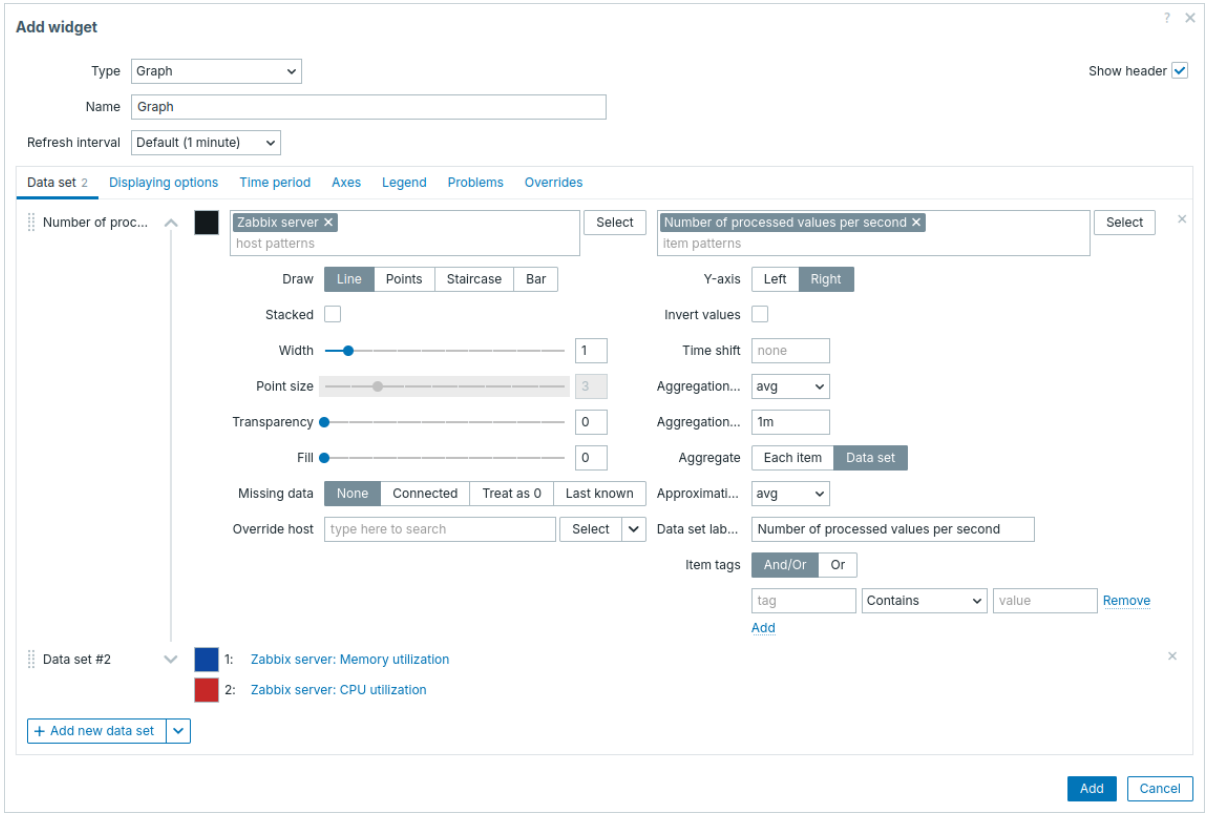


También está disponible el widget heredado Graph (classic).

La información mostrada en el widget *Graph* se puede descargar como una imagen PNG seleccionando la opción *Download image* en el menú *Actions* del widget ⋯, que aparece al pasar el cursor sobre el widget. Según la configuración del navegador, la imagen se guarda automáticamente en la carpeta `Downloads` o se abre un cuadro de diálogo *Save* que permite elegir la ubicación.

Configuración

Para configurar, seleccione *Gráfico* como tipo:



Conjunto de datos

La pestaña **Conjunto de datos** permite seleccionar los datos para el gráfico mediante la adición de conjuntos de datos. Se pueden añadir dos tipos de conjuntos de datos:

- *Patrones de item*: se muestran los datos de los items coincidentes. Puede elegir un único color base o seleccionar una fila de la paleta para asignar colores distintos a cada item coincidente.
- *Lista de items*: se muestran los datos de los items seleccionados. Puede elegir individualmente el color de cada item mediante el selector.

De forma predeterminada, se añade un conjunto de datos de tipo *Patrones de item*.

Conjunto de datos	Para el conjunto de datos **Patrones de item**: Seleccione o introduzca patrones de host e item; los datos de los items que coincidan con estos patrones se mostrarán en el gráfico; se pueden mostrar hasta 50 items. Se pueden utilizar patrones comodín para la selección (por ejemplo, ∗ devolverá resultados que coincidan con cero o más caracteres). Para especificar un patrón comodín, introduzca la cadena manualmente y pulse *Enter*. El símbolo comodín siempre se interpreta, por lo que no es posible añadir individualmente, por ejemplo, un item llamado *item** si existen otros items coincidentes (por ejemplo, *item2*, *item3*). Especificar patrones de host e item es obligatorio para los conjuntos de datos de tipo ”Patrones de item”. Consulte también: Detalles de configuración del conjunto de datos. Para el conjunto de datos **Lista de items**: Seleccione los items para el gráfico haciendo clic en el botón *Añadir item*. También puede seleccionar widgets compatibles como fuente de datos para los items haciendo clic en el botón *Añadir widget*. Especificar items o widgets es obligatorio para los conjuntos de datos de tipo ”Lista de items”. Consulte también: Detalles de configuración del conjunto de datos. Tenga en cuenta que solo se permiten tipos de item numéricos. Al configurar el widget en un dashboard de template, el parámetro para especificar patrones de host no está disponible y el parámetro para especificar una lista de items solo permite seleccionar items configurados en el template.

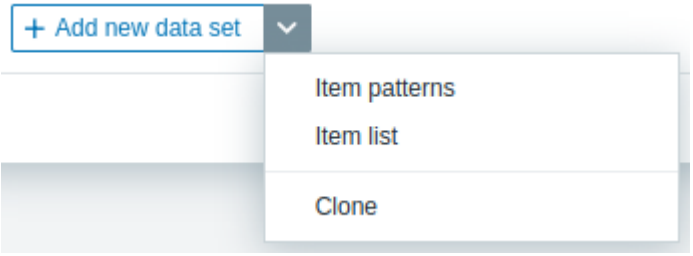
Dibujar	Elija el tipo de dibujo de la métrica. Tipos de dibujo posibles: *Línea* (establecido de forma predeterminada), *Puntos*, *Escalera* y *Barras*. Tenga en cuenta que, si solo hay un punto de datos en el gráfico de línea/escalera, se dibuja como un punto independientemente del tipo de dibujo. El tamaño del punto se calcula a partir del ancho de la línea, pero no puede ser inferior a 3 píxeles, aunque el ancho de la línea sea menor.
Apilado	Marque la casilla para mostrar los datos apilados (se muestran áreas rellenas). Esta opción está deshabilitada cuando se selecciona el tipo de dibujo *Puntos*.
Ancho	Establezca el ancho de la línea. Esta opción está disponible cuando se selecciona el tipo de dibujo *Línea* o *Escalera*.
Tamaño del punto	Establezca el tamaño del punto. Esta opción está disponible cuando se selecciona el tipo de dibujo *Puntos*.
Transparencia	Establezca el nivel de transparencia.
Relleno	Establezca el nivel de relleno. Esta opción está disponible cuando se selecciona el tipo de dibujo *Línea* o *Escalera*.
Datos faltantes	Seleccione la opción para mostrar los datos faltantes: **Ninguno**: el espacio se deja vacío; **Conectado**: se conectan los dos valores de los extremos; **Tratar como 0**: los datos faltantes se muestran como valores 0; **Último conocido**: los datos faltantes se muestran con el mismo valor que el último valor conocido; no se aplica a los tipos de dibujo *Puntos* y *Barras*.
Anular host	Seleccione un widget compatible o el selector de host del dashboard como fuente de datos para los hosts. Este parámetro no está disponible al configurar el widget en un dashboard de template.
Eje Y	Seleccione el lado del gráfico donde se mostrará el eje Y.
Invertir valores	Marque la casilla para multiplicar todos los valores del eje Y por -1 antes de mostrarlos en el gráfico. Esto solo afecta a la representación visual; los datos originales no se modifican. Esta opción no afecta a la información emergente, la leyenda ni la línea de percentil. Si se selecciona la opción *Triggers simples* para el eje, los umbrales de los triggers se mostrarán invertidos.
Desplazamiento temporal	Especifique el desplazamiento temporal si es necesario. Puede utilizar sufijos de tiempo en este campo. Se permiten valores negativos.
Función de agregación	Especifique qué función de agregación se utilizará para cada item o para un conjunto de datos completo dentro del *Intervalo de agregación* seleccionado: **min**: muestra el valor más pequeño; **max**: muestra el valor más grande; **avg**: muestra el valor promedio; **sum**: muestra la suma de los valores; **count**: muestra el número de valores; **first**: muestra el primer valor; **last**: muestra el último valor. Si se selecciona **no utilizada**, se muestran todos los valores (sin agregación). Solo se pueden mostrar datos numéricos para *min*, *max*, *avg* y *sum*. Para *count*, los datos no numéricos se convertirán a numéricos. Consulte también: Agregación en gráficos.
Intervalo de agregación	Especifique el intervalo para agregar los valores. Puede utilizar sufijos de tiempo en este campo. Un valor numérico sin sufijo se considerará en segundos. Tenga en cuenta que, si el widget está configurado para mostrar datos históricos basados en tendencias (*Selección de datos históricos* está establecido en *Tendencias* o *Automático*), se recomienda utilizar un intervalo de agregación que sea múltiplo de 1 hora (por ejemplo, 3600, 60m, 1h, 3h, etc.). Las tendencias almacenan valores agregados por hora, por lo que utilizar un intervalo de agregación que no sea múltiplo de 1 hora (por ejemplo, 100s, 7min, 15min, 90min, etc.) puede producir resultados difíciles de interpretar.
Agregar	Especifique si se deben agregar los datos: **Cada item**: cada item del conjunto de datos se agregará y mostrará por separado; **Conjunto de datos**: todos los items del conjunto de datos se agregarán y mostrarán como un único valor.

Aproximación	Especifique qué valor se mostrará cuando exista más de un valor por píxel vertical del gráfico: **all**: muestra los valores más pequeño, más grande y promedio; **min**: muestra el valor más pequeño; **max**: muestra el valor más grande; **avg**: muestra el valor promedio. Esta configuración resulta útil al mostrar un gráfico para un período de tiempo amplio con un intervalo de actualización frecuente (como un año de valores recopilados cada 10 minutos).
Etiqueta del conjunto de datos	Especifique la etiqueta del conjunto de datos que se muestra en la configuración del gráfico *Conjunto de datos* y en la *Leyenda* del gráfico (para conjuntos de datos agregados). Todos los conjuntos de datos se numeran, incluidos aquellos con una *Etiqueta del conjunto de datos* especificada. Si no se especifica ninguna etiqueta, el conjunto de datos se etiquetará automáticamente según su número (por ejemplo, ”Conjunto de datos n.º 2”, ”Conjunto de datos n.º 3”, etc.). La numeración de los conjuntos de datos se recalcula después de reordenarlos o arrastrarlos. Las etiquetas de conjuntos de datos demasiado largas se acortarán para ajustarse al espacio disponible (por ejemplo, ”Número de proc...”).
Etiquetas de item	Especifique las etiquetas para filtrar los items mostrados en el widget. Es posible incluir o excluir etiquetas y valores de etiqueta específicos. Se pueden establecer varias condiciones. La coincidencia de nombres de etiqueta siempre distingue entre mayúsculas y minúsculas. Hay varios operadores disponibles para cada condición: **Existe**: incluye los nombres de etiqueta especificados; **Es igual a**: incluye los nombres y valores de etiqueta especificados (distingue entre mayúsculas y minúsculas); **Contiene**: incluye los nombres de etiqueta especificados cuyos valores contienen la cadena introducida (coincidencia de subcadena, sin distinguir entre mayúsculas y minúsculas); **No existe**: excluye los nombres de etiqueta especificados; **No es igual a**: excluye los nombres y valores de etiqueta especificados (distingue entre mayúsculas y minúsculas); **No contiene**: excluye los nombres de etiqueta especificados cuyos valores contienen la cadena introducida (coincidencia de subcadena, sin distinguir entre mayúsculas y minúsculas). Hay dos tipos de cálculo para las condiciones: **Y/O**: deben cumplirse todas las condiciones; las condiciones que tienen el mismo nombre de etiqueta se agruparán mediante la condición O; **O**: basta con que se cumpla una condición.

Detalles de configuración del conjunto de datos

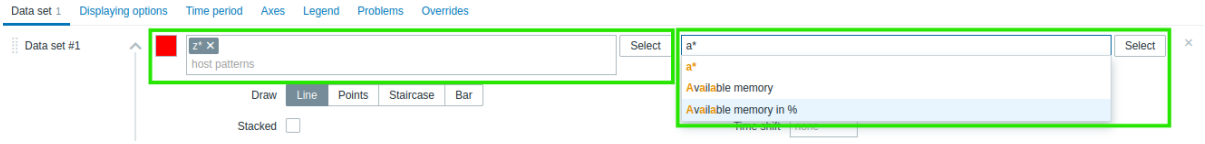
Los conjuntos de datos existentes se muestran en una lista. Puede:

- Hacer clic en el icono de mover y arrastrar un conjunto de datos a una nueva posición en la lista.
- Hacer clic en el icono de expandir para mostrar los detalles del conjunto de datos. Al expandirlo, este icono se convierte en un icono de contraer.
- Hacer clic en el icono de color para abrir el selector. Puede introducir un código hexadecimal, elegir una muestra de color sólido o cambiar a la pestaña *Paleta* y seleccionar una fila de colores predefinidos. El color elegido se aplica directamente a los conjuntos de datos de tipo *Lista de items* o se utiliza como base para generar tonalidades en *Patrones de items*. Utilice Tab para desplazarse entre los controles del cuadro de diálogo, las teclas de flecha para navegar por las muestras o las filas de la paleta, Enter para seleccionar y Esc para cancelar.
- Hacer clic en el botón *Añadir nuevo conjunto de datos* para añadir un conjunto de datos vacío que permita seleccionar patrones de host y de item. Si hace clic en el icono que apunta hacia abajo junto al botón *Añadir nuevo conjunto de datos*, aparece un menú desplegable que permite añadir un nuevo conjunto de datos de tipo *Patrones de items* o *Lista de items*, o *Clonar* el conjunto de datos abierto actualmente. Si todos los conjuntos de datos están contraídos, la opción *Clonar* no está disponible.

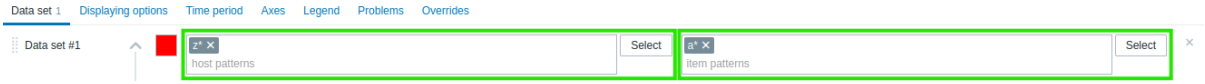


El conjunto de datos **Patrones de item** contiene los campos *Patrones de host* y *Patrones de item* que reconocen tanto nombres completos como patrones que contienen un símbolo comodín (*). Esta funcionalidad le permite seleccionar todos los nombres de host y nombres de item que contengan el patrón seleccionado. Al escribir el nombre del item o el patrón del item en el campo *Patrones de item*, sólo se muestran en la lista desplegable los items que pertenecen al/los nombre(s) de host seleccionado(s).

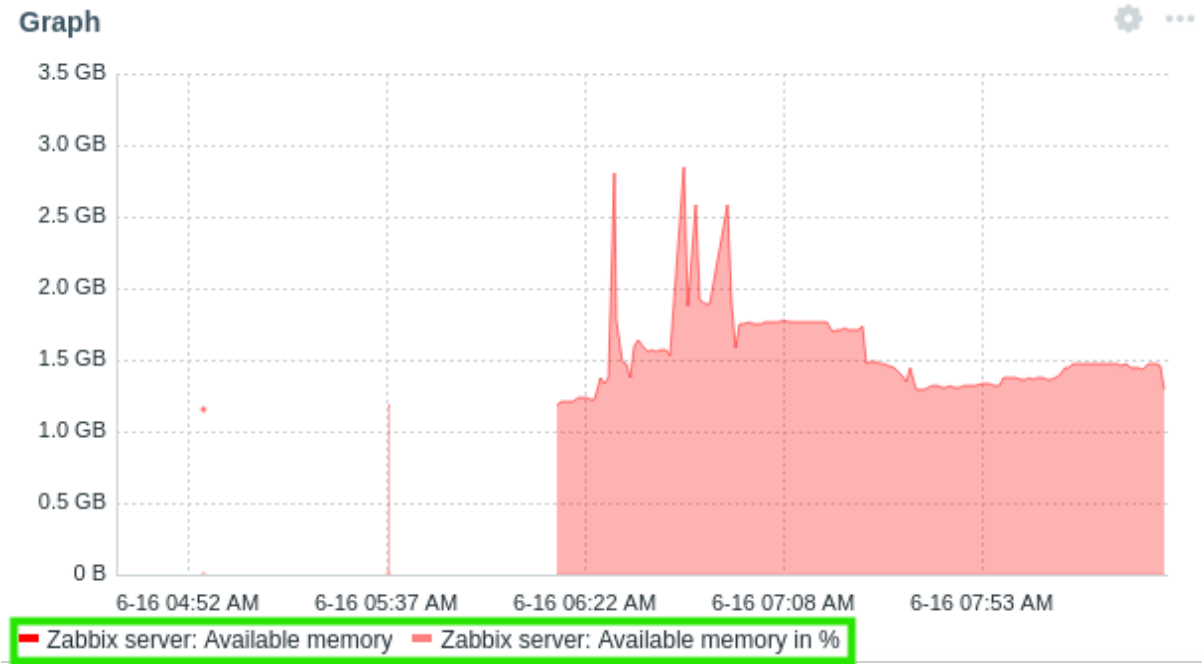
Por ejemplo, al escribir el patrón **z*** en el campo *Patrones de host*, la lista desplegable muestra todos los nombres de host que contienen este patrón: **z*, Zabbix server, Zabbix proxy**. Después de pulsar *Intro*, este patrón es aceptado y se muestra como **z***. Del mismo modo, al escribir el patrón **a*** en el campo *Patrones de item*, la lista desplegable muestra todos los nombres de item que contienen este patrón: **a*, Available memory, Available memory in %**.



Después de pulsar *Intro*, el patrón es aceptado y se muestra como **a***.



El gráfico muestra entonces todos los items que pertenecen al/los nombre(s) de host seleccionado(s).



El conjunto de datos de **Lista de elementos** contiene el botón *Agregar elemento* que le permite agregar elementos para que se muestren en el gráfico. También puede agregar widgets compatibles como fuente de datos para los elementos haciendo clic en el botón *Agregar widget*.

Por ejemplo, al hacer clic en el botón *Agregar elemento* se abre una ventana emergente que contiene un parámetro *Equipo*. Después de seleccionar un equipo, todos sus elementos disponibles para la selección se muestran en una lista.

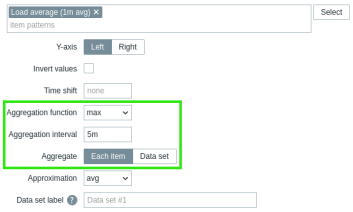
Después de seleccionar uno o más elementos, se mostrarán en la lista de elementos del conjunto de datos y en el gráfico.

Agregación en gráficos

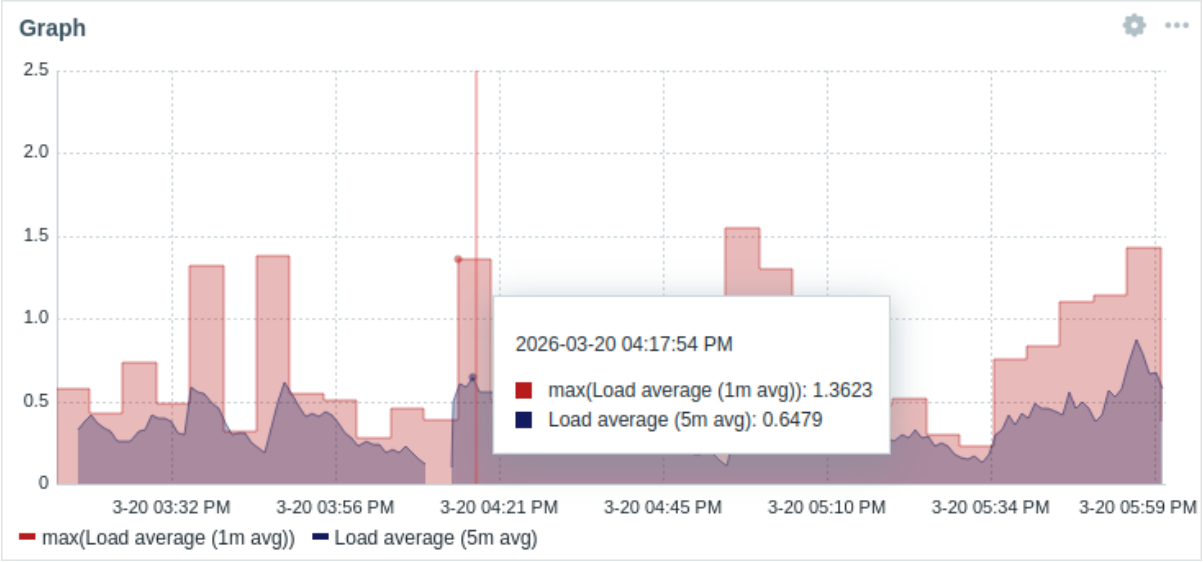
La agregación permite mostrar un valor agregado (mínimo, máximo, promedio, etc.) para el intervalo especificado (5 minutos, una hora, un día, etc.) en lugar de todos los valores individuales. Dentro de un conjunto de datos, la agregación se puede aplicar a cada item por separado o a todo el conjunto de datos.

Algunos casos de uso de la función de agregación pueden ser los siguientes: - El número promedio de solicitudes a un server NGINX por día. - El menor espacio en disco entre los clústeres por semana.

En el siguiente ejemplo, la función de agregación ”max” se aplica al item con un intervalo de agregación de cinco minutos:



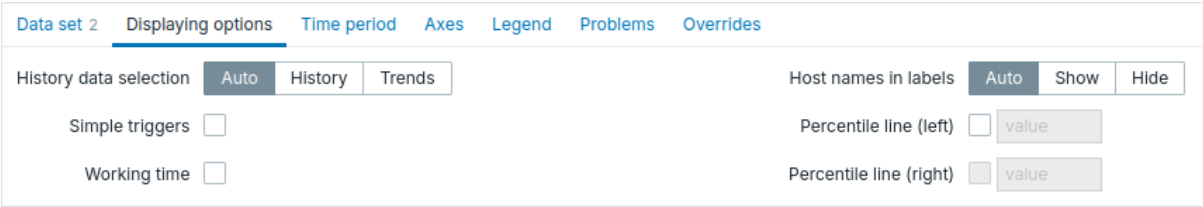
El ejemplo muestra la carga máxima de CPU durante cinco minutos (agregada a partir de valores promedio de un minuto), en comparación con la carga promedio de CPU durante cinco minutos.



Cuando se selecciona *Mostrar función de agregación* para la leyenda, los items se muestran entre paréntesis, precedidos por la función de agregación en la leyenda y en la información sobre herramientas. Si se especifica *Etiqueta del conjunto de datos* para un gráfico, la etiqueta se muestra entre paréntesis, precedida por la función de agregación.

Opciones de visualización

La pestaña **Opciones de visualización** permite definir la selección de datos históricos:

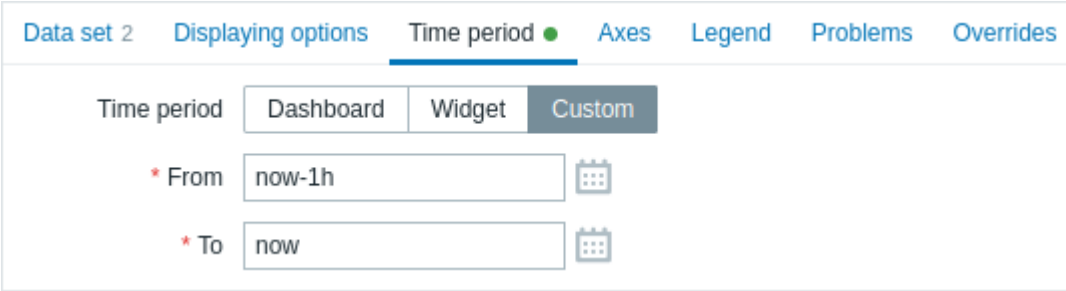


Selección de datos históricos	Establezca el origen de los datos del gráfico: **Auto**: los datos se obtienen según el algoritmo clásico del gráfico (predeterminado); **History**: datos del historial; **Trends**: datos de las tendencias.
Triggers simples	Marque la casilla para mostrar los umbrales de los triggers simples. Los umbrales se dibujarán como líneas discontinuas utilizando el color de gravedad del trigger. Un trigger simple es un trigger con una función (solo `last`, `max`, `min`, `avg`) para un item en la expresión. Se puede dibujar un máximo de tres triggers. Tenga en cuenta que el trigger debe encontrarse dentro del rango dibujado para ser visible.
Tiempo de trabajo	Marque la casilla para mostrar el tiempo de trabajo en el gráfico. El tiempo de trabajo (días laborables) se muestra en los gráficos con un fondo blanco, mientras que el tiempo no laborable se muestra en gris (con el tema frontend predeterminado *Original blue*).

Nombres de host en las etiquetas	Elija si desea mostrar los nombres de los host en la leyenda del gráfico y en la información sobre herramientas que aparece al pasar el cursor sobre los datos del gráfico: **Auto**: los nombres de los host se muestran solo si hay más de un host en todos los conjuntos de datos (predeterminado). **Show**: se muestran los nombres de los host. **Hide**: se ocultan los nombres de los host. Este parámetro no está disponible al configurar el widget en un dashboard de template.
Línea de percentil (izquierda)	Marque la casilla e introduzca el valor del percentil para mostrar el percentil especificado como una línea en el eje Y izquierdo del gráfico. Por ejemplo, si se establece un percentil del 95 %, la línea del percentil se situará en el nivel por debajo del cual se encuentra el 95 % de los valores.
Línea de percentil (derecha)	Marque la casilla e introduzca el valor del percentil para mostrar el percentil especificado como una línea en el eje Y derecho del gráfico. Por ejemplo, si se establece un percentil del 95 %, la línea del percentil se situará en el nivel por debajo del cual se encuentra el 95 % de los valores.

Período de tiempo

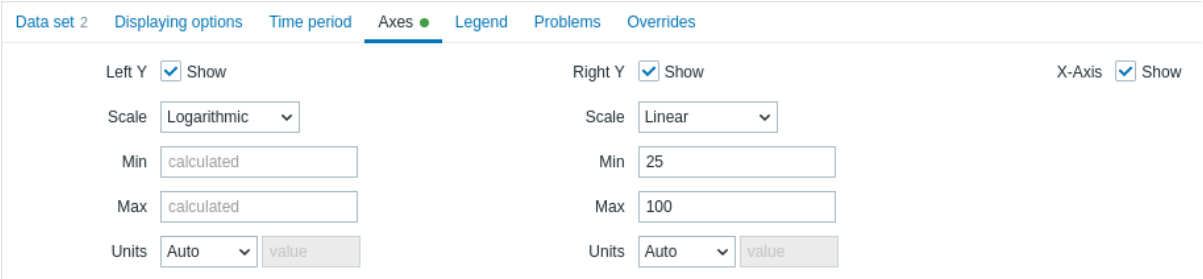
La pestaña **Período de tiempo** permite establecer el período de tiempo para el que se mostrarán los datos en el gráfico:



Período de tiempo	Seleccione la fuente de datos para el período de tiempo: **Dashboard**: utiliza el selector de período de tiempo del dashboard; **Widget**: utiliza un widget compatible (establecido en el parámetro *Widget*); **Personalizado**: utiliza un período de tiempo personalizado establecido en los parámetros *Desde* y *Hasta*; si se establece, se mostrará un icono de reloj en la esquina superior derecha del widget, que indicará la hora establecida al pasar el ratón por encima. Tenga en cuenta que, independientemente de la configuración de *Período de tiempo* del widget, los widgets compatibles aún pueden utilizarlo como fuente de datos para el período de tiempo.
Widget	Introduzca o seleccione un widget compatible como fuente de datos para el período de tiempo. Este parámetro está disponible si *Período de tiempo* está establecido en ”Widget”.
Desde	Introduzca o seleccione el inicio del período de tiempo. Se admite la sintaxis de tiempo relativo (`now`, `now/d`, `now/w-1w`, etc.). Este parámetro está disponible si *Período de tiempo* está establecido en ”Personalizado”.
Hasta	Introduzca o seleccione el final del período de tiempo. Se admite la sintaxis de tiempo relativo (`now`, `now/d`, `now/w-1w`, etc.). Este parámetro está disponible si *Período de tiempo* está establecido en ”Personalizado”.

Ejes

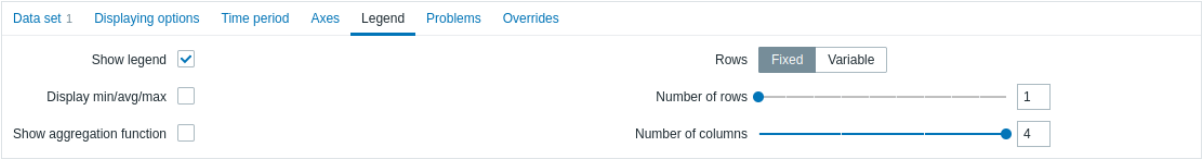
La pestaña **Ejes** permite personalizar cómo se muestran los ejes:



Y izquierdo	Marque esta casilla para hacer visible el eje Y izquierdo. La casilla puede estar deshabilitada si no está seleccionada en la pestaña *Conjunto de datos* o *Anulaciones*.
Y derecho	Marque esta casilla para hacer visible el eje Y derecho. La casilla puede estar deshabilitada si no está seleccionada en la pestaña *Conjunto de datos* o *Anulaciones*.
Eje X	Desmarque esta casilla para ocultar el eje X (marcada de forma predeterminada).
Escala	Seleccione la escala para los valores del eje del gráfico en la lista desplegable: **Lineal**: los valores del eje aumentan en una cantidad fija (por ejemplo, 10, 20, 30), lo que resulta adecuado para datos que cambian de forma constante o abarcan un rango pequeño o moderado; **Logarítmica**: los valores del eje aumentan exponencialmente (por ejemplo, 10, 100, 1000), lo que resulta adecuado para datos que cambian rápidamente o abarcan un rango amplio.
Mín.	Establezca el valor mínimo del eje correspondiente. Se especifica el valor mínimo del rango visible del eje Y.
Máx.	Establezca el valor máximo del eje correspondiente. Se especifica el valor máximo del rango visible del eje Y.
Unidades	Seleccione la unidad para los valores del eje del gráfico en la lista desplegable: **Automático**: los valores del eje se muestran utilizando la unidad del primer item del conjunto de datos; **Estático**: los valores del eje se muestran utilizando la unidad especificada en el campo de entrada *valor*; si el campo se deja vacío, solo se muestran valores numéricos.

Leyenda

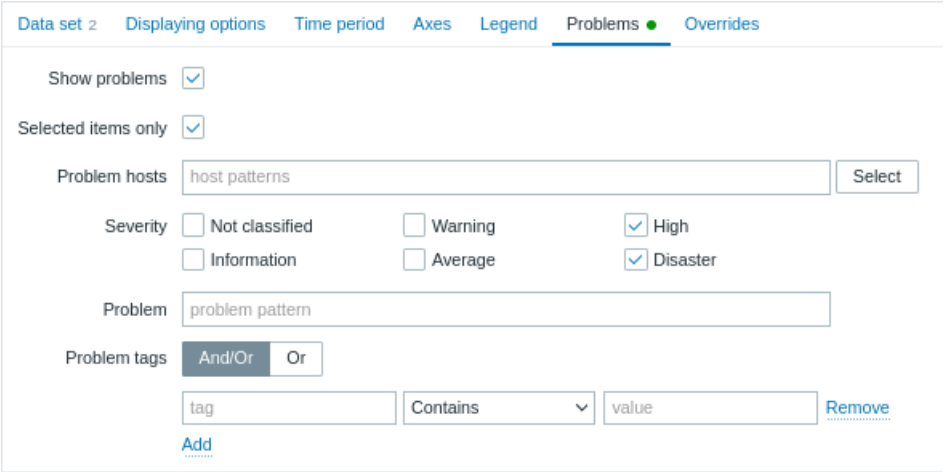
La pestaña **Leyenda** permite personalizar la leyenda del gráfico:



Mostrar leyenda	Desmarque esta casilla para ocultar la leyenda en el gráfico (marcada de forma predeterminada).
Mostrar mín./prom./máx.	Marque esta casilla para mostrar los valores mínimo, promedio y máximo del item en la leyenda.
Mostrar función de agregación	Marque esta casilla para mostrar la función de agregación en la leyenda.
Filas	Seleccione el modo de visualización de las filas de la leyenda: **Fijo**: el número de filas mostradas está determinado por el valor del parámetro *Número de filas*; **Variable**: el número de filas mostradas está determinado por la cantidad de items configurados, sin superar el valor del parámetro *Número máximo de filas*.
Número de filas/ Número máximo de filas	Si *Filas* está establecido en ”Fijo”, establezca el número de filas de la leyenda que se mostrarán (1-10). Si *Filas* está establecido en ”Variable”, establezca el número máximo de filas de la leyenda que se mostrarán (1-10).
Número de columnas	Establezca el número de columnas de la leyenda que se mostrarán (1-4). Este parámetro está disponible si *Mostrar mín./prom./máx.* no está marcado.

Problemas

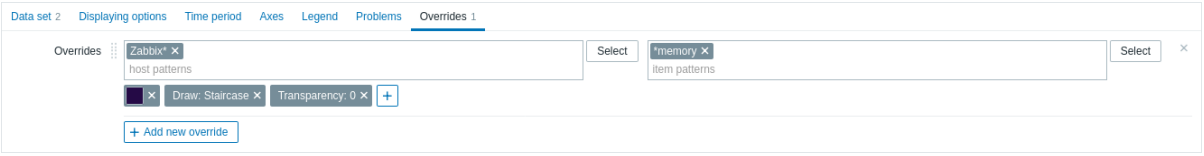
La pestaña **Problemas** permite personalizar la visualización de los problemas:



Mostrar problemas	Marque esta casilla para habilitar la visualización de problemas en el gráfico (sin marcar, es decir, deshabilitada de forma predeterminada).
Solo items seleccionados	Marque esta casilla para incluir únicamente los problemas de los items seleccionados que se mostrarán en el gráfico.
Hosts con problemas	Seleccione los hosts con problemas que se mostrarán en el gráfico. Se pueden utilizar patrones comodín (por ejemplo, * devolverá resultados que coincidan con cero o más caracteres). Para especificar un patrón comodín, introduzca la cadena manualmente y pulse *Enter*. Mientras escribe, observe cómo se muestran todos los hosts coincidentes en la lista desplegable. Este parámetro no está disponible al configurar el widget en un dashboard de template.
Gravedad	Marque las gravedades de los problemas para filtrar los problemas que se mostrarán en el gráfico. Si no se marca ninguna gravedad, se mostrarán todos los problemas.
Problema	Especifique el nombre del problema que se mostrará en el gráfico.
Etiquetas de problemas	Especifique las etiquetas de los problemas para limitar el número de problemas mostrados en el widget. Es posible incluir y excluir etiquetas y valores de etiquetas específicos. Se pueden establecer varias condiciones. La coincidencia de nombres de etiquetas siempre distingue entre mayúsculas y minúsculas. Hay varios operadores disponibles para cada condición: **Existe**: incluye los nombres de etiqueta especificados; **Es igual a**: incluye los nombres y valores de etiqueta especificados (distingue entre mayúsculas y minúsculas); **Contiene**: incluye los nombres de etiqueta especificados cuyos valores contienen la cadena introducida (coincidencia de subcadena, sin distinguir entre mayúsculas y minúsculas); **No existe**: excluye los nombres de etiqueta especificados; **No es igual a**: excluye los nombres y valores de etiqueta especificados (distingue entre mayúsculas y minúsculas); **No contiene**: excluye los nombres de etiqueta especificados cuyos valores contienen la cadena introducida (coincidencia de subcadena, sin distinguir entre mayúsculas y minúsculas). Hay dos tipos de cálculo para las condiciones: **Y/O**: deben cumplirse todas las condiciones; las condiciones que tienen el mismo nombre de etiqueta se agruparán mediante la condición *O*; **O**: basta con que se cumpla una condición.

Anulaciones

La pestaña **Anulaciones** permite añadir anulaciones personalizadas para conjuntos de datos:



Las anulaciones son útiles cuando se seleccionan varios items para un conjunto de datos mediante el comodín * y se desea cambiar la forma en que los items se muestran de forma predeterminada (por ejemplo, el color base predeterminado u otra propiedad).

Las anulaciones existentes, si las hay, se muestran en una lista.
Para añadir una nueva anulación:

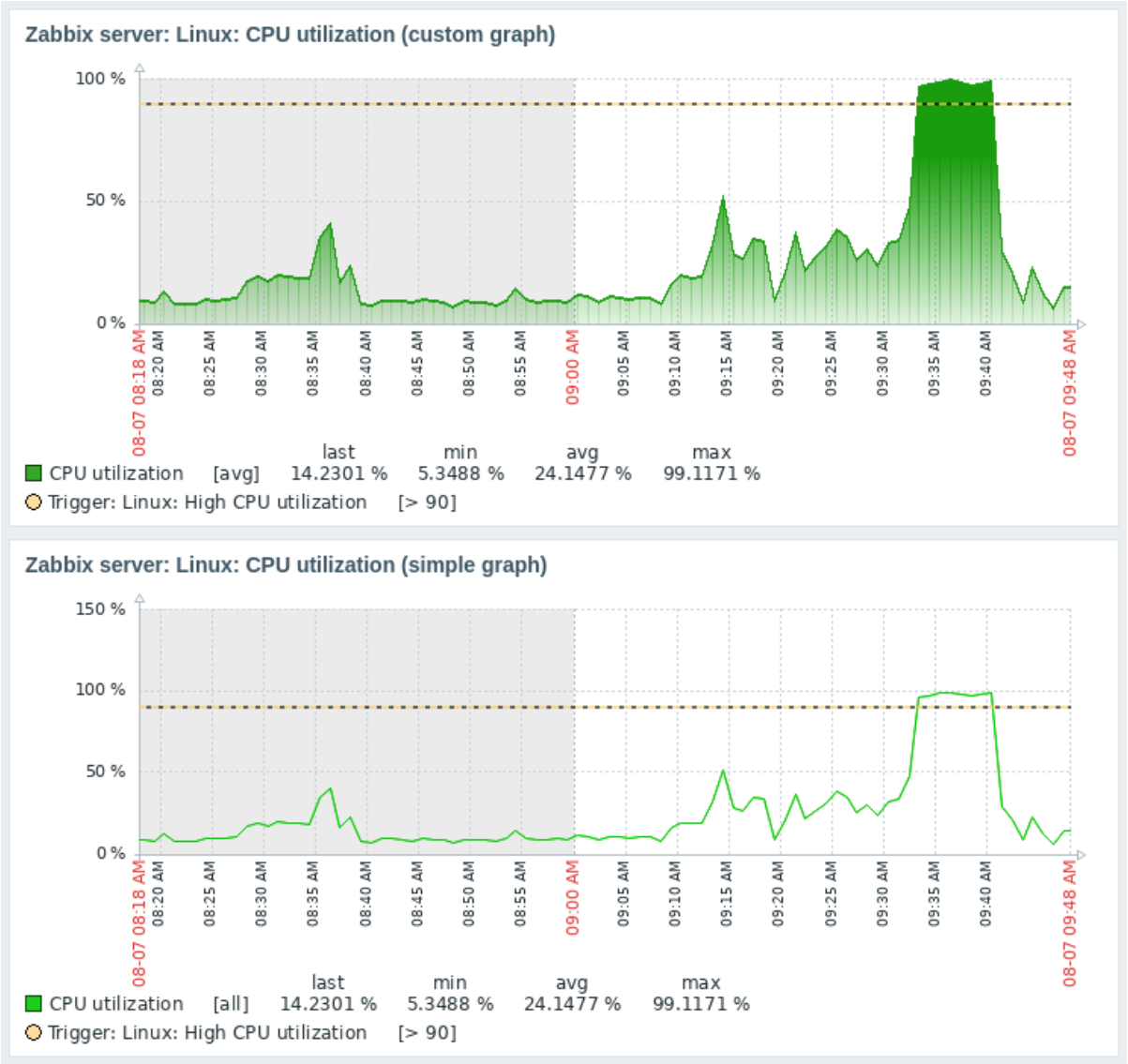
- Haga clic en el botón 

- Seleccione los hosts y los items para la anulación.
  Como alternativa, puede introducir patrones de host y de item.
  Se pueden utilizar patrones con comodines (por ejemplo, * devolverá los resultados que coincidan con cero o más caracteres).
  Para especificar un patrón con comodines, introduzca la cadena manualmente y pulse *Enter*.
  Mientras escribe, observe cómo se muestran todos los hosts coincidentes en la lista desplegable.
  El símbolo de comodín siempre se interpreta, por lo que no es posible añadir individualmente, por ejemplo, un item denominado "item*" si existen otros items coincidentes (por ejemplo, item2, item3).
  Los parámetros de patrón de host y patrón de item son obligatorios.
  El parámetro para especificar patrones de host no está disponible al configurar el widget en un dashboard de template.
  El parámetro para especificar una lista de items solo permite seleccionar items configurados en el template al configurar el widget en un dashboard de template.
- Haga clic en  para seleccionar los parámetros de anulación.
  Debe seleccionarse al menos un parámetro de anulación.
  Para obtener descripciones de los parámetros, consulte la pestaña *Conjunto de datos* anterior.

Gráfico (clásico)

Descripción general

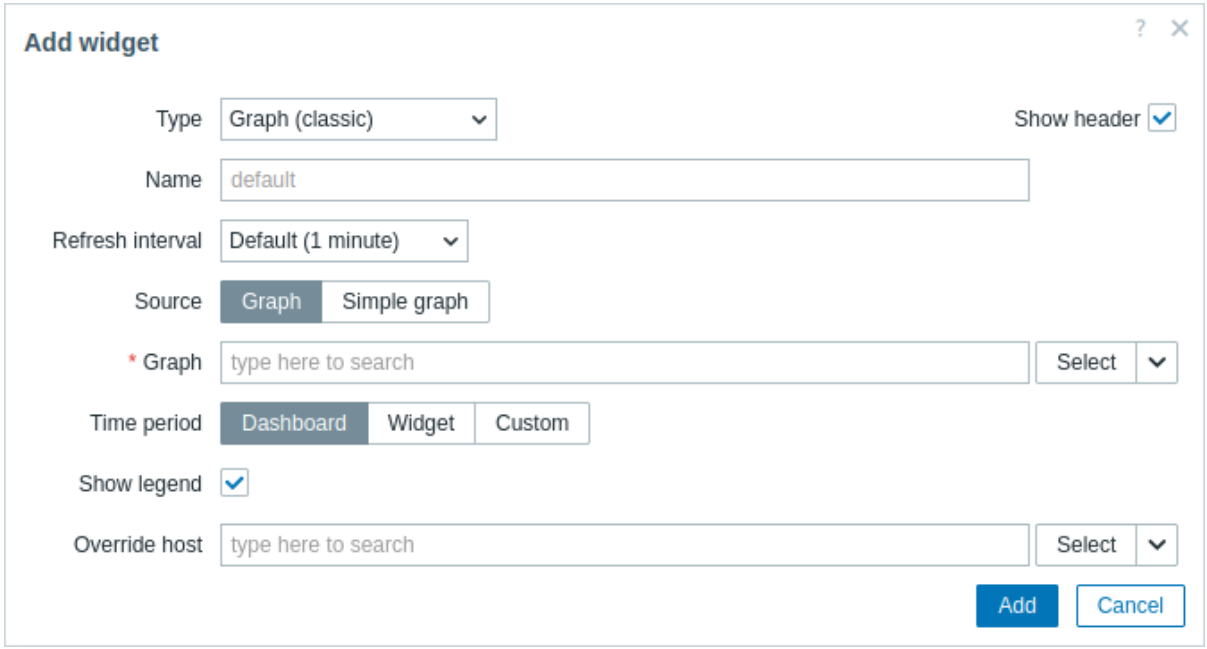
El widget *Graph (classic)* muestra los datos numéricos de un item como una gráfica personalizada o una gráfica simple basada en una imagen. Admite la selección del período de tiempo, líneas de trigger y la visualización opcional de una leyenda.



La información que se muestra en el widget *Graph (classic)* se puede descargar como una imagen PNG seleccionando la opción *Download image* en el menú *Actions* del widget ⋯, que aparece al pasar el cursor sobre el widget. Según la configuración del navegador, la imagen se guarda automáticamente en la carpeta `Downloads` o se abre un cuadro de diálogo *Save* que permite elegir la ubicación.

Configuración

Para configurar, seleccione *Graph (classic)* como tipo:



Además de los parámetros que son comunes para todos los widgets, puede establecer las siguientes opciones específicas:

Source	Seleccione el tipo de gráfico: **Graph** - gráfico personalizado; **Simple graph** - gráfico simple.
Graph	Seleccione el gráfico personalizado a mostrar. Alternativamente, seleccione un widget compatible como fuente de datos para los gráficos. Este parámetro está disponible si *Source* está configurado en ”Graph”.
Item	Seleccione el item a mostrar en un gráfico simple. Alternativamente, seleccione un widget compatible como fuente de datos para los items. Este parámetro está disponible si *Source* está configurado en ”Simple graph”.
Time period	Establezca un período de tiempo para mostrar los datos en el gráfico. Seleccione la fuente de datos para el período de tiempo: **Dashboard** - use el selector de período de tiempo del dashboard; **Widget** - use un widget compatible (establecido en el parámetro *Widget*); **Custom** - use un período de tiempo personalizado establecido en los parámetros *From* y *To*; si se establece, se mostrará un icono de reloj en la esquina superior derecha del widget, indicando el tiempo configurado al pasar el ratón. Tenga en cuenta que, independientemente de la configuración del *Time period* del widget, los widgets compatibles aún pueden usarlo como fuente de datos para el período de tiempo.
Widget	Introduzca o seleccione un widget compatible como fuente de datos para el período de tiempo. Este parámetro está disponible si *Time period* está configurado en ”Widget”.
From	Introduzca o seleccione el inicio del período de tiempo. Se admite la sintaxis de tiempo relativo (`now`, `now/d`, `now/w-1w`, etc.). Este parámetro está disponible si *Time period* está configurado en ”Custom”.
To	Introduzca o seleccione el final del período de tiempo. Se admite la sintaxis de tiempo relativo (`now`, `now/d`, `now/w-1w`, etc.). Este parámetro está disponible si *Time period* está configurado en ”Custom”.
Show legend	Desmarque esta casilla para ocultar la leyenda en el gráfico (marcada por defecto).
Override host	Seleccione un widget compatible o el selector de host del dashboard como fuente de datos para los hosts. Este parámetro no está disponible al configurar el widget en un dashboard de template.

La leyenda del gráfico consta de tres secciones:

- Items y sus valores agregados
- Percentiles (si están configurados)
- Triggers (si hay alguno asociado a los items mostrados)

Si la altura del gráfico dentro del widget es insuficiente, la leyenda puede no mostrarse o mostrarse solo parcialmente. Los triggers y percentiles se ocultan primero, seguidos de la leyenda de los items. Para mostrar la leyenda completa, aumente el tamaño vertical del widget.

Note:
No se pueden mostrar más de 3 líneas de trigger. Si hay más triggers, se priorizan para mostrar los triggers con menor severidad.

Prototipo de gráfico

Descripción general

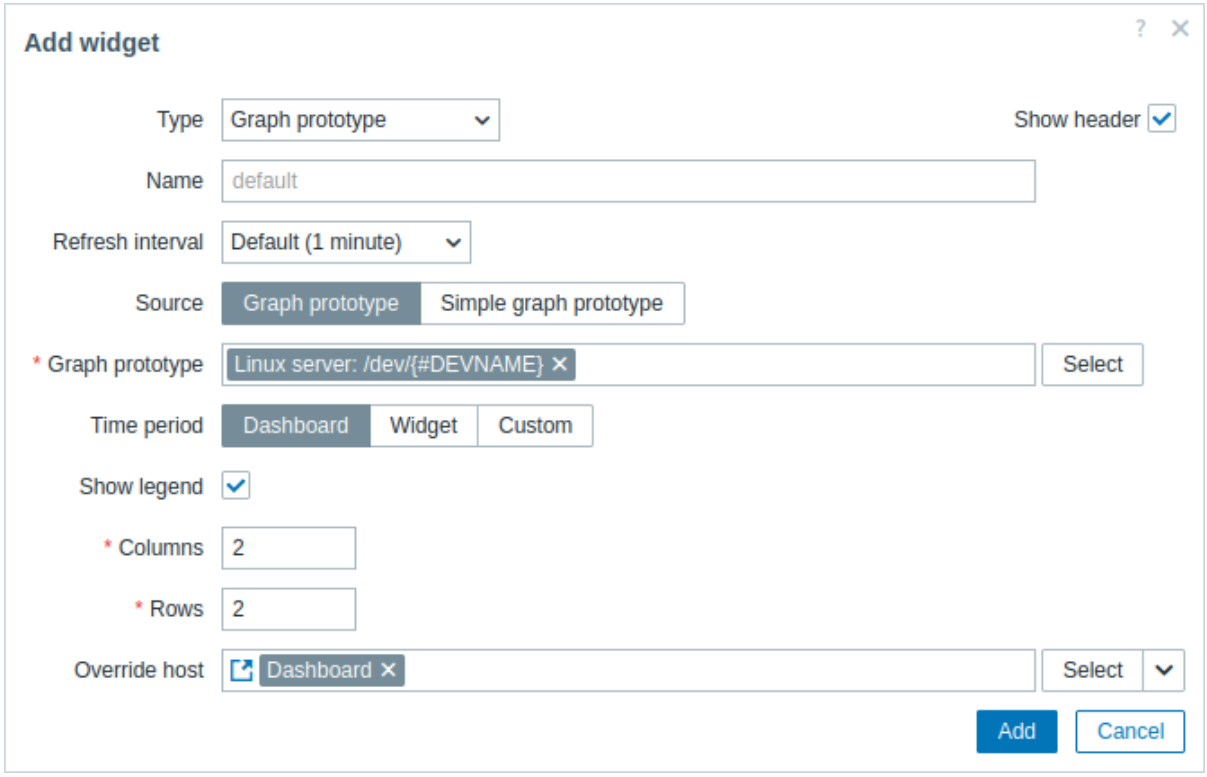
El widget *Prototipo de gráfico* muestra una cuadrícula de gráficos personalizados creados automáticamente basados en reglas de descubrimiento de bajo nivel prototipos de gráficos o prototipos de ítems.



Los gráficos se organizan en un diseño de cuadrícula configurable; si el número de gráficos excede el diseño, se habilita la paginación en el widget.

Configuración

Para configurar, seleccione *Graph prototype* como tipo de widget:



Además de los parámetros comunes a todos los widgets, puede establecer las siguientes opciones específicas:

Source	Seleccione el origen de los gráficos: **Graph prototype** o **Simple graph prototype**.
Graph prototype	Seleccione un prototipo de gráfico para mostrar los gráficos descubiertos por el prototipo de gráfico. Este parámetro está disponible si *Source* está establecido en ”Graph prototype”.
Item prototype	Seleccione un prototipo de item para mostrar gráficos simples de los items descubiertos por el prototipo de item. Este parámetro está disponible si *Source* está establecido en ”Simple graph prototype”.
Time period	Establezca el período de tiempo durante el cual se mostrarán los datos en los gráficos. Seleccione el origen de datos para el período de tiempo: **Dashboard**: utiliza el selector de período de tiempo del dashboard; **Widget**: utiliza un widget compatible (establecido en el parámetro *Widget*); **Custom**: utiliza un período de tiempo personalizado establecido en los parámetros *From* y *To*; si se establece, se mostrará un icono de reloj en la esquina superior derecha del widget, que indicará la hora establecida al pasar el ratón por encima. Tenga en cuenta que, independientemente de la configuración de *Time period* del widget, los widgets compatibles pueden seguir utilizándolo como origen de datos para el período de tiempo.
Widget	Introduzca o seleccione un widget compatible como origen de datos para el período de tiempo. Este parámetro está disponible si *Time period* está establecido en ”Widget”.
From	Introduzca o seleccione el inicio del período de tiempo. Se admite la sintaxis de tiempo relativo (`now`, `now/d`, `now/w-1w`, etc.). Este parámetro está disponible si *Time period* está establecido en ”Custom”.
To	Introduzca o seleccione el final del período de tiempo. Se admite la sintaxis de tiempo relativo (`now`, `now/d`, `now/w-1w`, etc.). Este parámetro está disponible si *Time period* está establecido en ”Custom”.
Show legend	Desmarque esta casilla para ocultar la leyenda de los gráficos (marcada de forma predeterminada).
Override host	Seleccione un widget compatible o el selector de host del dashboard como origen de datos para los hosts. Este parámetro no está disponible al configurar el widget en un dashboard de template.
Columns	Introduzca el número de columnas de gráficos que se mostrarán dentro de un widget de prototipo de gráfico.
Rows	Introduzca el número de filas de gráficos que se mostrarán dentro de un widget de prototipo de gráfico.

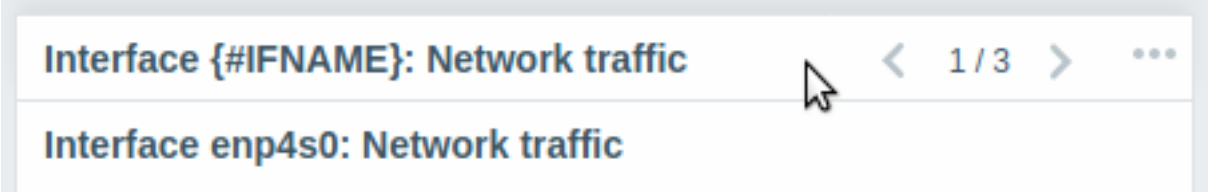
La leyenda del gráfico consta de tres secciones:

- Items y sus valores agregados
- Percentiles (si están configurados)
- Triggers (si hay alguno asociado con los items mostrados)

Si la altura del gráfico dentro del widget es insuficiente, es posible que la leyenda no se muestre o que solo se muestre parcialmente. Los triggers y los percentiles se ocultan primero, seguidos de la leyenda de los items. Para mostrar la leyenda completa, aumente el tamaño vertical del widget.

Note:
No se pueden mostrar más de 3 líneas de trigger. Si hay más triggers, se priorizan para su visualización los triggers con menor nivel de severidad.

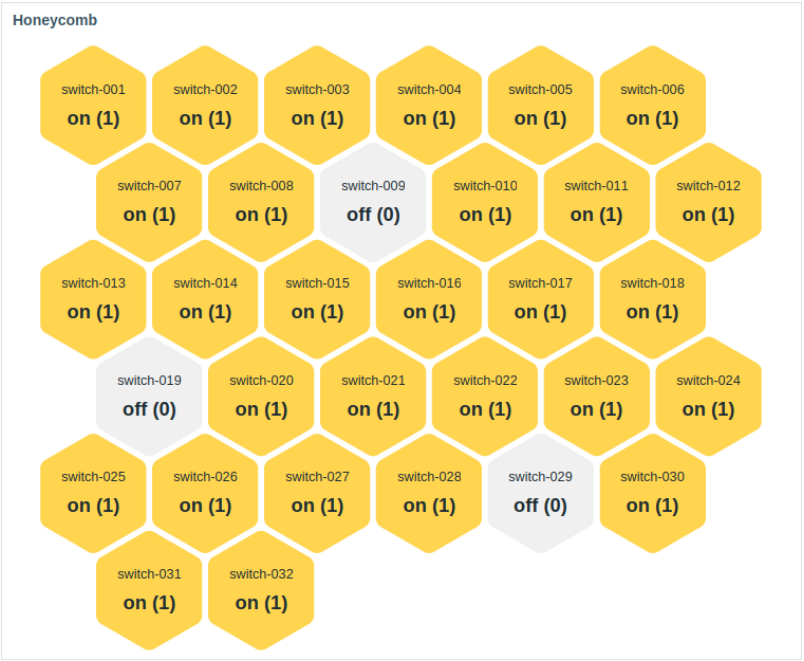
Aunque los parámetros *Columns* y *Rows* permiten ajustar más de un gráfico en el widget, aún puede haber más gráficos descubiertos que columnas/filas en el widget. En este caso, la paginación estará disponible en el widget y un encabezado desplegable permitirá cambiar entre las páginas mediante las flechas izquierda y derecha:

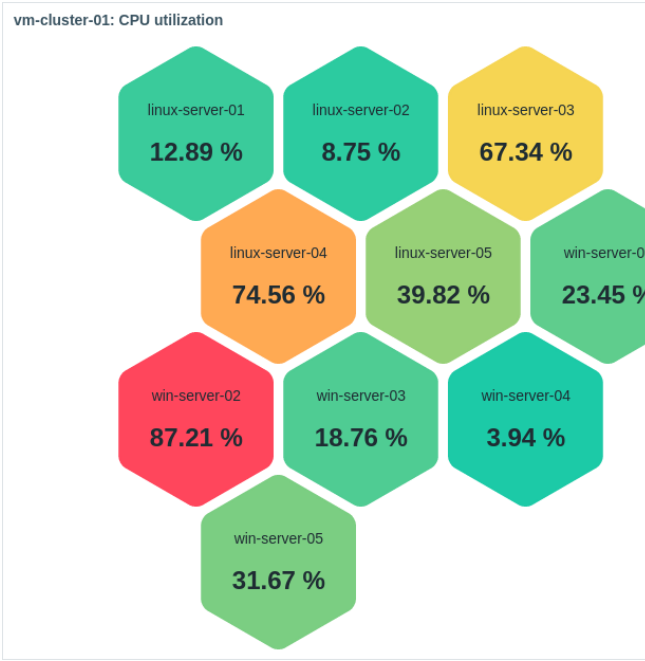


Honeycomb

Descripción general

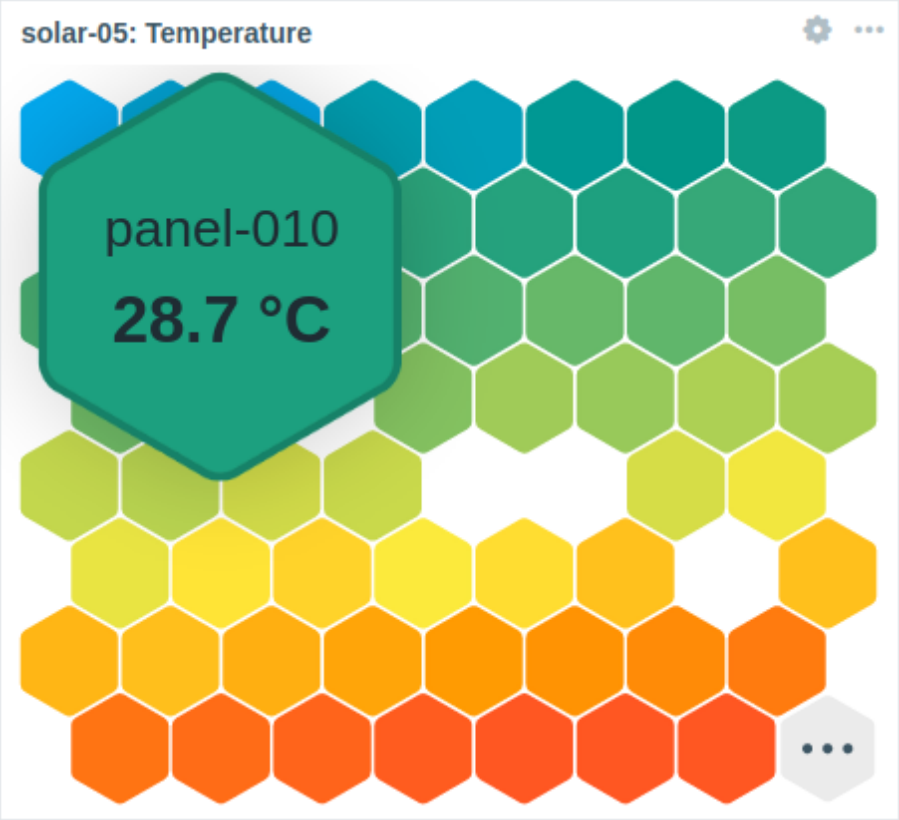
El widget *Honeycomb* muestra hosts o items como celdas hexagonales en una cuadrícula. Este diseño facilita la detección de hosts/items con problemas, la identificación de grupos de problemas, la comparación de grupos de un vistazo y mucho más.





El número de celdas mostradas está limitado por el tamaño del widget y el tamaño mínimo de las celdas (32px). Si no caben todas las celdas, la última muestra puntos suspensivos. El widget se puede redimensionar; el número, el tamaño y la posición de las celdas se ajustan dinámicamente.

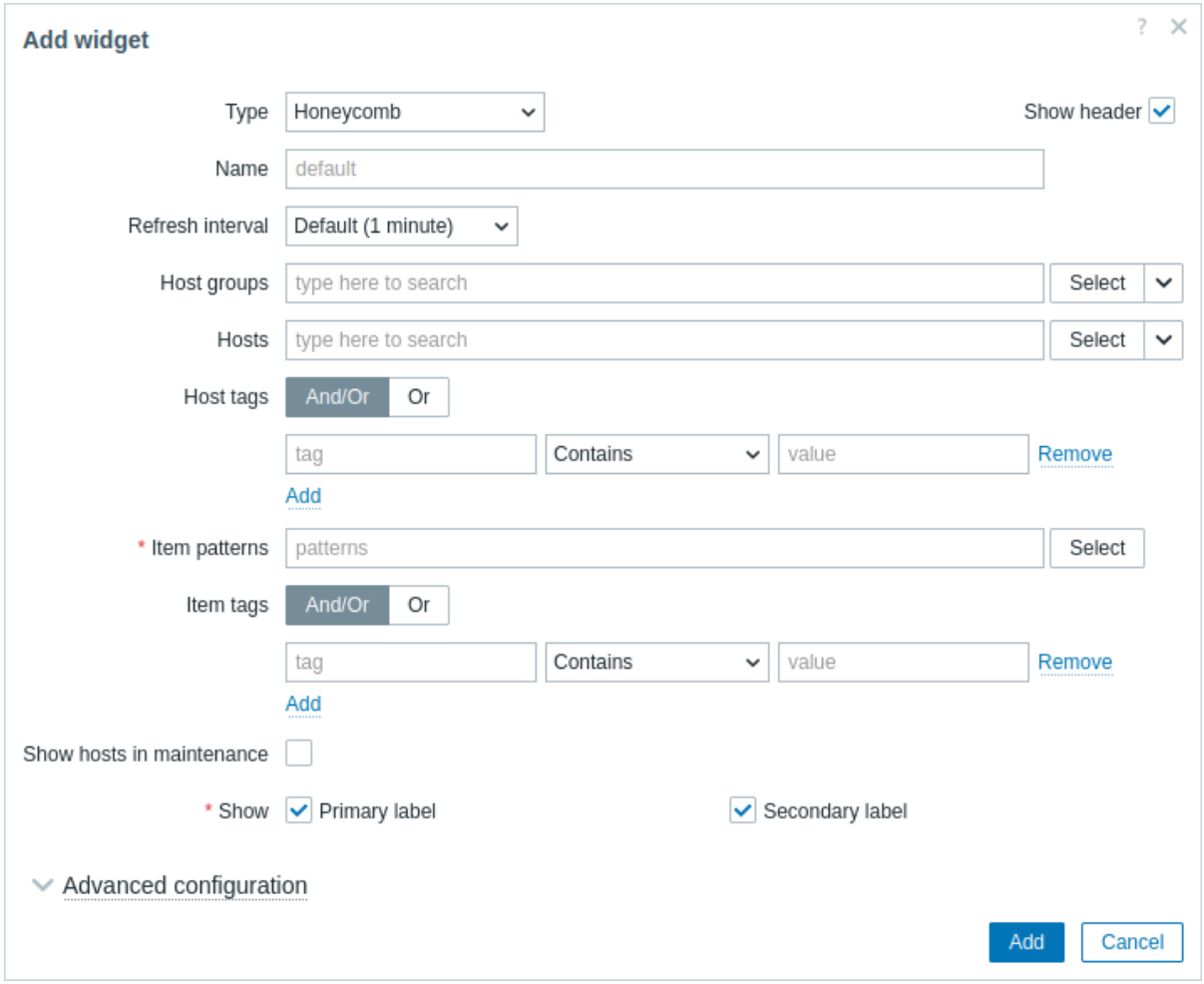
Al pasar el ratón por encima, la celda enfocada aumenta de tamaño para mejorar su visibilidad. Al hacer clic en una celda, se resalta su borde hasta que se selecciona otra celda.



La información mostrada en el widget *Honeycomb* se puede descargar como una imagen PNG seleccionando la opción *Descargar imagen* en el menú *Acciones* del widget ⋯, que aparece al pasar el ratón por encima del widget. Según la configuración del navegador, la imagen se guarda automáticamente en la carpeta `Downloads` o se abre un cuadro de diálogo *Guardar* que permite elegir la ubicación.

Configuración

Para configurar, seleccione *Honeycomb* como tipo:



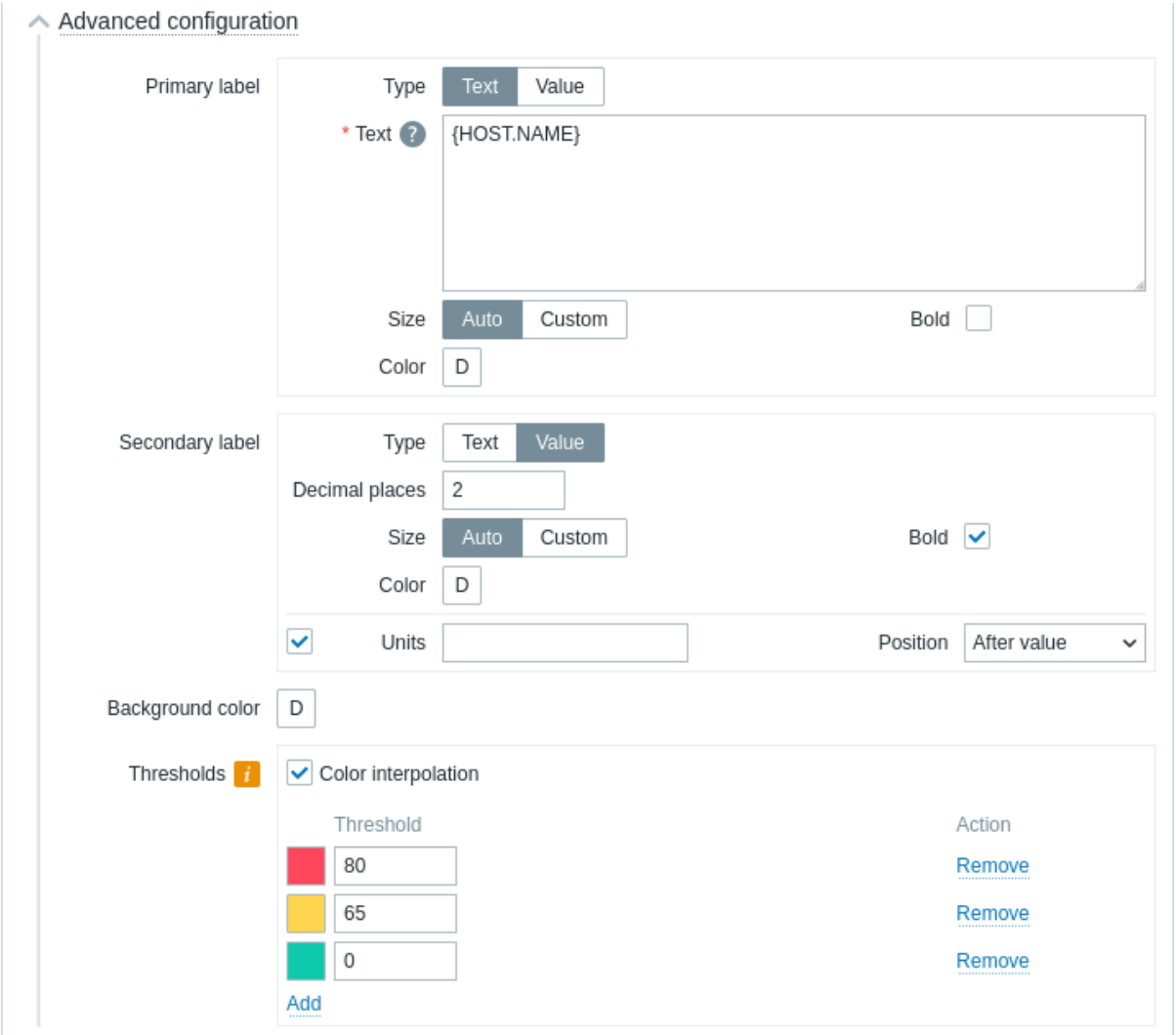
Además de los parámetros que son comunes para todos los widgets, puede establecer las siguientes opciones específicas:

Grupos de hosts	Seleccione grupos de hosts. Como alternativa, seleccione un widget compatible como origen de datos para los grupos de hosts. Este campo tiene autocompletado, por lo que al comenzar a escribir el nombre de un grupo se ofrecerá una lista desplegable de grupos coincidentes. Al seleccionar un grupo de hosts padre, se seleccionan implícitamente todos los grupos de hosts anidados; si no se selecciona ningún grupo de hosts, el widget mostrará todos los grupos de hosts que contengan hosts con items que coincidan con el *Patrón de item* seleccionado (véase más abajo). Este parámetro no está disponible al configurar el widget en un dashboard de template.
Hosts	Seleccione hosts. Como alternativa, seleccione un widget compatible o el dashboard como origen de datos para los hosts. Este campo tiene autocompletado, por lo que al comenzar a escribir el nombre de un host se ofrecerá una lista desplegable de hosts coincidentes. Si no se selecciona ningún host, el widget mostrará todos los hosts con items que coincidan con el *Patrón de item* seleccionado (véase más abajo). Este parámetro no está disponible al configurar el widget en un dashboard de template.

Etiquetas de host	Especifique etiquetas para limitar el número de hosts mostrados en el widget. Es posible incluir, así como excluir, etiquetas y valores de etiqueta específicos. Se pueden establecer varias condiciones. La coincidencia del nombre de la etiqueta siempre distingue entre mayúsculas y minúsculas. Hay varios operadores disponibles para cada condición: **Existe** - incluir los nombres de etiqueta especificados; **Igual a** - incluir los nombres y valores de etiqueta especificados (distingue entre mayúsculas y minúsculas); **Contiene** - incluir los nombres de etiqueta especificados cuyos valores de etiqueta contengan la cadena introducida (coincidencia de subcadena, sin distinguir entre mayúsculas y minúsculas); **No existe** - excluir los nombres de etiqueta especificados; **No es igual a** - excluir los nombres y valores de etiqueta especificados (distingue entre mayúsculas y minúsculas); **No contiene** - excluir los nombres de etiqueta especificados cuyos valores de etiqueta contengan la cadena introducida (coincidencia de subcadena, sin distinguir entre mayúsculas y minúsculas). Hay dos tipos de cálculo para las condiciones: **And/Or** - deben cumplirse todas las condiciones; las condiciones que tengan el mismo nombre de etiqueta se agruparán por la condición Or; **Or** - basta con que se cumpla una condición. Este parámetro no está disponible al configurar el widget en un dashboard de template.
Patrones de item	Introduzca patrones de item o seleccione items existentes como patrones de item. Los datos de los items que coincidan con los patrones introducidos o seleccionados se mostrarán en el honeycomb. El parámetro *Patrones de item* es obligatorio. Se pueden usar patrones comodín para la selección (por ejemplo, `*` devolverá items que coincidan con cero o más caracteres; `Zabbix*` devolverá items que comiencen con "Zabbix"). Para especificar un patrón comodín, introduzca la cadena manualmente y pulse *Enter*. Cuando comience a escribir, una lista desplegable mostrará los items coincidentes limitados a los que pertenezcan a los *Hosts* seleccionados o a hosts dentro de los *Grupos de hosts* seleccionados, si los hubiera. El símbolo comodín siempre se interpreta, por lo tanto, no es posible añadir, por ejemplo, un item llamado *item** de forma individual, si hay otros items coincidentes (p. ej., item2, item3). Al configurar el widget en un dashboard de template, este parámetro permite seleccionar solo items configurados en el template.
Etiquetas de item	Especifique etiquetas para limitar el número de items mostrados en el widget. Para más información, consulte *Etiquetas de host* más arriba.
Mostrar hosts en mantenimiento	Marque esta casilla para mostrar hosts en mantenimiento (en este caso, se mostrará el icono de mantenimiento junto al nombre del host). Este parámetro se etiqueta como *Mostrar datos en mantenimiento* al configurar el widget en un dashboard de template.
Mostrar	Marque esta casilla para mostrar el elemento correspondiente de la celda honeycomb: etiqueta principal, etiqueta secundaria. Debe seleccionarse al menos un elemento.
Configuración avanzada	Haga clic en el encabezado *Configuración avanzada* para mostrar las opciones de configuración avanzada.

Configuración avanzada

Las opciones de configuración avanzada están disponibles en la sección desplegable *Configuración avanzada* y solo para los elementos seleccionados en el campo *Mostrar* (véase arriba), así como el color de fondo o los umbrales de las celdas del panal.



Etiqueta principal/secundaria

Tipo — Seleccione el tipo de etiqueta:
Texto: la etiqueta mostrará el texto especificado para el parámetro *Texto*;
Valor: la etiqueta mostrará el valor del item con el número de decimales especificado para el parámetro *Posiciones decimales*.

Texto — Introduzca el texto de la etiqueta. Este texto puede sustituir el nombre predeterminado del item.
Se admite texto multilínea. Es posible combinar texto y macros compatibles.
Se admiten `{HOST.\*}`, `{ITEM.\*}`, `{INVENTORY.\*}` y macros de usuario.
Las celdas del panal se ordenan alfabéticamente por nombre de host y, dentro de cada host, por nombre de item.
Este parámetro está disponible si *Tipo* está establecido en *Texto*.

Posiciones decimales — Introduzca el número de posiciones decimales que se mostrarán con el valor.
Este parámetro está disponible si *Tipo* está establecido en *Valor* y solo afecta a los items que devuelven datos numéricos (float).

Tamaño — Seleccione el tamaño de la etiqueta:
Automático: usar el tamaño de etiqueta ajustado automáticamente;
Personalizado: introducir un tamaño de etiqueta personalizado (en porcentaje, relativo al tamaño de la celda del panal).
Tenga en cuenta que las etiquetas que no se ajusten al tamaño de la celda del panal se truncarán.

Negrita — Marque la casilla para mostrar las etiquetas en negrita.

Color — Seleccione el color de la etiqueta en el selector de color.
"D" representa el color predeterminado, que depende del tema del frontend. Para volver al color predeterminado, haga clic en el botón *Usar predeterminado* del selector de color.

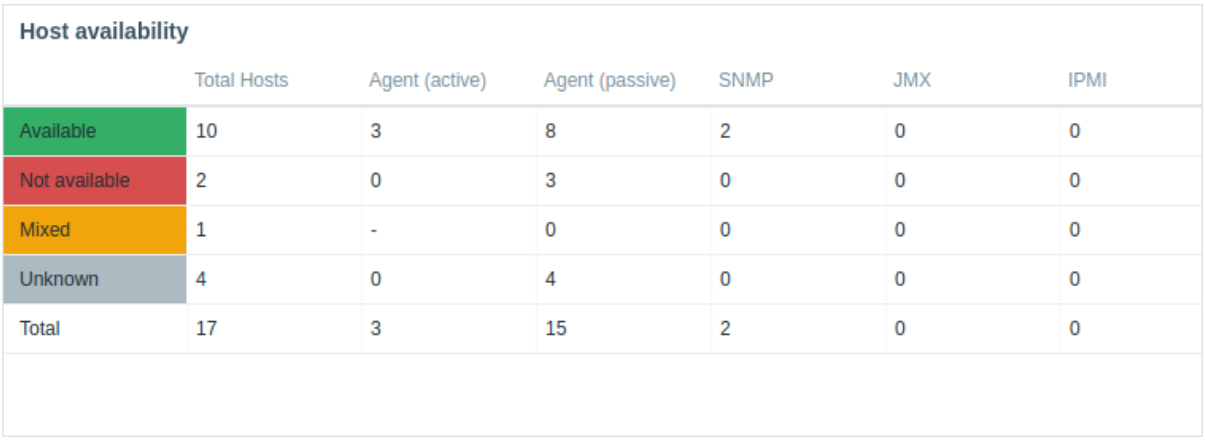
Unidades

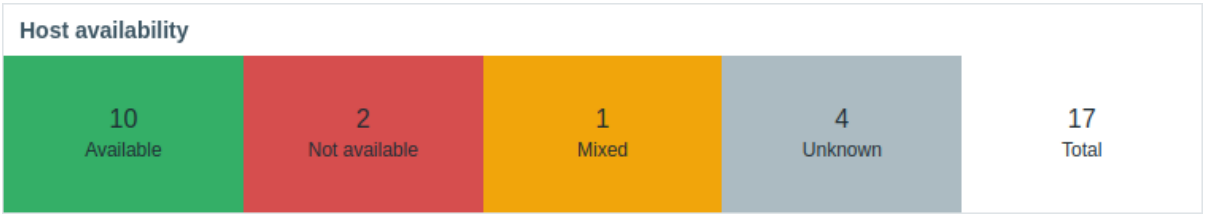
Unidades	Marque la casilla para mostrar las unidades con el valor del item. Si introduce un nombre de unidad, este sustituirá a las unidades establecidas en la configuración del item. Este parámetro está disponible si *Tipo* está establecido en *Valor*.
Posición	Seleccione la posición de las unidades del item (antes o después del valor del item). Este parámetro se ignora para las siguientes unidades relacionadas con el tiempo: unixtime, uptime, s. Este parámetro está disponible si *Tipo* está establecido en *Valor* y la opción *Unidades* está seleccionada.
Color de fondo	
Color de fondo	Seleccione el color de fondo de las celdas del panal en el selector de color. ”D” representa el color predeterminado, que depende del tema del frontend. Para volver al color predeterminado, haga clic en el botón *Usar predeterminado* del selector de color.
Umbrales	
Interpolación de color	Marque la casilla para habilitar una transición suave entre los colores de umbral de las celdas del panal. Este parámetro está disponible si se han establecido dos o más umbrales.
Umbral	Haga clic en *Añadir* para agregar un umbral, seleccione un color de umbral en el selector de color y especifique un valor numérico. La lista de umbrales se ordenará en orden ascendente al guardarse. Tenga en cuenta que los colores configurados como umbrales se mostrarán correctamente solo para items numéricos. Se admiten sufijos (por ejemplo, ”1d”, ”2w”, ”4K”, ”8G”). Se admiten mapeos de valores.

Disponibilidad del host

Descripción general

El widget *Disponibilidad de equipos* muestra la disponibilidad de los equipos en los grupos de equipos seleccionados, lo que le permite supervisar el número de equipos que están activos o inactivos.



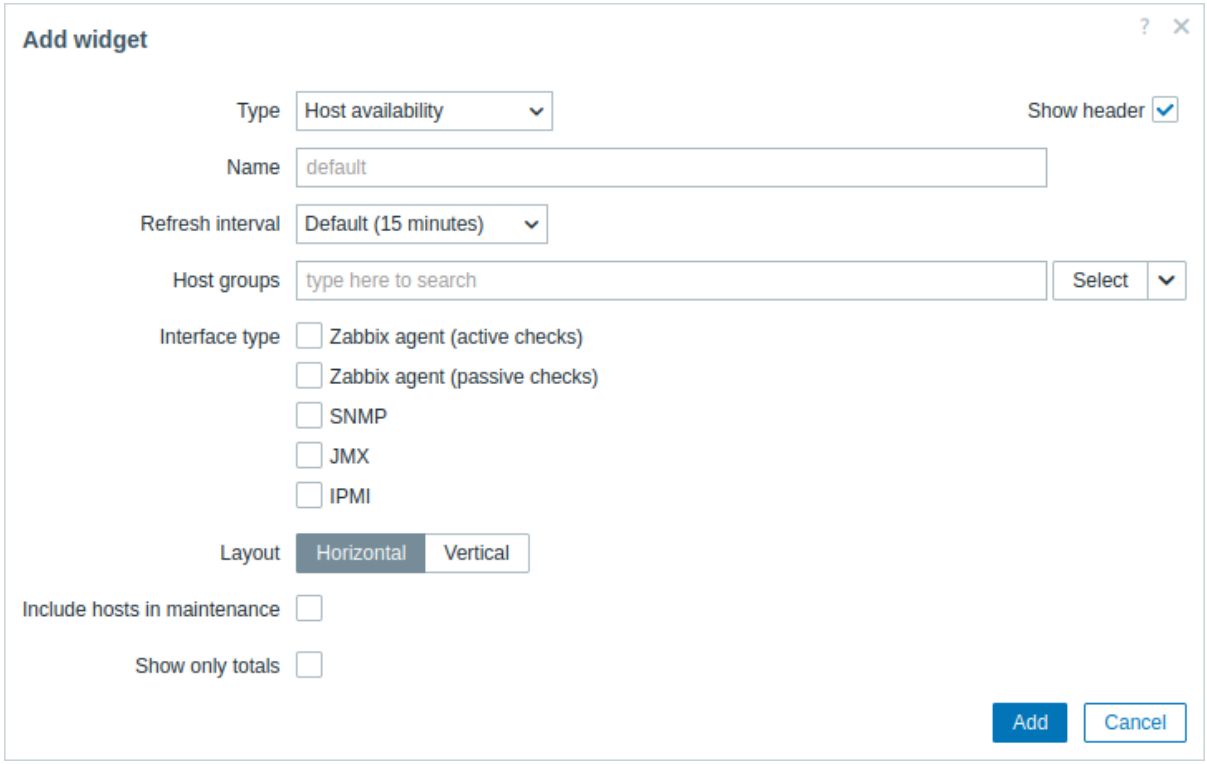


La disponibilidad de los equipos se cuenta de la siguiente manera:

- *Disponible*: equipos con todas las interfaces disponibles
- *No disponible*: equipos con todas las interfaces no disponibles
- *Mixto*: equipos con al menos una interfaz no disponible y al menos una disponible o desconocida
- *Desconocido*: equipos con al menos una interfaz desconocida, pero ninguna no disponible
- *Total*: total de todos los equipos

Configuración

Para configurar, seleccione *Disponibilidad de equipos* como tipo:



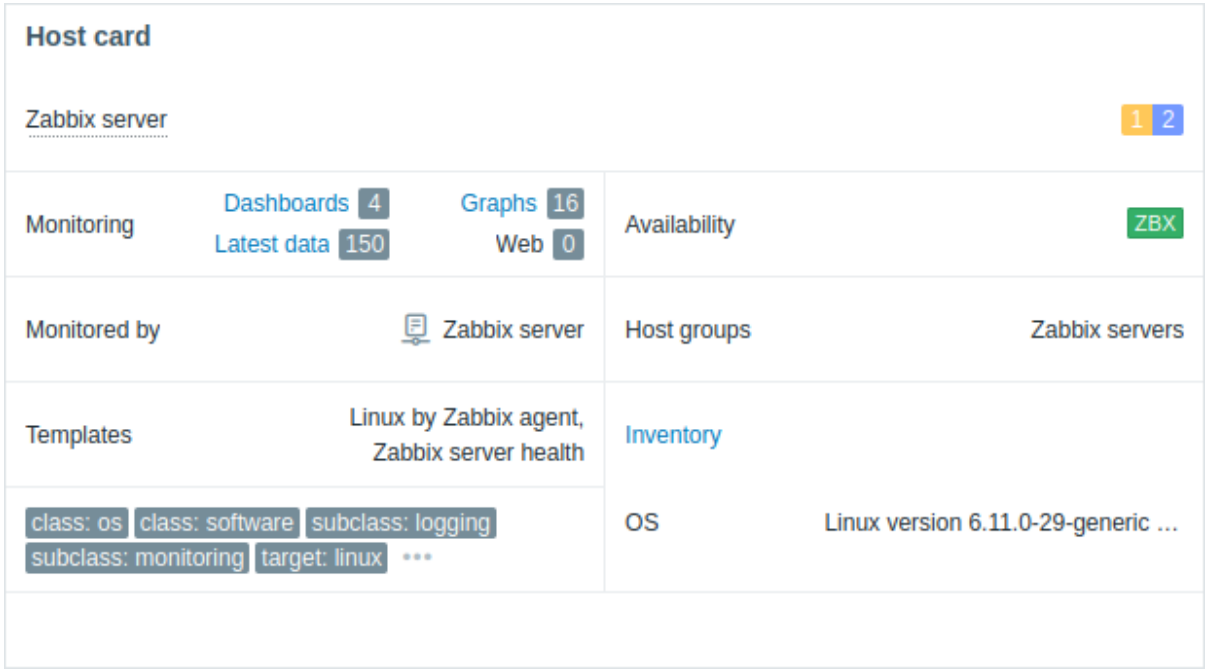
Además de los parámetros que son comunes para todos los widgets, puede establecer las siguientes opciones específicas:

Grupos de equipos	Seleccione grupos de equipos. Alternativamente, seleccione un widget compatible como fuente de datos para los grupos de equipos. Este campo es autocompletado, por lo que al comenzar a escribir el nombre de un grupo se mostrará un menú desplegable con los grupos coincidentes. Este parámetro no está disponible al configurar el widget en un panel de plantilla.
Tipo de interfaz	Seleccione para qué interfaces de equipos desea ver los datos de disponibilidad: *Agente Zabbix (chequeos activos)*, *Agente Zabbix (chequeos pasivos)*, *SNMP*, *JMX*, *IPMI*. Para *Agente Zabbix (chequeos activos)*, la celda *Mixto* siempre estará vacía ya que este tipo de ítems no puede tener múltiples interfaces. Por defecto, se muestra la disponibilidad de todas las interfaces si no se selecciona ninguna.
Disposición	Seleccione visualización horizontal (columnas) o visualización vertical (líneas).
Incluir equipos en mantenimiento	Incluye los equipos que están en mantenimiento en las estadísticas. Este parámetro se etiqueta como *Mostrar datos en mantenimiento* al configurar el widget en un panel de plantilla.
Mostrar solo totales	Si está marcado, se muestra el total de equipos, sin desglose por interfaces. Esta opción se desactiva si solo se selecciona una interfaz.

Tarjeta de host

Descripción general

El widget *Tarjeta de host* muestra los detalles de un único host, lo que le permite evaluar rápidamente el estado y la configuración de un host (disponibilidad, problemas, inventario, etc.).

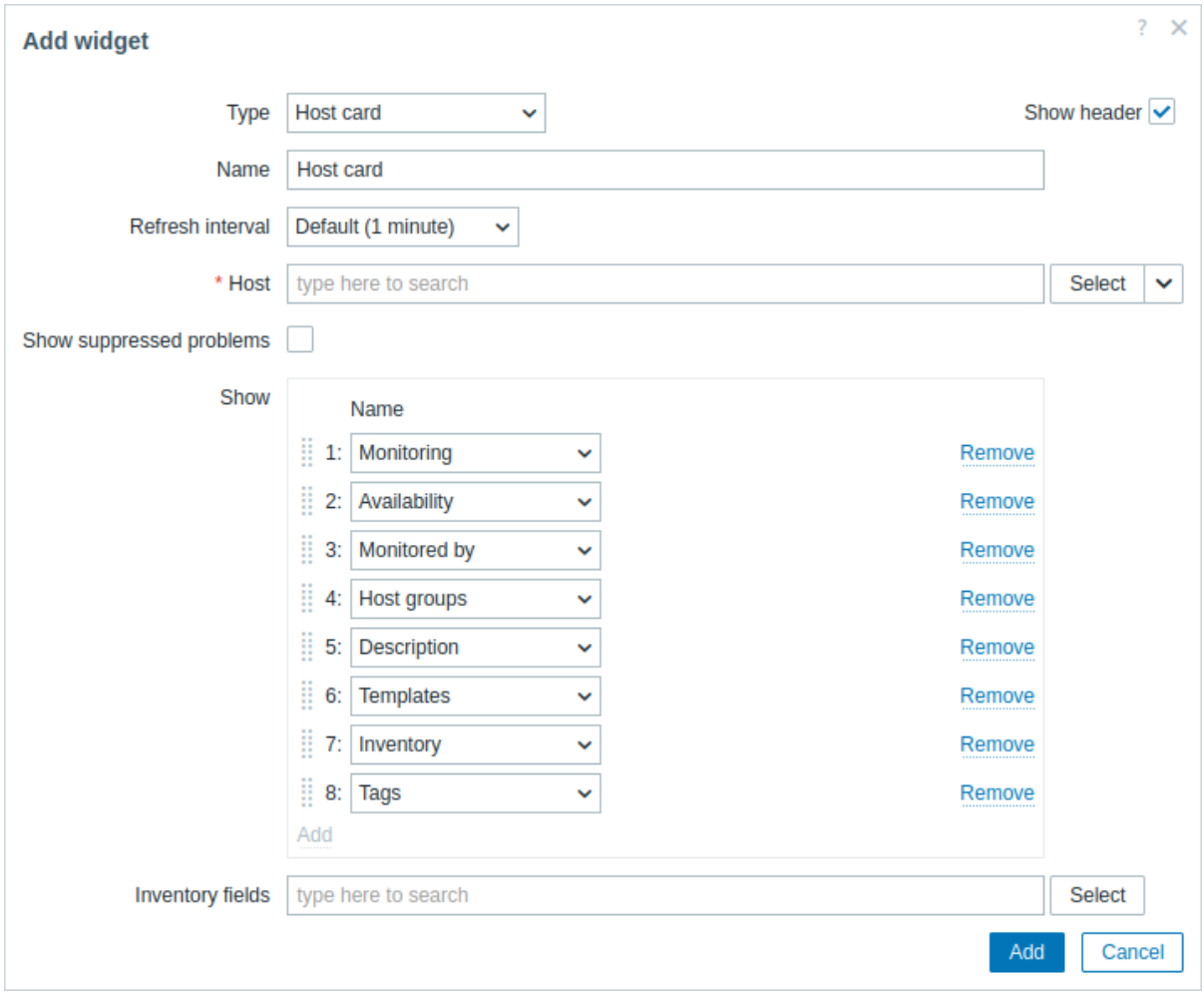


El host puede establecerse directamente en la configuración del widget o seleccionarse desde un widget compatible o desde el selector de host en el panel.

El widget puede mostrar varias secciones, cada una mostrando información diferente. El diseño de las secciones se ajusta automáticamente en función del ancho del widget; cuando se expande horizontalmente, las secciones se reorganizan automáticamente en varias columnas.

Configuración

Para configurar, seleccione *Tarjeta de equipo* como tipo:



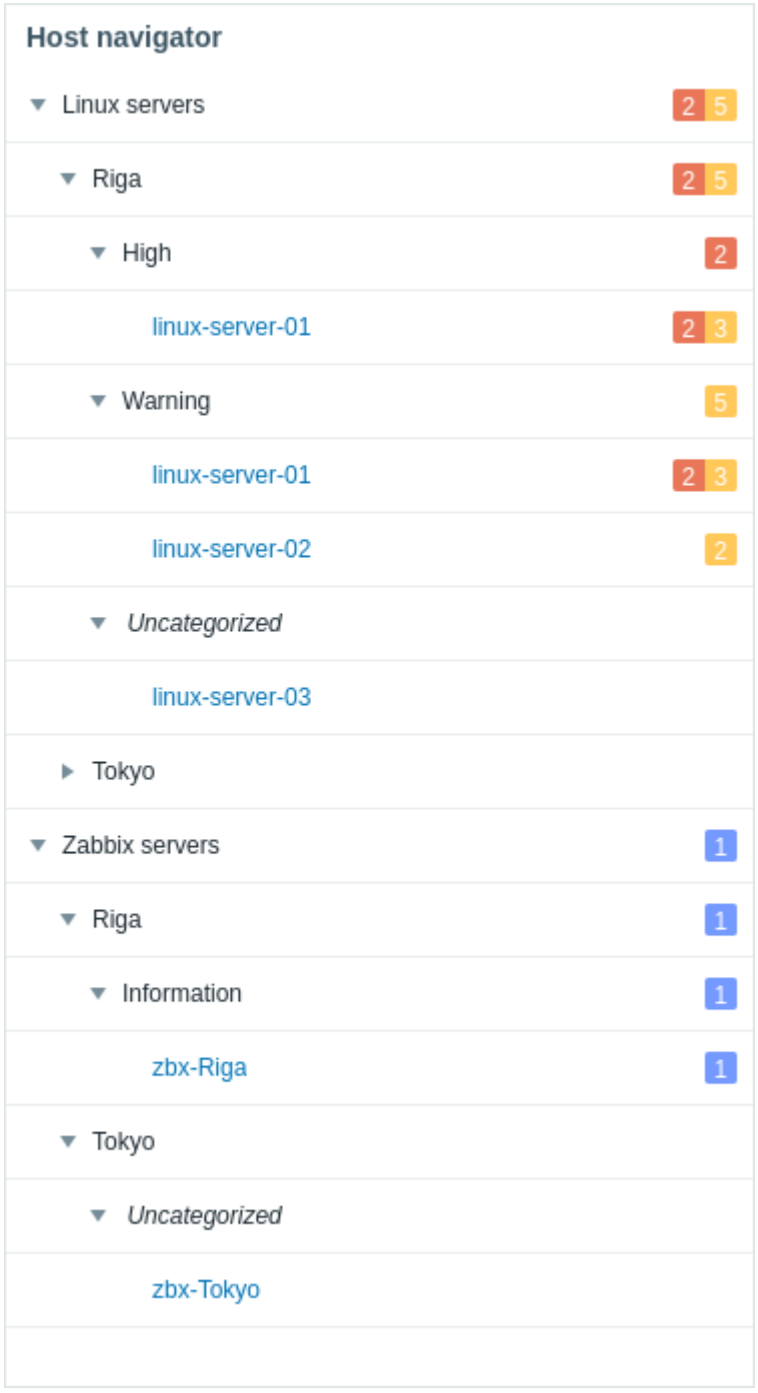
Además de los parámetros que son comunes para todos los widgets, puede establecer los siguientes parámetros específicos:

Equipo	Seleccione el equipo. Alternativamente, seleccione un widget compatible o el panel como fuente de datos para los equipos. Las sugerencias coincidentes aparecen a medida que escribe. La sección superior del widget siempre muestra el **nombre del equipo** (al hacer clic se abre el menú del equipo): - Si el equipo está deshabilitado, se muestra una etiqueta roja *Deshabilitado* junto al nombre del equipo. - Si el equipo está en mantenimiento, se muestra el icono de mantenimiento junto al nombre del equipo. La sección superior del widget también muestra el número de **problemas del equipo** agrupados por severidad (al hacer clic se abren los problemas del equipo). Este parámetro no está disponible al configurar el widget en un panel de plantilla.
Mostrar problemas suprimidos	Marque esta casilla para incluir los problemas suprimidos en el cálculo de problemas.
Mostrar	Agregue secciones para mostrar en el widget. Las secciones se pueden reorganizar arrastrando y soltando.
Grupos de equipos	Muestra los grupos de equipos a los que pertenece el equipo.
Descripción	Muestra la descripción del equipo.
Supervisión	Muestra enlaces rápidos para navegar a los Paneles del equipo, Últimos datos, Gráficas y Escenarios web. El número junto a cada enlace indica el número de entidades respectivas.
Disponibilidad	Muestra la disponibilidad del equipo por interfaz.
Supervisado por	Muestra si el equipo es supervisado por el servidor Zabbix o por un proxy o grupo de proxies específico.
Plantillas	Muestra las plantillas vinculadas al equipo.
Inventario	Muestra los campos de inventario del equipo.
Etiquetas	Muestra las etiquetas del equipo.
Campos de inventario	Seleccione los campos de inventario que se mostrarán. Las sugerencias coincidentes aparecen a medida que escribe. Si no se especifican campos de inventario, se mostrarán todos los campos de inventario rellenados. Este parámetro está disponible si *Mostrar* contiene *Inventario*.

Navegador de host

Descripción general

El widget *Navegador de equipos* muestra una lista de equipos basada en varias opciones de filtrado y agrupamiento.



Los grupos por los que se organizan los equipos pueden expandirse o contraerse.

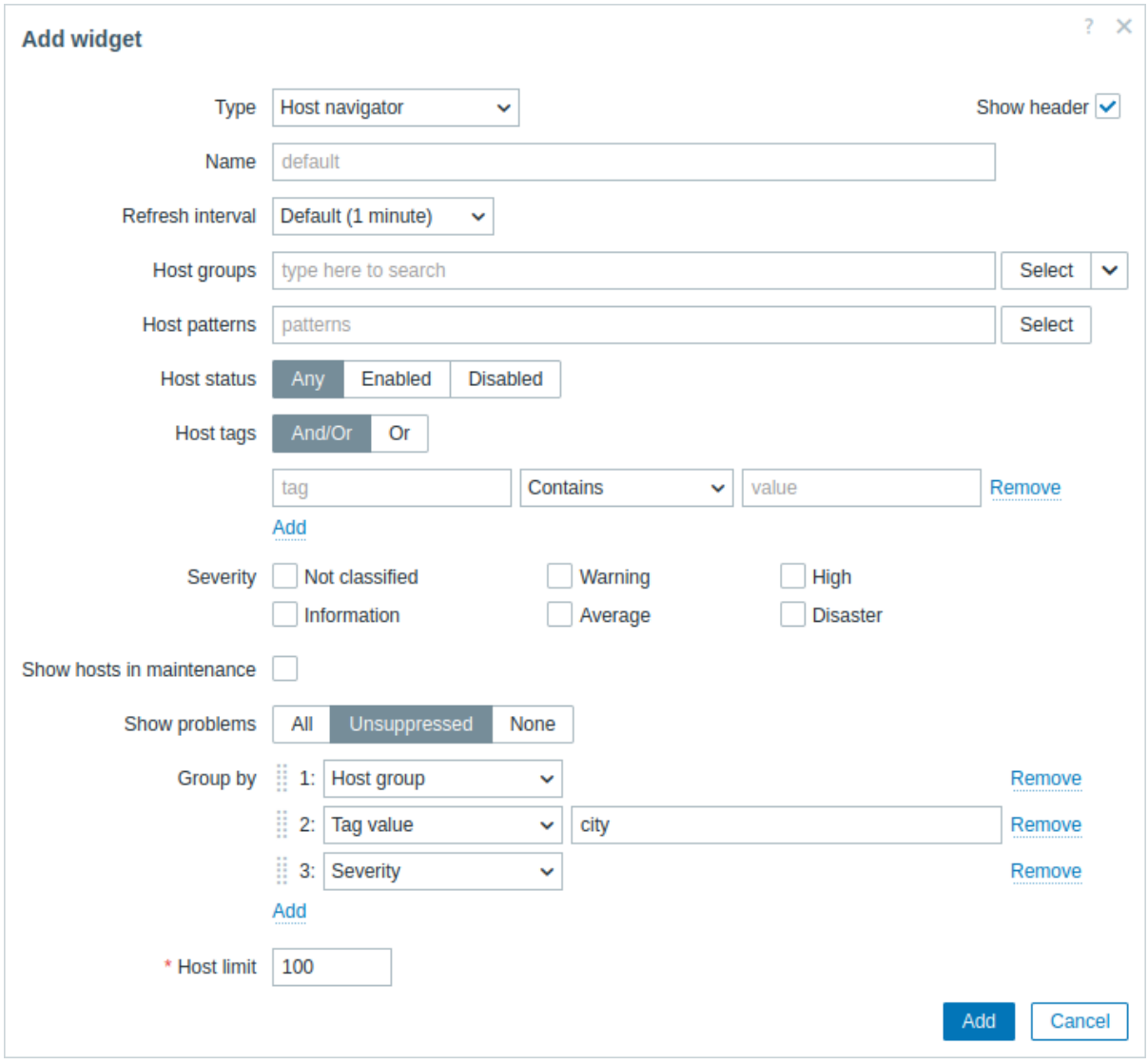
Para los grupos, problemas y equipos en mantenimiento, se puede acceder a detalles adicionales mediante sugerencias al pasar el ratón.

El widget es más útil para controlar lo que muestran otros widgets en función del equipo seleccionado.



Configuración

Para configurar, seleccione *Host navigator* como tipo:



Además de los parámetros comunes a todos los widgets, puede establecer las siguientes opciones específicas:

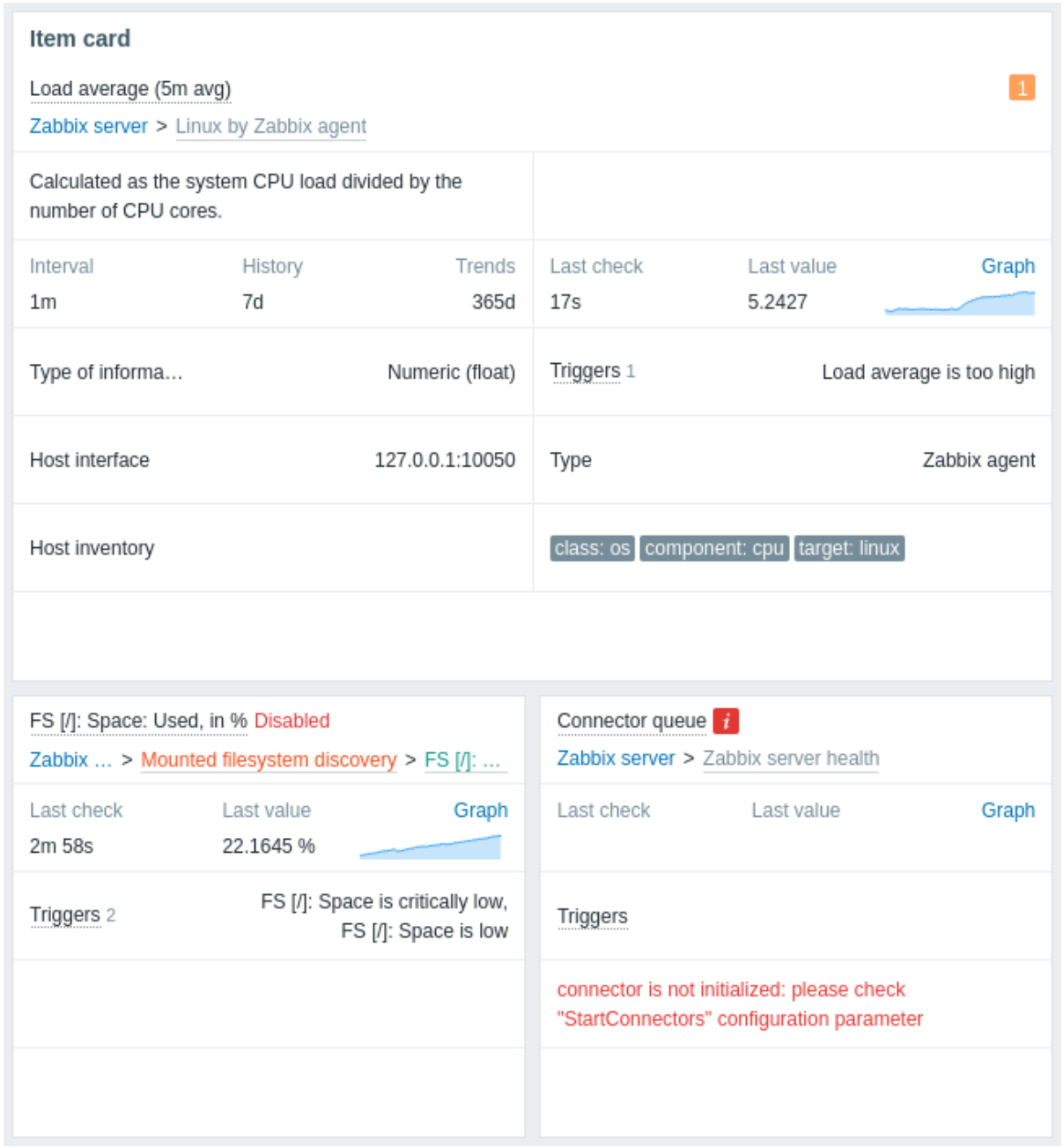
Host groups	Seleccione grupos de hosts. Como alternativa, seleccione un widget compatible como fuente de datos para los grupos de hosts. Este campo permite autocompletar; al comenzar a escribir el nombre de un grupo, se mostrará una lista desplegable con los grupos coincidentes. Al seleccionar un grupo de hosts principal, se seleccionan implícitamente todos los grupos de hosts anidados; si no se selecciona ningún grupo de hosts, el widget mostrará todos los hosts de todos los grupos de hosts. Este parámetro no está disponible al configurar el widget en un dashboard de template.
Host patterns	Introduzca patrones de hosts o seleccione hosts existentes como patrones de hosts. Los hosts que coincidan con los patrones especificados se mostrarán en el navegador de hosts. Este campo permite autocompletar; al comenzar a escribir el nombre de un host, se mostrará una lista desplegable con los hosts coincidentes. Si no se selecciona ningún host, el widget mostrará todos los hosts. Se pueden utilizar patrones comodín para la selección (por ejemplo, `*` devolverá los hosts que coincidan con cero o más caracteres; `Zabbix*` devolverá los hosts cuyo nombre comience por "Zabbix"). Para especificar un patrón comodín, introduzca la cadena manualmente y pulse *Enter*. Al comenzar a escribir, una lista desplegable mostrará los hosts coincidentes, limitada a los hosts pertenecientes a los hosts incluidos en los *Host groups* seleccionados, si los hay. Este parámetro no está disponible al configurar el widget en un dashboard de template.
Host status	Filtre los hosts que se mostrarán según su estado (cualquiera, habilitado, deshabilitado).
Host tags	Este parámetro no está disponible al configurar el widget en un dashboard de template. Especifique etiquetas para filtrar los hosts mostrados en el widget. Es posible incluir y excluir etiquetas y valores de etiquetas específicos. Se pueden establecer varias condiciones. La coincidencia de nombres de etiquetas siempre distingue entre mayúsculas y minúsculas. Hay varios operadores disponibles para cada condición: **Exists**: incluye los nombres de etiquetas especificados; **Equals**: incluye los nombres y valores de etiquetas especificados (distingue entre mayúsculas y minúsculas); **Contains**: incluye los nombres de etiquetas especificados cuyos valores contienen la cadena introducida (coincidencia de subcadena, sin distinguir entre mayúsculas y minúsculas); **Does not exist**: excluye los nombres de etiquetas especificados; **Does not equal**: excluye los nombres y valores de etiquetas especificados (distingue entre mayúsculas y minúsculas); **Does not contain**: excluye los nombres de etiquetas especificados cuyos valores contienen la cadena introducida (coincidencia de subcadena, sin distinguir entre mayúsculas y minúsculas). Hay dos tipos de cálculo para las condiciones: **And/Or**: deben cumplirse todas las condiciones; las condiciones que tienen el mismo nombre de etiqueta se agruparán mediante la condición Or; **Or**: basta con que se cumpla una condición.
Severity	Este parámetro no está disponible al configurar el widget en un dashboard de template. Marque las gravedades de los problemas para filtrar los hosts con problemas que se mostrarán en el widget. Si no se marca ninguna gravedad, se mostrarán todos los hosts con todos los problemas.
Show hosts in maintenance	Marque esta casilla para mostrar los hosts en mantenimiento (en este caso, se mostrará un icono de mantenimiento junto al nombre del host). Este parámetro se denomina *Show data in maintenance* al configurar el widget en un dashboard de template.

Show problems	Filtre los problemas que se mostrarán según su estado (todos, no suprimidos, ninguno). Los problemas se indican mediante bloques de colores al final de la fila del host. Los colores se basan en la gravedad del problema, que se puede personalizar en *Administration* > *General* > *Trigger displaying options*. Las gravedades de los problemas se pueden cambiar al actualizar problemas.
Group by	Añada un atributo de agrupación por el que se agruparán los hosts seleccionados: **Host group**: agrupa los hosts por su grupo de hosts; **Tag value**: introduzca un nombre de etiqueta para agrupar los hosts según los valores de esta etiqueta (por ejemplo, introduzca ”city” para agrupar los hosts por los valores ”Riga”, ”Tokyo”, etc.); **Severity**: agrupa los hosts por la gravedad de sus problemas. Si *Show problems* está configurado para mostrar problemas, estos se mostrarán de la siguiente manera: - para cada grupo de gravedad, solo se muestra el recuento de problemas correspondiente; - para cada host, se muestran todos sus recuentos de problemas. Tenga en cuenta que los hosts se agruparán únicamente por las gravedades marcadas en el parámetro *Severity*; si no se marca ninguna gravedad, todos los hosts se agruparán por todas las gravedades. Los atributos de agrupación se pueden reordenar arrastrándolos hacia arriba o hacia abajo mediante el controlador situado delante del nombre del grupo. Tenga en cuenta que el orden de los atributos de agrupación determina el orden de anidamiento de los grupos. Por ejemplo, especificar varios nombres de etiquetas (1: color, 2: city) hará que los hosts se agrupen por color (red, blue, etc.) y, a continuación, por ciudad (Riga, Tokyo, etc.). Un host puede mostrarse en varios grupos según los atributos de agrupación configurados (por ejemplo, al agrupar por grupo de hosts cuando el host pertenece a varios grupos de hosts). Al hacer clic en dichos hosts, se seleccionan y resaltan en todos los grupos. Los hosts que no coincidan con los atributos de agrupación configurados se muestran en el grupo *Uncategorized*. Se pueden especificar hasta 10 atributos de agrupación, y todos deben ser únicos. Si no se especifica ningún atributo de agrupación, los hosts no se agruparán.
Host limit	Introduzca el número máximo de hosts que se mostrarán. Los valores posibles van de 1 a 9999. Cuando haya más hosts disponibles para mostrar que el límite establecido, se mostrará un mensaje correspondiente debajo de los hosts mostrados (por ejemplo, ”100 of 100+ hosts are shown”). Tenga en cuenta que el límite de hosts configurado también afecta a la visualización de los grupos configurados; por ejemplo, si el límite de hosts está establecido en 100 y los hosts se agrupan por valores de etiquetas (más de 200), solo se mostrarán en el widget los primeros 100 valores de etiquetas con los hosts correspondientes. Este parámetro no se ve afectado por el parámetro *Limit for search and filter results* en *Administration* > *General* > *GUI*. Este parámetro no está disponible al configurar el widget en un dashboard de template.

Tarjeta de item

Descripción general

El widget *Tarjeta de elemento* muestra los detalles de un único elemento, lo que le permite evaluar rápidamente el estado y la configuración de un elemento (datos más recientes, errores, descripción, etc.).

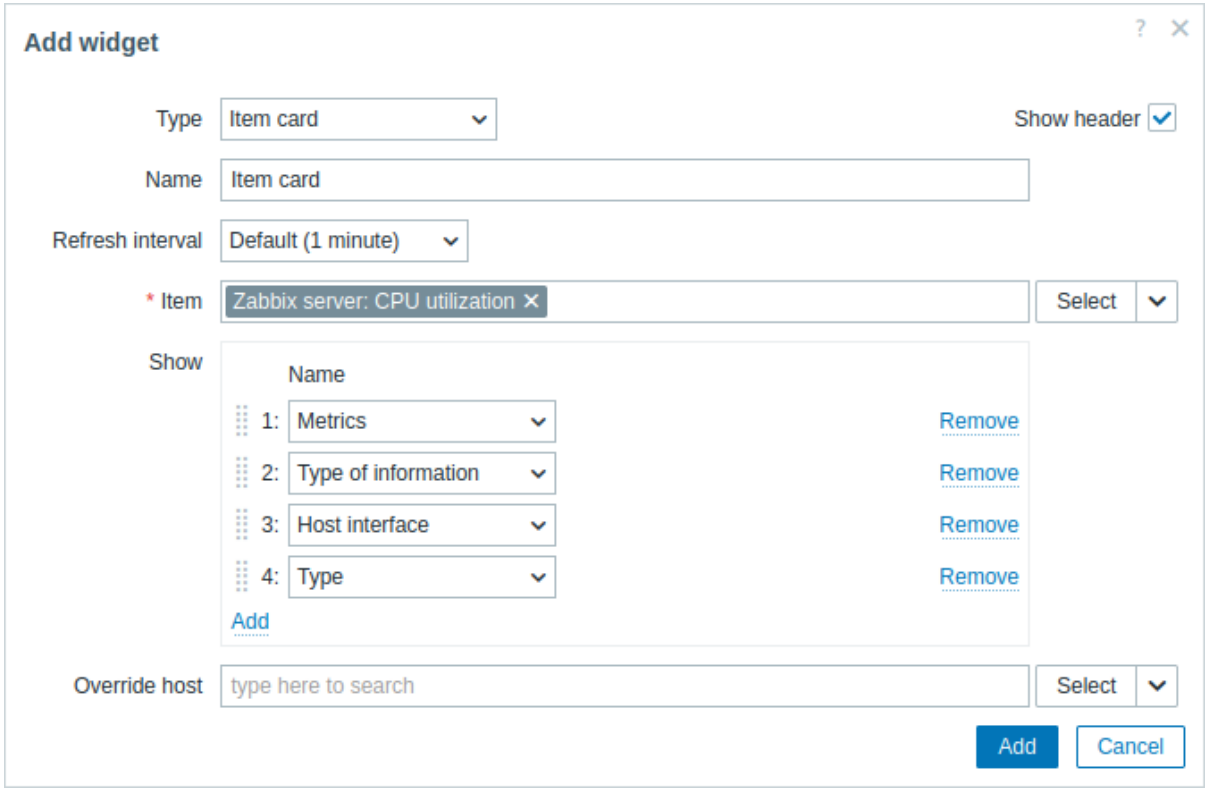


El elemento puede establecerse directamente en la configuración del widget o seleccionarse desde un widget compatible en el panel.

El widget puede mostrar varias secciones, cada una mostrando información diferente. El diseño de las secciones se ajusta automáticamente en función del ancho del widget; cuando se expande horizontalmente, las secciones se reorganizan automáticamente en varias columnas.

Configuración

Para configurar, seleccione *Tarjeta de elemento* como tipo:



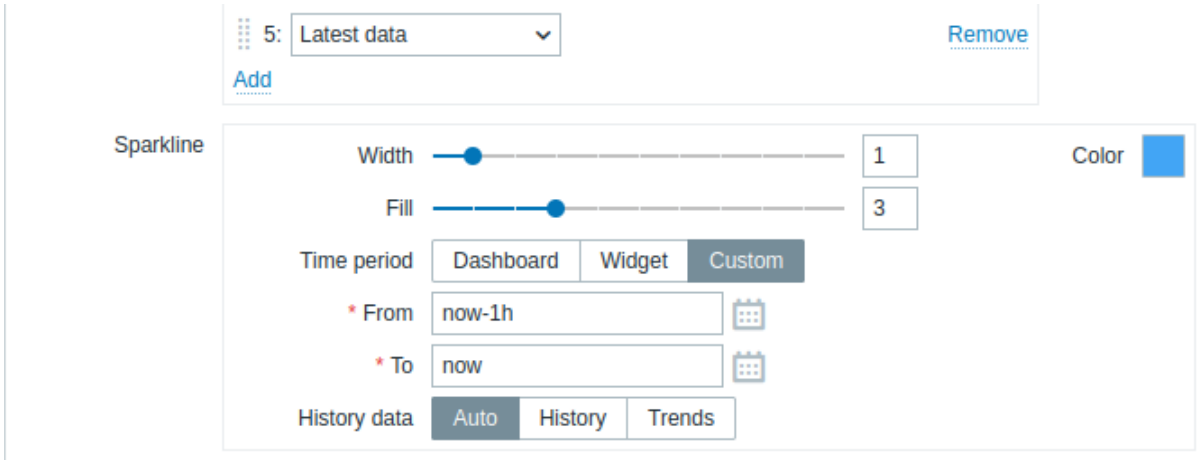
Además de los parámetros que son comunes para todos los widgets, puede establecer las siguientes opciones específicas:

Elemento	Seleccione el elemento. Alternativamente, seleccione un widget compatible como fuente de datos para los elementos. Las sugerencias coincidentes aparecen a medida que escribe. La sección superior del widget siempre muestra el **nombre del elemento** (al hacer clic se abre el menú del elemento): - Si el elemento está deshabilitado, se muestra una etiqueta roja *Deshabilitado* junto al nombre del elemento. - En caso de errores, se muestra un icono cuadrado con la letra "i". En caso de problemas, se muestra un icono cuadrado con el recuento de problemas para cada gravedad de problema. Al pasar el cursor sobre un icono se abre una ventana emergente con detalles. - En un panel de plantilla, se muestra el nombre técnico (en lugar del nombre visible) del elemento. La sección superior del widget también muestra la **ruta del elemento**—su equipo y, si corresponde, plantilla, regla de bajo nivel (LLD) o elemento maestro (al hacer clic en una entidad se abre su configuración). La ruta del elemento solo se muestra a los usuarios con permisos para las entidades respectivas.
Mostrar	Agregue secciones para mostrar en el widget. Las secciones se pueden reorganizar arrastrando y soltando.
Descripción	Muestra la descripción del elemento. Las macros se resuelven.
Texto de error	Muestra el error del elemento (si lo hay).
Métricas	Muestra la siguiente información: - Intervalo de actualización del elemento (excepto para elementos trapper); - Configuración de historial del elemento (si la hay); - Configuración de tendencias del elemento (si la hay). Las macros se resuelven. Si el elemento utiliza intervalos personalizados, se muestra un icono de ayuda junto al intervalo de actualización del elemento. Al pasar el cursor sobre el icono se abre una ventana emergente con detalles del intervalo personalizado.

Últimos datos	Muestra la siguiente información: - Tiempo desde la última comprobación del elemento; - Último valor del elemento; - Enlace a gráfico simple/historial de los valores del elemento (si se almacenan historial o tendencias); - Gráfico sparkline configurable (si el elemento es numérico). Para elementos de tipo binario, se muestra una miniatura o la opción *Mostrar* en lugar del último valor. Al pasar el cursor sobre la miniatura se abre una ventana emergente con la imagen. Al pasar el cursor sobre *Mostrar* se abre una ventana emergente con el valor del elemento (cadena Base64). En un panel de plantilla, se muestra *Sin datos* en lugar del último valor del elemento.
Tipo de información	Muestra el tipo de información del elemento.
Disparadores	Muestra la siguiente información: - Etiqueta *Disparadores* con el recuento de disparadores (al pasar el cursor sobre *Disparadores* se abre una ventana emergente con detalles de los disparadores); - Lista de nombres de disparadores (las macros se resuelven).
Interfaz del equipo	Muestra la interfaz del equipo utilizada por el elemento, o *Sin datos* si el elemento no tiene una interfaz de equipo (por ejemplo, elementos dependientes, comprobaciones simples). En un panel de plantilla, se muestra *Sin datos*.
Tipo	Muestra el tipo de elemento.
Inventario del equipo	Muestra el campo de inventario del equipo que el valor del elemento completa.
Etiquetas	Muestra las etiquetas del elemento.
Anular equipo	Seleccione un widget compatible o el selector de equipos del panel como fuente de datos para los equipos. Este parámetro no está disponible al configurar el widget en un panel de plantilla.

Sparkline

Las opciones de configuración del gráfico Sparkline solo están disponibles si la sección *Últimos datos* se añade al widget.



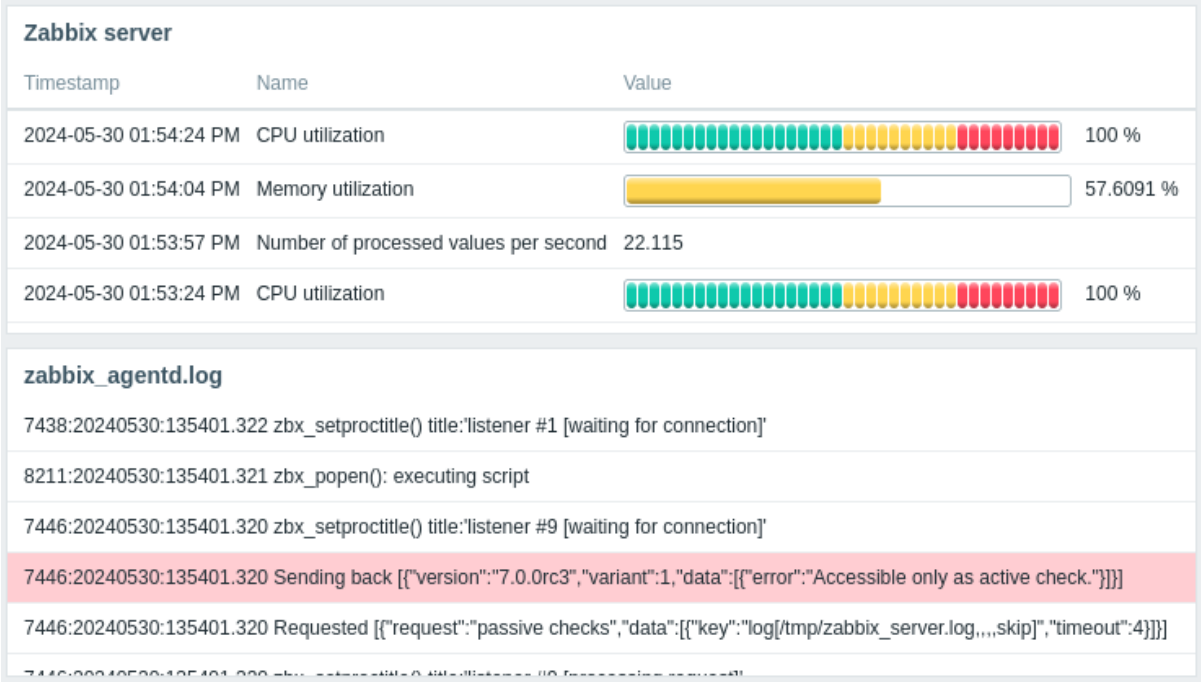
Ancho	Establezca el grosor de la línea del gráfico utilizando el control deslizante o introduciendo manualmente un valor en el rango de 0 a 10.
Color	Seleccione el color de la línea y del relleno.
Relleno	Establezca el nivel de transparencia del color de relleno utilizando el control deslizante o introduciendo manualmente un valor en el rango de 0 a 10.
Período de tiempo	Seleccione la fuente de datos para el período de tiempo: **Panel**: utilice el selector de período de tiempo del panel; **Widget**: utilice un widget compatible (establecido en el parámetro *Widget*); **Personalizado**: utilice un período de tiempo personalizado establecido en los parámetros *Desde* y *Hasta*; si se establece, se mostrará un icono de reloj en la esquina superior derecha del widget, indicando el tiempo configurado al pasar el ratón. Tenga en cuenta que, independientemente de la configuración del *Período de tiempo* del widget, los widgets compatibles aún pueden utilizarlo como fuente de datos para el período de tiempo.

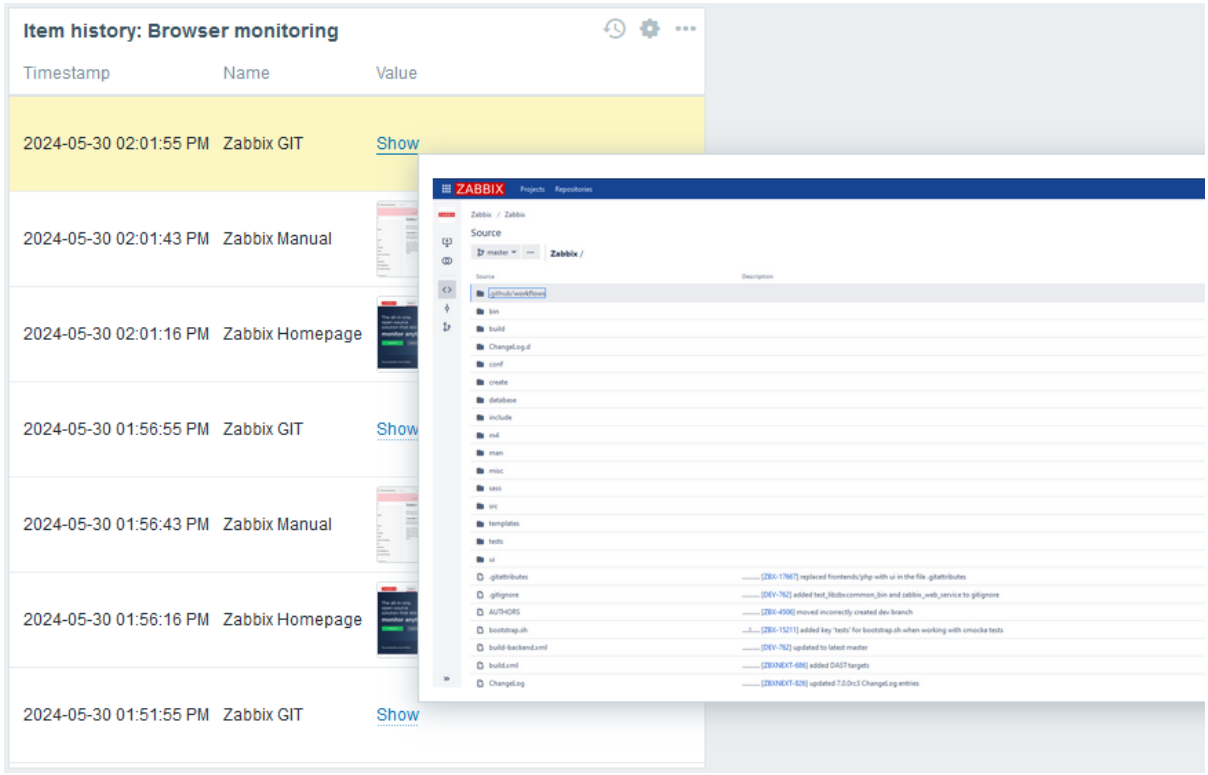
Widget	Introduzca o seleccione un widget compatible como fuente de datos para el período de tiempo. Este parámetro está disponible si *Período de tiempo* está configurado en "Widget".
Desde	Introduzca o seleccione el inicio del período de tiempo. Se admite la sintaxis de tiempo relativo (`now`, `now/d`, `now/w-1w`, etc.). Este parámetro está disponible si *Período de tiempo* está configurado en "Personalizado".
Hasta	Introduzca o seleccione el final del período de tiempo. Se admite la sintaxis de tiempo relativo (`now`, `now/d`, `now/w-1w`, etc.). Este parámetro está disponible si *Período de tiempo* está configurado en "Personalizado".
Datos históricos	Tome los datos del historial o de las tendencias: **Auto**: selección automática; **Historial**: tomar datos del historial; **Tendencias**: tomar datos de tendencias. Este parámetro solo se aplica a datos numéricos. Los datos no numéricos siempre se tomarán del historial.

Historial de item

Descripción general

El widget *Item history* muestra los datos más recientes de todos los tipos de item (numérico, texto, etc.) en formato de tabla. Puede mostrar barras de progreso e imágenes (útil para browser items) y puede resaltar valores (útil para monitorización de archivos de registro).

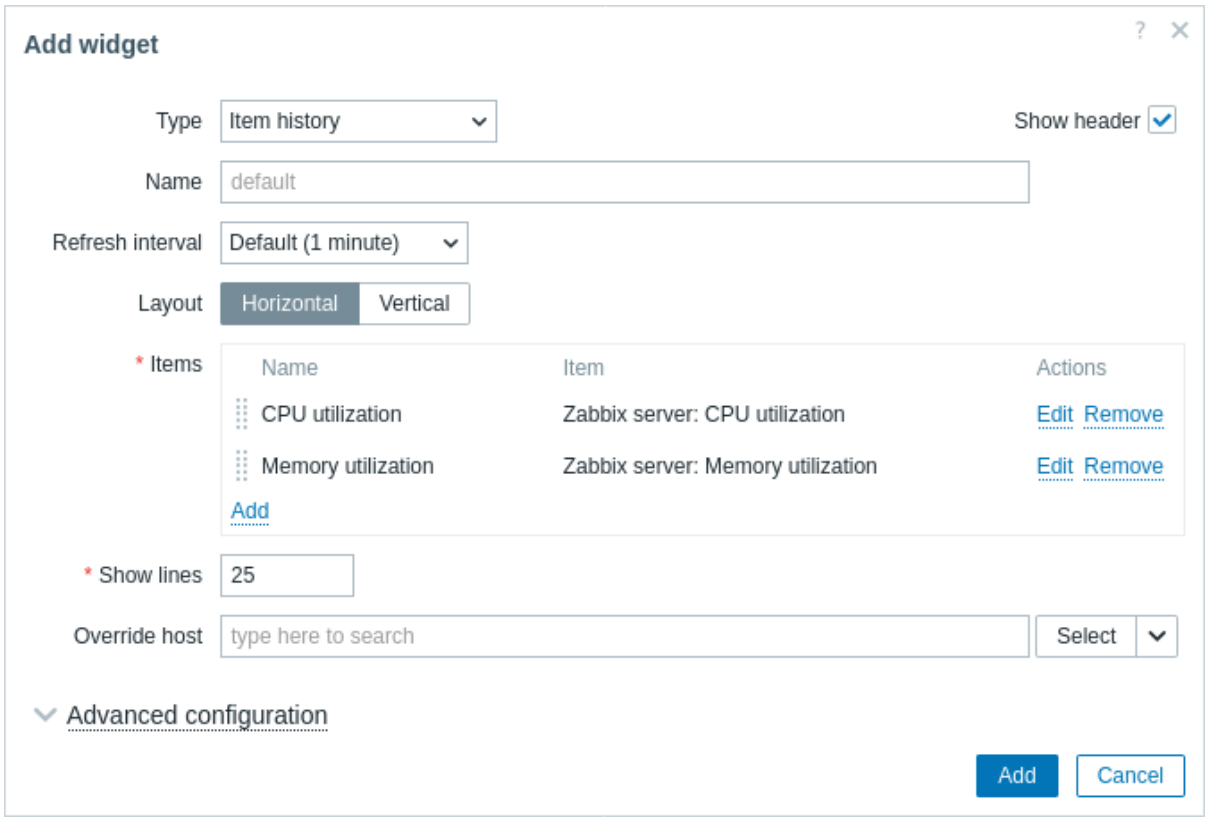




Se pueden mostrar hasta 1000 registros.

Configuración

Para configurar, seleccione *Historial de ítem* como tipo:



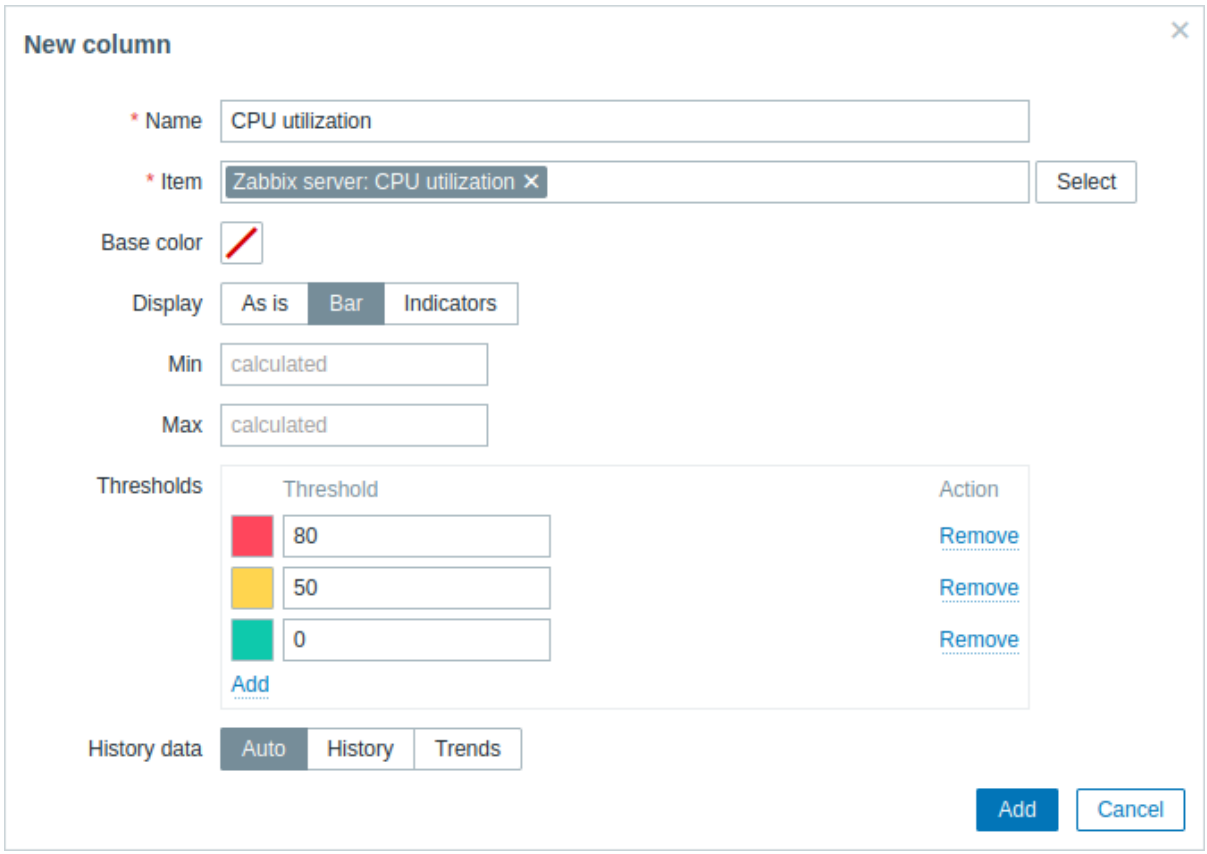
Además de los parámetros que son comunes para todos los widgets, puede establecer las siguientes opciones específicas:

Disposición	Seleccione la opción de disposición para las columnas de ítems: **Horizontal**: los ítems se mostrarán horizontalmente, los valores verticalmente; **Vertical**: los ítems se mostrarán verticalmente, los valores horizontalmente.

Ítems	Agregue columnas de ítems para mostrar. El orden de los ítems determina su orden de visualización. Los ítems pueden reordenarse arrastrándolos hacia arriba o hacia abajo mediante el asa antes del nombre del ítem.
Mostrar líneas	Especifique el número de líneas de valores de ítem a mostrar.
Anular equipo	Seleccione un widget compatible o el selector de equipos del panel como fuente de datos para los equipos. Este parámetro no está disponible al configurar el widget en un panel de plantilla.
Configuración avanzada	Haga clic en la etiqueta *Configuración avanzada* para mostrar las opciones de configuración avanzada.

Configuración de columnas

Para configurar las columnas de métricas, haga clic en *Añadir* en el parámetro *Métricas*:



Parámetros comunes de columna:

Nombre	Introduzca el nombre de la columna. Si se deja vacío, se utiliza el nombre de la métrica del parámetro *Métrica*.
Métrica	Seleccione la métrica. Tenga en cuenta que los parámetros de configuración de la columna varían según el tipo de información de la métrica seleccionada; para más información, consulte los parámetros individuales a continuación. Al configurar el widget en un panel de plantilla, solo se pueden seleccionar métricas configuradas en la plantilla.
Color base	Seleccione el color de fondo de la columna o el color de relleno si *Visualización* está configurado en ”Barra” o ”Indicadores”. Tenga en cuenta que el color base puede ser sobrescrito por colores de umbral o de resaltado.

Parámetros de columna específicos para métricas de tipo numérico:

Mostrar	Seleccione cómo se debe mostrar el valor de la métrica: **Tal cual** - como texto normal; **Barra** - como una barra sólida y rellena de color; **Indicadores** - como una barra segmentada y rellena de color.
Mín	Introduzca el valor mínimo para barra/indicadores. Si se deja vacío, el widget usará el valor mínimo de la métrica. Este parámetro solo está disponible cuando *Mostrar* está configurado en ”Barra” o ”Indicadores”.
Máx	Introduzca el valor máximo para barra/indicadores. Si se deja vacío, el widget usará el valor máximo de la métrica. Este parámetro solo está disponible cuando *Mostrar* está configurado en ”Barra” o ”Indicadores”.
Umbrales	Haga clic en *Añadir* para agregar un umbral, seleccione un color de umbral en el selector de color y especifique un valor numérico. La lista de umbrales se ordenará en orden ascendente al guardar. Se admiten Sufijos (por ejemplo, ”1d”, ”2w”, ”4K”, ”8G”). Se admiten Mapeos de valores.
Datos históricos	Seleccione si desea tomar datos del histórico o de tendencias: **Auto** - selección automática; **Histórico** - tomar datos del histórico; **Tendencias** - tomar datos de tendencias.

Parámetros de columna específicos para items de tipo carácter, texto, log y JSON:

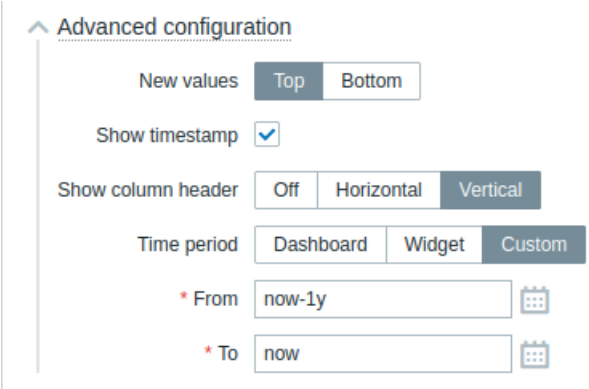
Resaltados	Haga clic en *Add* para agregar un resaltado, seleccione un color de resaltado en el selector de color y especifique una expresión regular. El color seleccionado se usará como color de fondo para los valores del item donde la expresión regular especificada coincida con el texto.
Mostrar	Seleccione cómo debe mostrarse el valor del item: **Tal cual** - exactamente como se recibe. Los saltos de línea se conservan si el valor contiene retornos de línea. El ajuste de línea se aplica si *Layout* está configurado en ”Vertical” y hay más de una columna configurada; **HTML** - como texto con formato HTML; **Una sola línea** - como una sola línea, truncada a una longitud especificada (1-500 caracteres). Al pasar el cursor o hacer clic en el valor truncado, se abre una ventana emergente con el valor completo.
Usar fuente monoespaciada	Marque esta casilla para mostrar el valor del item en fuente monoespaciada (desmarcada de forma predeterminada).
Mostrar hora local	Marque esta casilla para mostrar la hora local en lugar de la marca de tiempo en la columna de marca de tiempo. Tenga en cuenta que la casilla *Show timestamp* en la configuración avanzada también debe estar marcada. Este parámetro solo está disponible para items de tipo log.

Parámetros de columna específicos para métricas de tipo binario:

Mostrar miniatura	Marque esta casilla para mostrar una miniatura para binarios de imagen o una opción ”Mostrar” para binarios que no sean imágenes. Desmarque esta casilla para mostrar una opción ”Mostrar” para todos los valores de métricas binarias. Al pasar el ratón o hacer clic en la opción ”Mostrar” se abre una ventana emergente con el valor de la métrica (imagen o cadena Base64). Si el valor de la métrica es una cadena vacía, se muestra la opción ”Mostrar”; al pasar el ratón o hacer clic en ella se abre una ventana emergente que contiene ”Cadena vacía”.

Configuración avanzada

Las opciones de configuración avanzada están disponibles en la sección plegable *Configuración avanzada*:

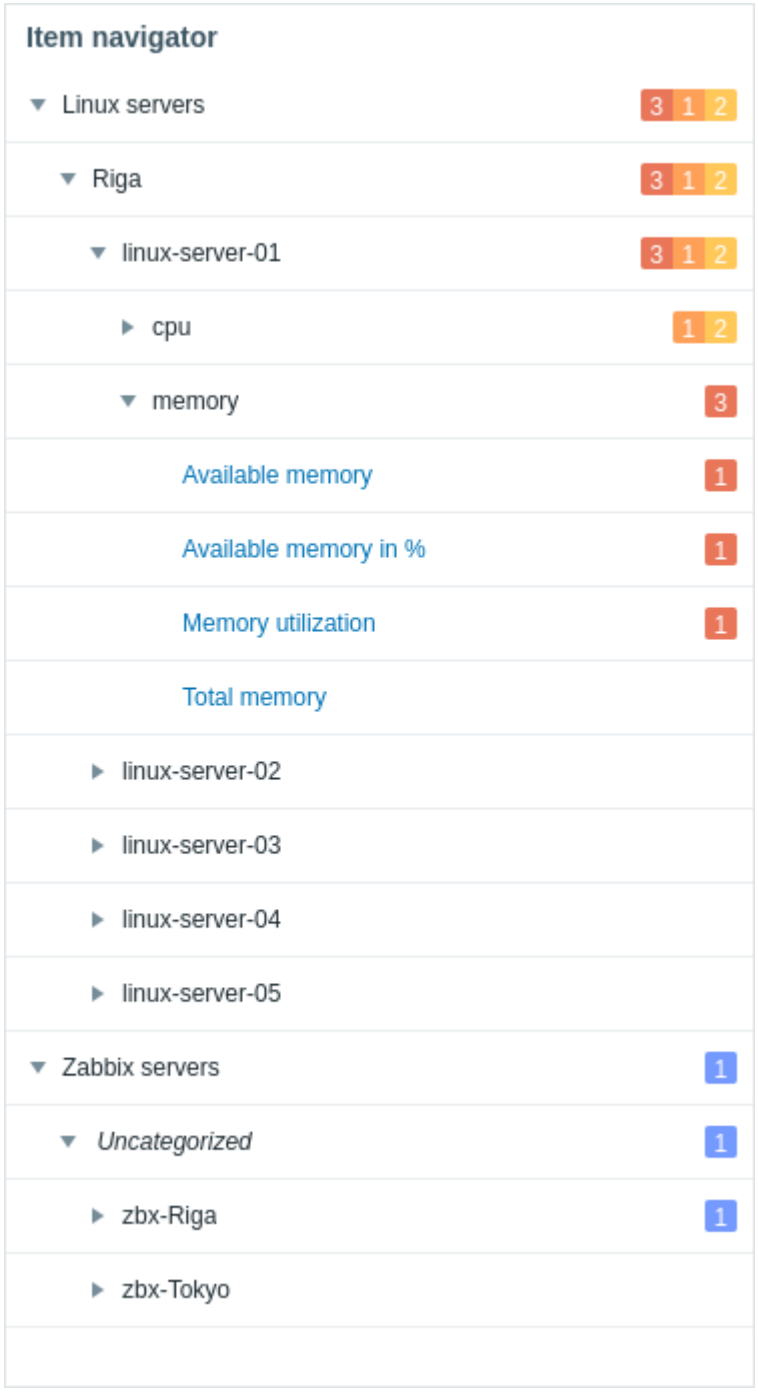


Nuevos valores	Seleccione dónde se deben agregar los nuevos valores del elemento: **Arriba**: en la parte superior de las columnas; **Abajo**: en la parte inferior de las columnas.
Mostrar marca de tiempo	Marque esta casilla para mostrar la columna de marca de tiempo (desmarcada por defecto).
Mostrar encabezado de columna	Seleccione la orientación del encabezado de columna: **Desactivado**: ocultar el encabezado; **Horizontal**: mostrar el encabezado horizontalmente; **Vertical**: mostrar el encabezado verticalmente.
Período de tiempo	Seleccione la fuente de datos para el período de tiempo: **Panel**: usar el selector de período de tiempo del panel; **Widget**: usar un widget compatible (establecido en el parámetro *Widget*); **Personalizado**: usar un período de tiempo personalizado establecido en los parámetros *Desde* y *Hasta*; si se establece, se mostrará un icono de reloj en la esquina superior derecha del widget, indicando el tiempo establecido al pasar el ratón. Tenga en cuenta que, independientemente de la configuración de *Período de tiempo* del widget, los widgets compatibles aún pueden usarlo como fuente de datos para el período de tiempo.
Widget	Introduzca o seleccione un widget compatible (*Gráfico*, *Gráfico (clásico)*, *Prototipo de gráfico*) como fuente de datos para el período de tiempo. Este parámetro está disponible si *Período de tiempo* está establecido en "Widget".
Desde	Introduzca o seleccione el inicio del período de tiempo. Se admite la sintaxis de tiempo relativo (`now`, `now/d`, `now/w-1w`, etc.). Este parámetro está disponible si *Período de tiempo* está establecido en "Personalizado".
Hasta	Introduzca o seleccione el final del período de tiempo. Se admite la sintaxis de tiempo relativo (`now`, `now/d`, `now/w-1w`, etc.). Este parámetro está disponible si *Período de tiempo* está establecido en "Personalizado".

Navegador de item

Descripción general

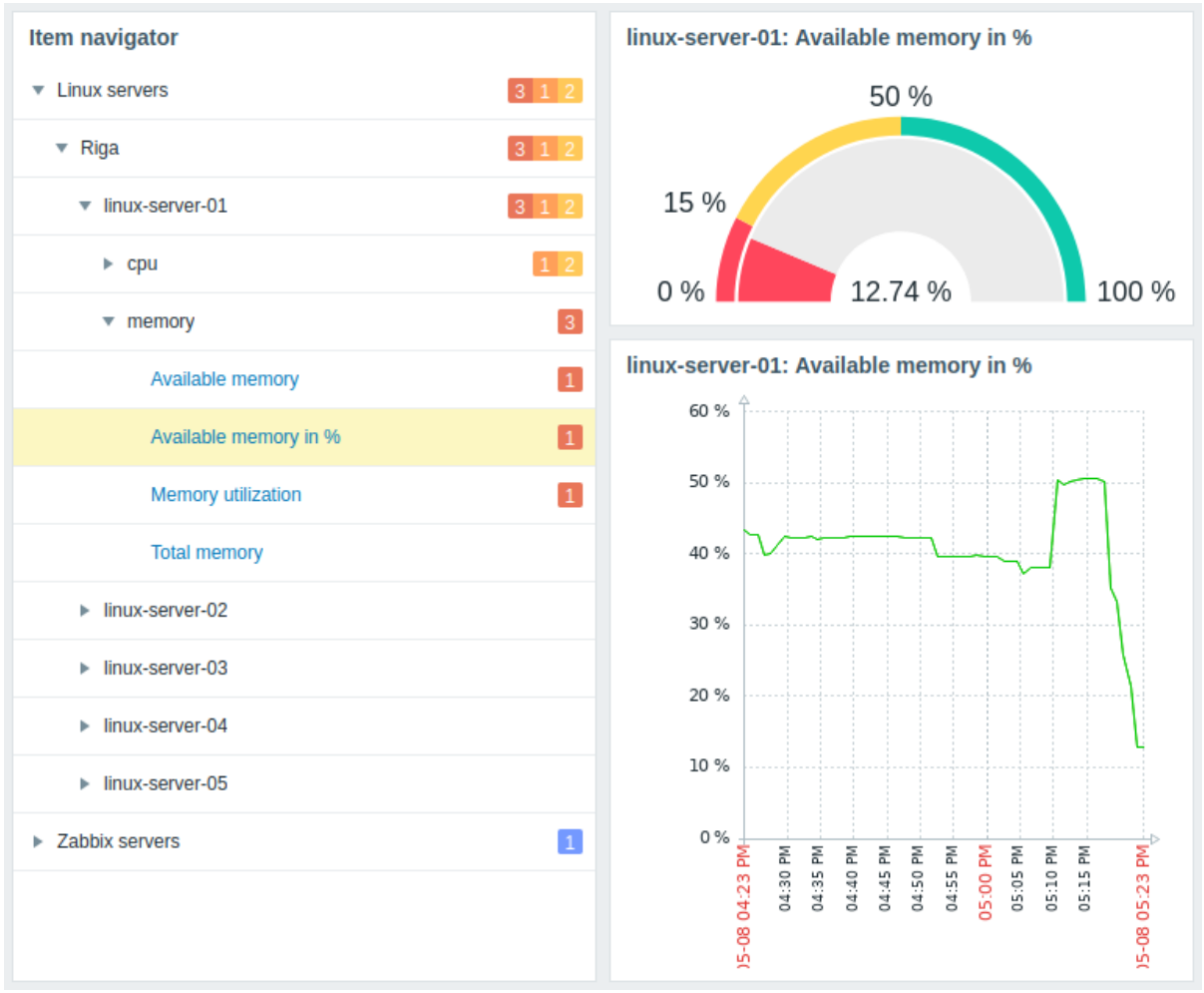
El widget *Navegador de elementos* muestra una lista de elementos basada en varias opciones de filtrado y agrupamiento.



Los grupos por los que se organizan los elementos pueden expandirse o contraerse.

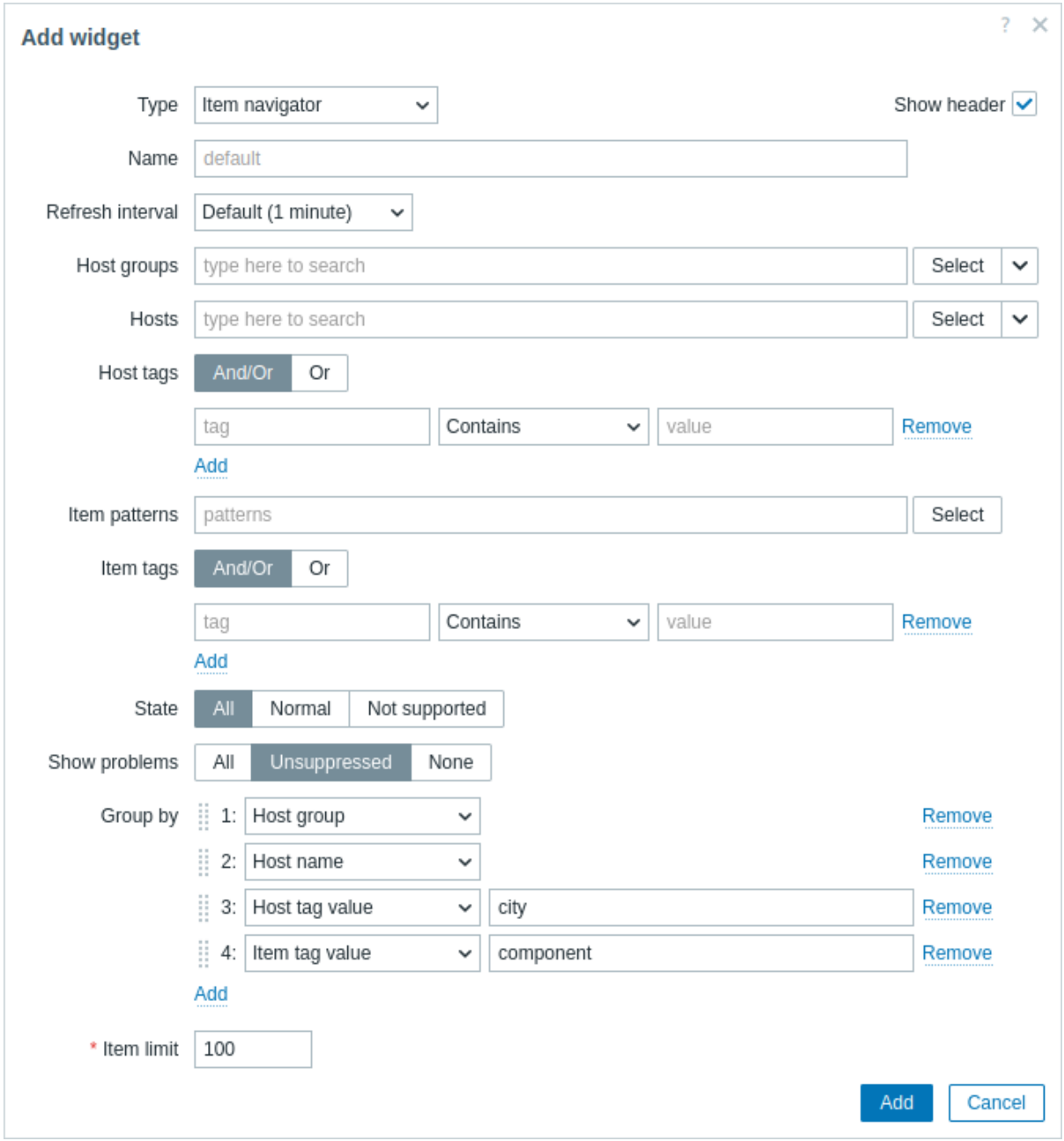
Para los grupos y problemas, se puede acceder a detalles adicionales mediante sugerencias al pasar el ratón.

El widget es más útil para controlar lo que muestran otros widgets en función del elemento seleccionado.



Configuración

Para configurar, seleccione *Navegador de items* como tipo:



Además de los parámetros comunes a todos los widgets, puede establecer las siguientes opciones específicas:

Grupos de hosts	Seleccione grupos de hosts. Como alternativa, seleccione un widget compatible como fuente de datos para los grupos de hosts. Este campo ofrece autocompletado; al comenzar a escribir el nombre de un grupo, se mostrará una lista desplegable con los grupos coincidentes. Al seleccionar un grupo de hosts principal, se seleccionan implícitamente todos los grupos de hosts anidados; si no se selecciona ningún grupo de hosts, el widget mostrará los items pertenecientes a todos los hosts de todos los grupos de hosts. Este parámetro no está disponible al configurar el widget en un dashboard de template.
Hosts	Seleccione hosts. Como alternativa, seleccione un widget compatible o el dashboard como fuente de datos para los hosts. Este campo ofrece autocompletado; al comenzar a escribir el nombre de un host, se mostrará una lista desplegable con los hosts coincidentes. Si no se selecciona ningún host, el widget mostrará los items pertenecientes a todos los hosts. Este parámetro no está disponible al configurar el widget en un dashboard de template.

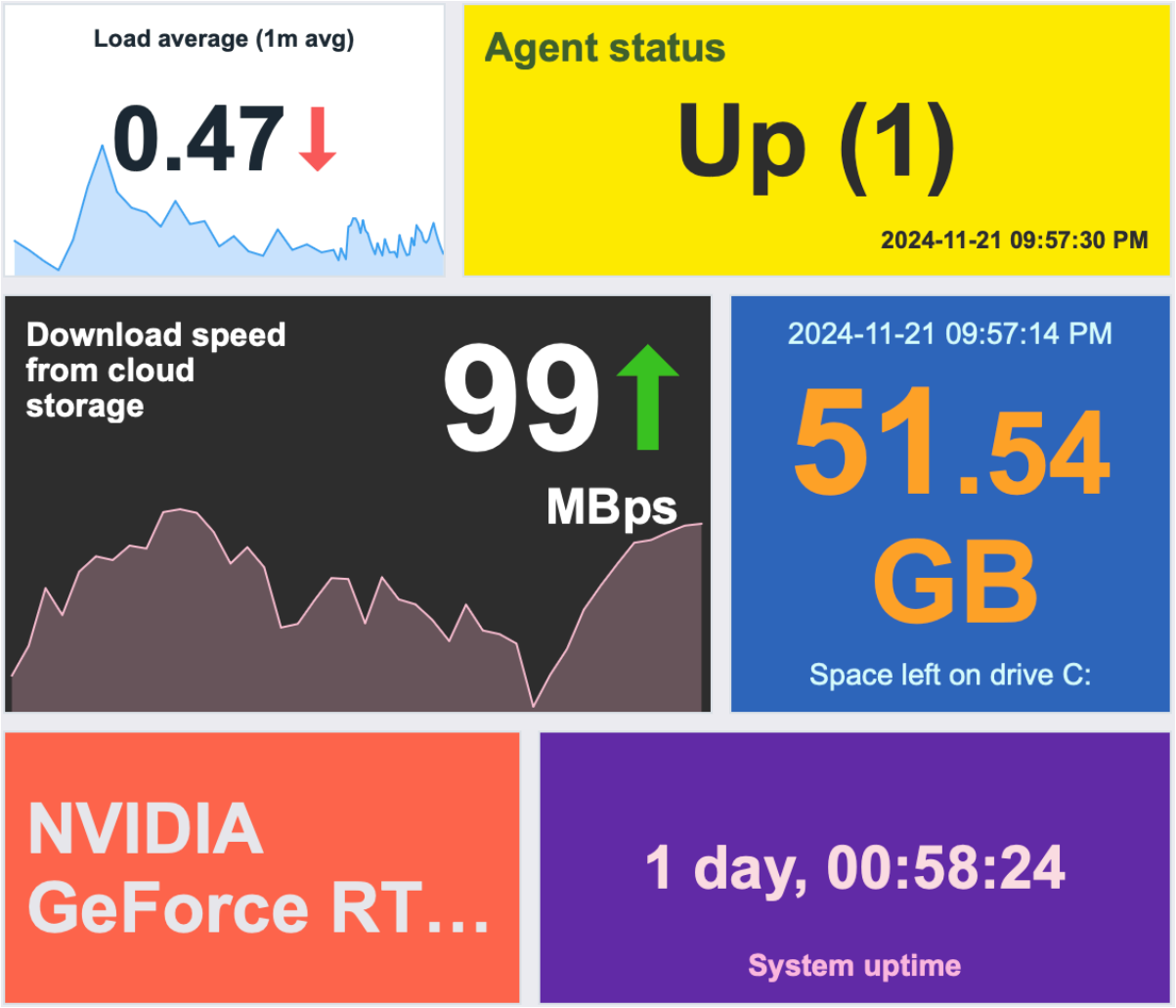
Etiquetas de host	Especifique las etiquetas de host para filtrar los items mostrados en el widget. Es posible incluir y excluir etiquetas y valores de etiquetas específicos. Se pueden establecer varias condiciones. La coincidencia de nombres de etiquetas siempre distingue entre mayúsculas y minúsculas. Hay varios operadores disponibles para cada condición: **Existe**: incluye los nombres de etiqueta especificados; **Es igual a**: incluye los nombres y valores de etiqueta especificados (distingue entre mayúsculas y minúsculas); **Contiene**: incluye los nombres de etiqueta especificados cuyos valores contienen la cadena introducida (coincidencia de subcadena, sin distinguir entre mayúsculas y minúsculas); **No existe**: excluye los nombres de etiqueta especificados; **No es igual a**: excluye los nombres y valores de etiqueta especificados (distingue entre mayúsculas y minúsculas); **No contiene**: excluye los nombres de etiqueta especificados cuyos valores contienen la cadena introducida (coincidencia de subcadena, sin distinguir entre mayúsculas y minúsculas). Hay dos tipos de cálculo para las condiciones: **Y/O**: deben cumplirse todas las condiciones; las condiciones que tienen el mismo nombre de etiqueta se agruparán mediante la condición O; **O**: basta con que se cumpla una condición. Este parámetro no está disponible al configurar el widget en un dashboard de template.
Patrones de item	Introduzca patrones de item o seleccione items existentes como patrones de item. Los items que coincidan con los patrones especificados se mostrarán en el navegador de items. Se pueden utilizar patrones con comodines para la selección (por ejemplo, `*` devolverá los items que coincidan con cero o más caracteres; `Zabbix*` devolverá los items que comiencen por "Zabbix"). Para especificar un patrón con comodines, introduzca la cadena manualmente y pulse *Enter*. Al comenzar a escribir, una lista desplegable mostrará los items coincidentes, limitada a los que pertenecen a los *Hosts* seleccionados o a los hosts incluidos en los *Grupos de hosts* seleccionados, si los hay. El símbolo de comodín siempre se interpreta; por lo tanto, no es posible añadir individualmente, por ejemplo, un item denominado *item**, si existen otros items coincidentes (por ejemplo, item2, item3). Al configurar el widget en un dashboard de template, este parámetro permite seleccionar únicamente los items configurados en el template.
Etiquetas de item	Especifique las etiquetas de item para filtrar los items mostrados en el widget. Es posible incluir y excluir etiquetas y valores de etiquetas específicos. Se pueden establecer varias condiciones. La coincidencia de nombres de etiquetas siempre distingue entre mayúsculas y minúsculas. Hay varios operadores disponibles para cada condición: **Existe**: incluye los nombres de etiqueta especificados; **Es igual a**: incluye los nombres y valores de etiqueta especificados (distingue entre mayúsculas y minúsculas); **Contiene**: incluye los nombres de etiqueta especificados cuyos valores contienen la cadena introducida (coincidencia de subcadena, sin distinguir entre mayúsculas y minúsculas); **No existe**: excluye los nombres de etiqueta especificados; **No es igual a**: excluye los nombres y valores de etiqueta especificados (distingue entre mayúsculas y minúsculas); **No contiene**: excluye los nombres de etiqueta especificados cuyos valores contienen la cadena introducida (coincidencia de subcadena, sin distinguir entre mayúsculas y minúsculas). Hay dos tipos de cálculo para las condiciones: **Y/O**: deben cumplirse todas las condiciones; las condiciones que tienen el mismo nombre de etiqueta se agruparán mediante la condición O; **O**: basta con que se cumpla una condición.
Estado	Filtre los items que se mostrarán según su estado (todos, normal, no soportado).

Mostrar problemas	Filtre los problemas que se mostrarán según su estado (todos, no suprimidos, ninguno). Los problemas se indican mediante bloques de colores al final de la fila del item. Los colores se basan en la gravedad del problema, que se puede personalizar en *Administración* > *General* > *Opciones de visualización de trigger*. Las gravedades de los problemas se pueden cambiar al actualizar problemas.
Agrupar por	Añada un atributo de agrupación por el que se agruparán los items: **Grupo de hosts**: agrupa los items por los grupos de hosts de sus hosts; **Nombre del host**: agrupa los items por sus hosts; **Valor de etiqueta de host**: introduzca un nombre de etiqueta para agrupar los items por los valores de esta etiqueta de host (por ejemplo, introduzca ”city” para agrupar los items por los valores ”Riga”, ”Tokyo”, etc.); **Valor de etiqueta de item**: introduzca un nombre de etiqueta para agrupar los items por los valores de esta etiqueta de item (por ejemplo, introduzca ”component” para agrupar los items por los valores ”cpu”, ”memory”, etc.). Los atributos de agrupación se pueden reordenar arrastrándolos hacia arriba o hacia abajo mediante el controlador situado delante del nombre del grupo. Tenga en cuenta que el orden de los atributos de agrupación determina el orden de anidamiento de los grupos. Por ejemplo, especificar varios nombres de etiquetas de host (1: color, 2: city) hará que los items se agrupen por color (red, blue, etc.) y después por ciudad (Riga, Tokyo, etc.). Un item puede mostrarse en varios grupos según los atributos de agrupación configurados (por ejemplo, al agrupar por grupo de hosts cuando el host del item pertenece a varios grupos de hosts). Al hacer clic en dichos items, se seleccionan y resaltan en todos los grupos. Los items que no coincidan con los atributos de agrupación configurados se muestran en el grupo *Sin categoría*. Si *Mostrar problemas* está configurado para mostrar problemas, estos se muestran de la siguiente manera: - para cada grupo, se muestra el recuento de problemas de todos los items subordinados; - para cada item, solo se muestra su recuento de problemas. Se pueden especificar hasta 10 atributos de agrupación, y todos deben ser únicos. Si no se especifica ningún atributo de agrupación, los items no se agruparán.
Límite de items	Introduzca el número máximo de items que se mostrarán. Los valores posibles van de 1 a 9999. Cuando hay más items disponibles para mostrar que el límite establecido, se muestra un mensaje correspondiente debajo de los items mostrados (por ejemplo, ”Se muestran 100 de más de 100 items”). Tenga en cuenta que el límite de items configurado también afecta a la visualización de los grupos configurados; por ejemplo, si el límite de items está establecido en 100 y los items se agrupan por sus hosts (cada uno con 200 items), solo se mostrará en el widget el primer host con sus 100 items. Este parámetro no se ve afectado por el parámetro *Límite para los resultados de búsqueda y filtrado* en *Administración* > *General* > *Interfaz gráfica de usuario*.

Valor del item

Descripción general

El widget *Item value* muestra el valor de un único item numérico o de cadena. Es útil para supervisar métricas importantes, visualizar umbrales y detectar cambios repentinos en los datos.



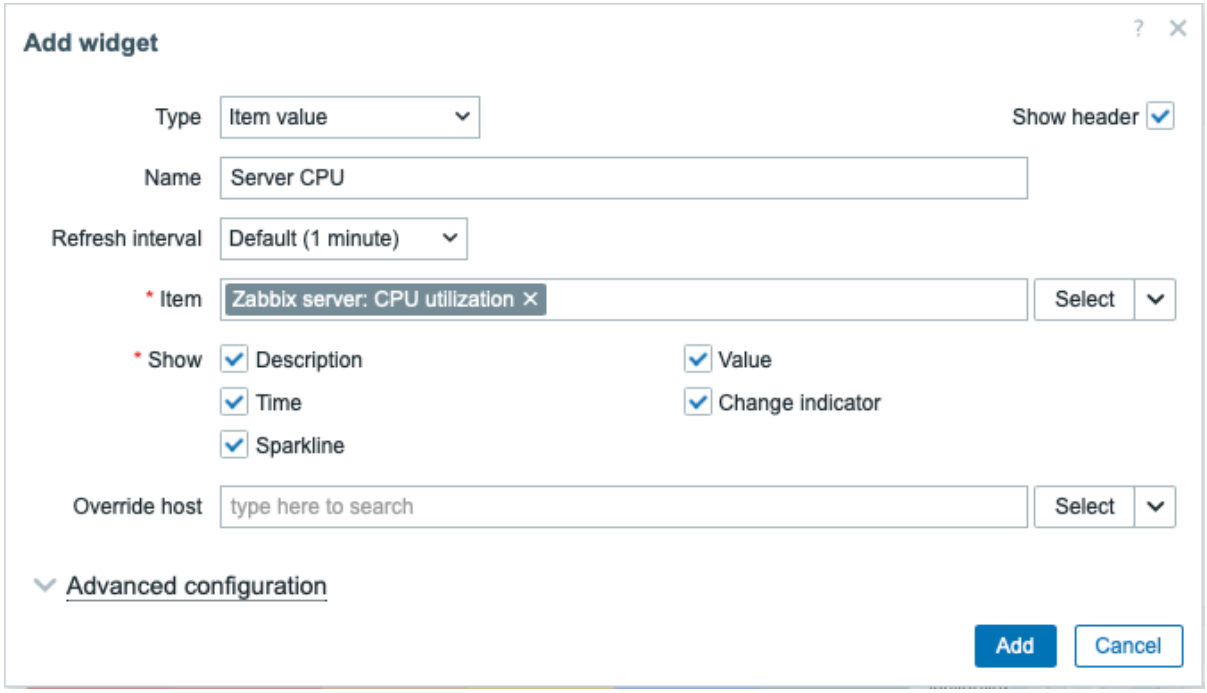
Puede configurar el widget para mostrar lo siguiente:

- Marca de tiempo de la métrica (2024-11-21 09:57:30)
- Descripción del item (Load average (1m), Agent status)
- Valor del item, indicador de cambio y unidades (0.47 ↓, 99 ↑ Mbps)
- Color de fondo
- Gráfico sparkline de los valores del período de tiempo especificado

Al hacer clic en el widget, se abre un gráfico ad hoc para items numéricos o los datos más recientes para items de cadena.

Configuración

Para configurar, seleccione *Valor del item* como tipo de widget:



Además de los parámetros comunes para todos los widgets, puede establecer las siguientes opciones específicas:

Item	Seleccione el item. Como alternativa, seleccione un widget compatible como fuente de datos para el item. Los items que devuelven datos binarios no son compatibles.
Mostrar	Marque la casilla para mostrar el elemento correspondiente: descripción, valor, hora, indicador de cambio, minigráfico. Desmárquela para ocultarlo. Debe seleccionarse al menos un elemento.
Sobrescribir host	Seleccione un widget compatible o el selector de host del dashboard como fuente de datos para hosts. Este parámetro no está disponible al configurar el widget en un dashboard de template.
Configuración avanzada	Haga clic en el encabezado *Configuración avanzada* para mostrar las opciones de configuración avanzada.

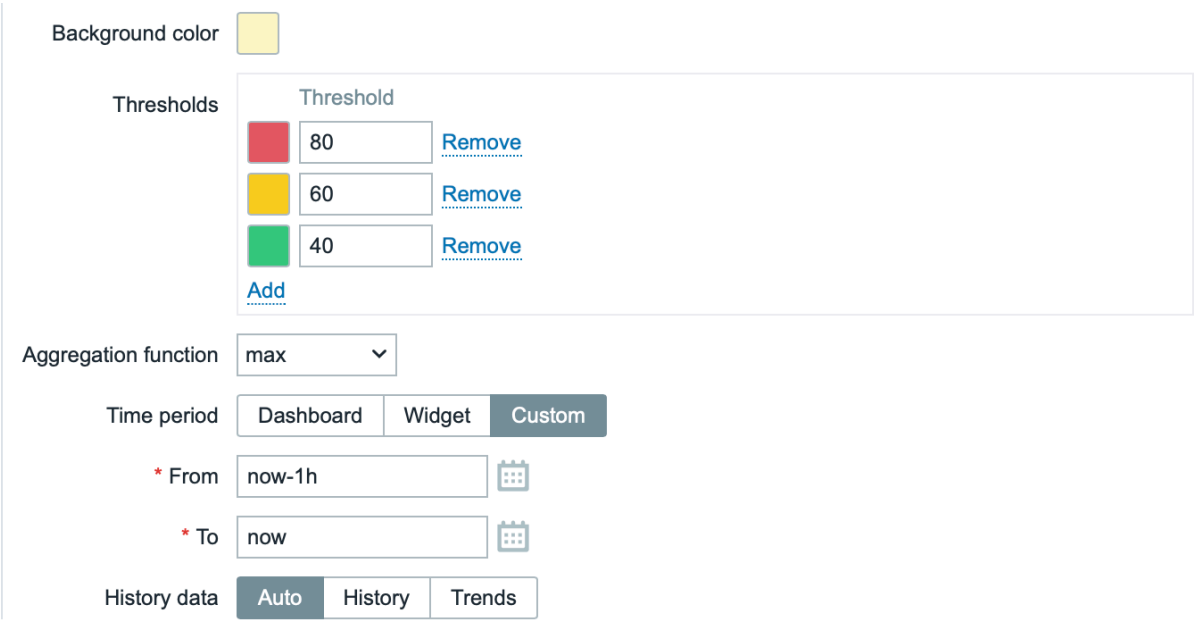
Configuración avanzada

Las opciones de configuración avanzada están disponibles en la sección plegable *Configuración avanzada*, y solo para aquellos elementos que están seleccionados en el campo *Mostrar* (ver arriba).

Tenga en cuenta que varios elementos no pueden ocupar el mismo espacio; si se colocan en el mismo espacio, se mostrará un mensaje de error.

Parámetros genéricos del widget

Estos parámetros determinan el color de fondo (estático o dinámico) de todo el widget y una función de agregación para mostrar los valores.

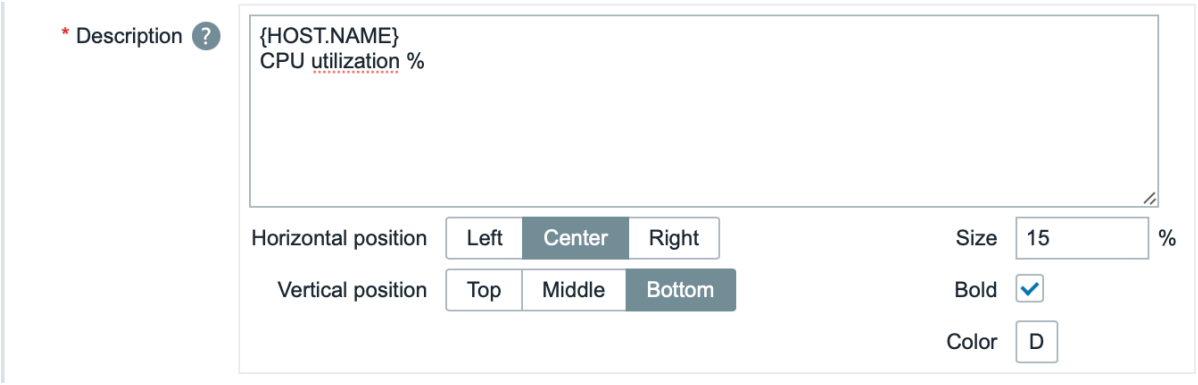


Color de fondo	Seleccione el color de fondo de todo el widget en el selector de color. D representa el color predeterminado (depende del tema del frontend). Para volver al valor predeterminado, haga clic en el botón *Usar predeterminado* del selector de color.
Umbrales	Configure el color de fondo dinámico de todo el widget. Haga clic en *Añadir* para añadir un umbral, seleccione el color de fondo en el selector de color y especifique un valor numérico. Cuando el valor del item sea igual o superior al valor del umbral, cambiará el color de fondo. La lista se ordenará en orden ascendente al guardarla. Tenga en cuenta que el color de fondo dinámico se mostrará correctamente solo para items numéricos.
Función de agregación	Especifique qué función de agregación se utilizará dentro del *Período de tiempo* seleccionado: **min** - muestra el valor más pequeño; **max** - muestra el valor más grande; **avg** - muestra el valor promedio; **count** - muestra la cantidad de valores; **sum** - muestra la suma de los valores; **first** - muestra el primer valor; **last** - muestra el último valor. Si se selecciona **no utilizar**, se muestra el valor más reciente (sin agregación). La agregación permite mostrar un valor agregado para el intervalo especificado (5 minutos, una hora, un día), en lugar del valor más reciente. Solo se pueden mostrar datos numéricos para *min*, *max*, *avg* y *sum*. Para *count*, los datos no numéricos se convertirán a numéricos.
Período de tiempo	Período de tiempo que se utilizará para agregar los valores. Seleccione la fuente de datos para el período de tiempo: **Dashboard** - utiliza el selector de período de tiempo del dashboard; **Widget** - utiliza un widget compatible (establecido en el parámetro *Widget*); **Personalizado** - utiliza un período de tiempo personalizado establecido en los parámetros *Desde* y *Hasta*; si se establece, se mostrará un icono de reloj en la esquina superior derecha del widget, que indicará la hora establecida al pasar el ratón por encima. Tenga en cuenta que, independientemente de la configuración de *Período de tiempo* del widget, los widgets compatibles aún pueden utilizarlo como fuente de datos para el período de tiempo. Este parámetro está oculto cuando *Función de agregación* está establecida en "no utilizar".
Widget	Introduzca o seleccione un widget compatible como fuente de datos para el período de tiempo. Este parámetro está disponible si *Período de tiempo* está establecido en "Widget".
Desde	Introduzca o seleccione el inicio del período de tiempo. Se admite la sintaxis de tiempo relativo (`now`, `now/d`, `now/w-1w`, etc.). Este parámetro está disponible si *Período de tiempo* está establecido en "Personalizado".
Hasta	Introduzca o seleccione el final del período de tiempo. Se admite la sintaxis de tiempo relativo (`now`, `now/d`, `now/w-1w`, etc.). Este parámetro está disponible si *Período de tiempo* está establecido en "Personalizado".

Datos históricos	Obtenga los datos del historial o de las tendencias: **Auto** - selección automática; **History** - obtiene los datos del historial; **Trends** - obtiene los datos de las tendencias. Esta configuración se aplica únicamente a los datos numéricos. Los datos no numéricos siempre se obtendrán del historial.

Descripción

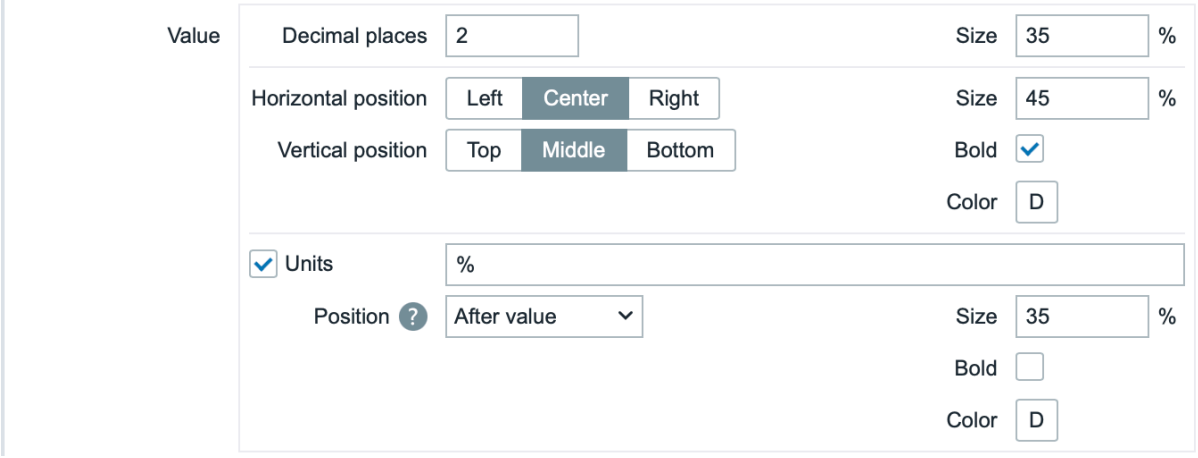
Estos parámetros determinan cómo debe mostrarse la descripción de la métrica.



Descripción	Introduzca la descripción de la métrica. Esta descripción puede sobrescribir el nombre por defecto de la métrica. Se admiten descripciones multilínea. Es posible una combinación de texto y macros soportadas. Se admiten {HOST.*}, {ITEM.*}, {INVENTORY.*} y macros de usuario.
Posición horizontal	Seleccione la posición horizontal de la descripción de la métrica: izquierda, derecha o centrada.
Posición vertical	Seleccione la posición vertical de la descripción de la métrica: arriba, abajo o en el medio.
Tamaño	Introduzca el tamaño de la fuente para la descripción de la métrica (en porcentaje relativo a la altura total del widget).
Negrita	Marque la casilla para mostrar la descripción de la métrica en negrita.
Color	Seleccione el color de la descripción de la métrica desde el selector de color. D significa color por defecto (depende del tema de la interfaz web). Para volver al valor por defecto, haga clic en el botón *Usar por defecto* en el selector de color.

Valor

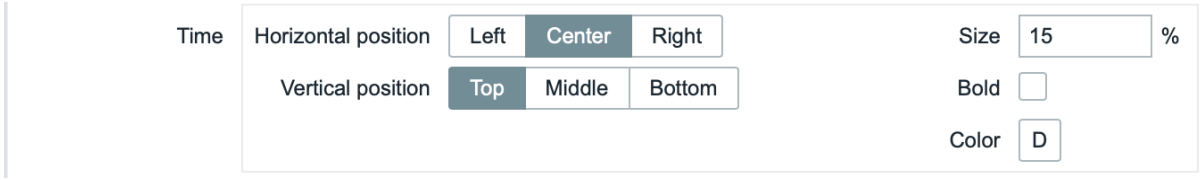
Estos parámetros determinan cómo se debe mostrar el valor de la métrica.



Decimales	Seleccione cuántos decimales se mostrarán con el valor. Este valor solo afectará a las métricas de tipo flotante. Para las métricas con unidades configuradas como ”s”, si este valor es mayor que 0 (el valor predeterminado es 2), el widget redondea la unidad de tiempo más significativa y muestra un valor numérico (por ejemplo, ”10.43m”) en lugar de convertir el valor en una cadena de tiempo completa. Cuando se establece en 0, los segundos se convierten en una cadena de tiempo legible (por ejemplo, ”4h 56m 30s”).
Tamaño	Introduzca la altura del tamaño de la fuente para los decimales (en porcentaje relativo a la altura total del widget).
Posición horizontal	Seleccione la posición horizontal del valor de la métrica: izquierda, derecha o centro.
Posición vertical	Seleccione la posición vertical del valor de la métrica: arriba, abajo o medio.
Tamaño	Introduzca la altura del tamaño de la fuente para el valor de la métrica (en porcentaje relativo a la altura total del widget). Tenga en cuenta que el tamaño del valor de la métrica tiene prioridad; otros elementos deben ceder espacio para el valor. Sin embargo, con el indicador de cambio, si el valor es demasiado grande, se truncará para mostrar el indicador de cambio.
Negrita	Marque la casilla para mostrar el valor de la métrica en negrita.
Color	Seleccione el color del valor de la métrica desde el selector de color. D significa color por defecto (depende del tema de la interfaz). Para volver al valor por defecto, haga clic en el botón *Usar por defecto* en el selector de color.
Unidades	
Unidades	Marque la casilla para mostrar las unidades junto al valor de la métrica. Si introduce un nombre de unidad, sobrescribirá la unidad de la configuración de la métrica.
Posición	Seleccione la posición de la unidad de la métrica: encima, debajo, antes o después del valor.
Tamaño	Introduzca la altura del tamaño de la fuente para la unidad de la métrica (en porcentaje relativo a la altura total del widget).
Negrita	Marque la casilla para mostrar la unidad de la métrica en negrita.
Color	Seleccione el color de la unidad de la métrica desde el selector de color. D significa color por defecto (depende del tema de la interfaz). Para volver al valor por defecto, haga clic en el botón *Usar por defecto* en el selector de color.

Hora

Estos parámetros determinan cómo se debe mostrar la hora (valor de reloj del historial de la métrica).



Posición horizontal	Seleccione la posición horizontal de la hora: izquierda, derecha o centro.
Posición vertical	Seleccione la posición vertical de la hora: arriba, abajo o medio.
Tamaño	Introduzca la altura del tamaño de la fuente para la hora (en porcentaje relativo a la altura total del widget).
Negrita	Marque la casilla para mostrar la hora en negrita.
Color	Seleccione el color de la hora desde el selector de color. D significa color por defecto (depende del tema de la interfaz). Para volver al valor por defecto, haga clic en el botón *Usar por defecto* en el selector de color.

Indicador de cambio

La sección permite seleccionar el color de los indicadores de cambio mediante el selector de color.



Los indicadores de cambio son los siguientes:

- ↑ - el valor del item ha aumentado (para items numéricos)
- ↓ - el valor del item ha disminuido (para items numéricos)
- ↕ - el valor del item ha cambiado (para items de tipo cadena y items con asignación de valores)

Nota:

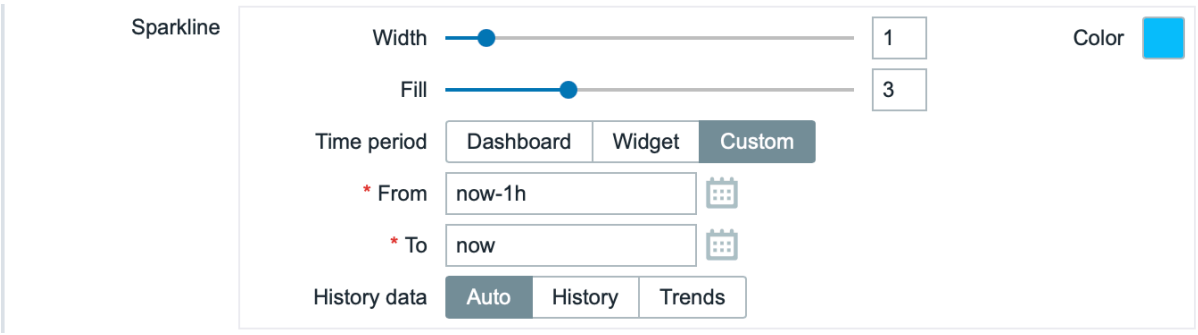
- Los indicadores de aumento (↑) y disminución (↓) no se muestran si solo hay un valor.
- Los valores siempre se comparan dentro del mismo ámbito; por ejemplo:
  - El valor más reciente se compara con el valor anterior.
  - Un valor mensual se compara con el mes anterior.
  - Para las agregaciones, el período anterior tiene la misma duración que el período seleccionado y termina inmediatamente antes del inicio del período seleccionado.

D representa el color predeterminado (depende del tema del frontend). Para volver al valor predeterminado, haga clic en el botón *Usar predeterminado* del selector de color.

El tamaño vertical del indicador de cambio es igual al tamaño del valor (la parte entera del valor para items numéricos).

Sparkline

Estos parámetros determinan cómo debe mostrarse el gráfico de sparkline.

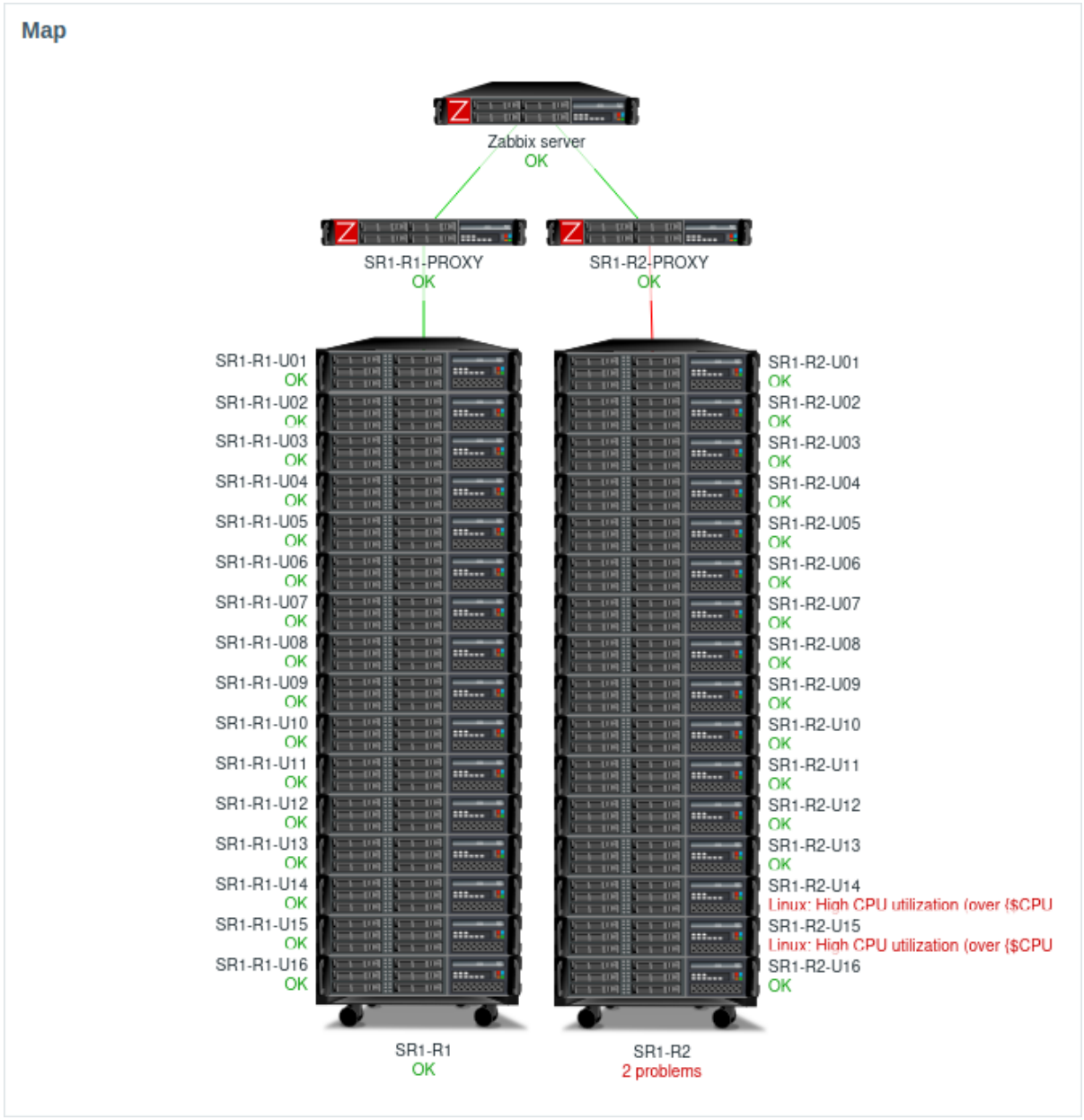


Ancho	Establezca el grosor de la línea del gráfico usando el control deslizante o introduciendo manualmente un valor en el rango de 0 a 10.
Color	Seleccione el color de la línea y del relleno.
Relleno	Establezca el nivel de transparencia del color de relleno usando el control deslizante o introduciendo manualmente un valor en el rango de 0 a 10.
Período de tiempo	Especifique el período de tiempo para los valores que se incluirán en el gráfico de sparkline. Seleccione la fuente de datos para el período de tiempo: **Panel**: use el selector de período de tiempo del panel; **Widget**: use un widget compatible (establecido en el parámetro *Widget*); **Personalizado**: use un período de tiempo personalizado establecido en los parámetros *Desde* y *Hasta*; si se establece, se mostrará un icono de reloj en la esquina superior derecha del widget, indicando el tiempo establecido al pasar el ratón. Tenga en cuenta que, independientemente de la configuración del *Período de tiempo* del widget, los widgets compatibles aún pueden usarlo como fuente de datos para el período de tiempo. Este parámetro no está disponible si la *Función de agregación* está configurada como "no utilizada".
Widget	Introduzca o seleccione un widget compatible como fuente de datos para el período de tiempo. Este parámetro está disponible si *Período de tiempo* está configurado como "Widget".
Desde	Introduzca o seleccione el inicio del período de tiempo. Se admite la sintaxis de tiempo relativo (`now`, `now/d`, `now/w-1w`, etc.). Este parámetro está disponible si *Período de tiempo* está configurado como "Personalizado".
Hasta	Introduzca o seleccione el final del período de tiempo. Se admite la sintaxis de tiempo relativo (`now`, `now/d`, `now/w-1w`, etc.). Este parámetro está disponible si *Período de tiempo* está configurado como "Personalizado".
Datos históricos	Tomar datos del historial o de tendencias: **Auto**: selección automática; **Historial**: tomar datos del historial; **Tendencias**: tomar datos de tendencias.

Mapa

Descripción general

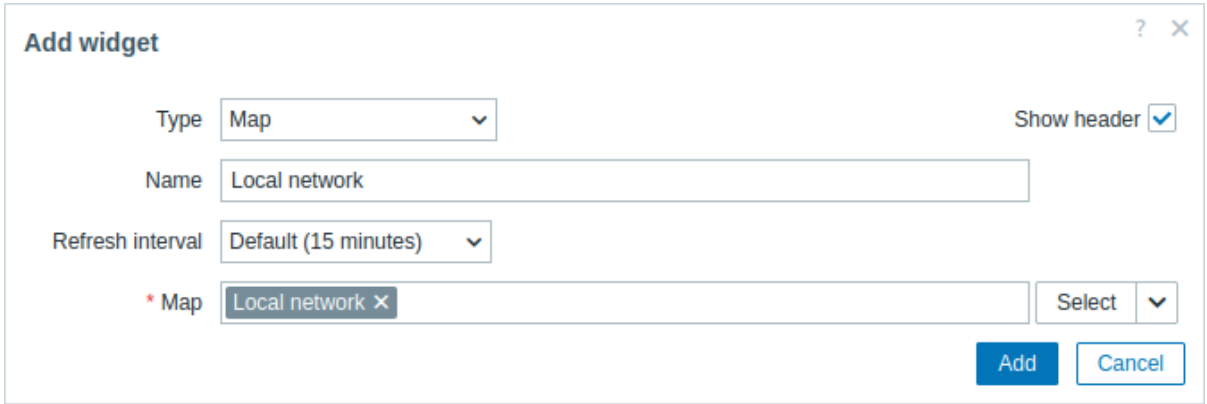
El widget *Mapa* puede mostrar un mapa de red o una visualización similar, proporcionándole una visión general dinámica de su red.



Puede combinarlo con el widget Árbol de navegación de mapas para mostrar dinámicamente el mapa seleccionado en el árbol.

Configuración

Para configurar, seleccione *Mapa* como tipo:



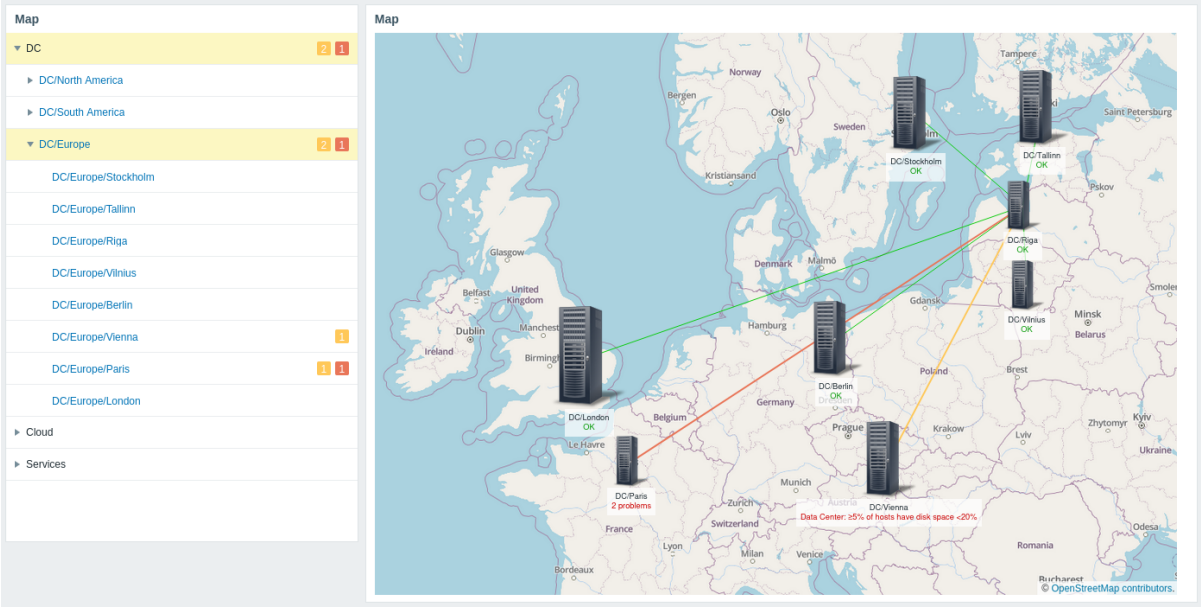
Además de los parámetros que son comunes para todos los widgets, puede establecer las siguientes opciones específicas:

Mapa	Establezca un mapa para mostrar. Alternativamente, seleccione un widget compatible como fuente de datos para mostrar el mapa. Este campo es autocompletado, por lo que al comenzar a escribir el nombre del mapa o widget se ofrecerá un menú desplegable con los mapas o widgets coincidentes.

Árbol de navegación del mapa

Descripción general

El widget *Árbol de navegación de mapas* muestra una jerarquía de mapas existentes y el recuento de problemas para cada mapa y grupo de mapas.

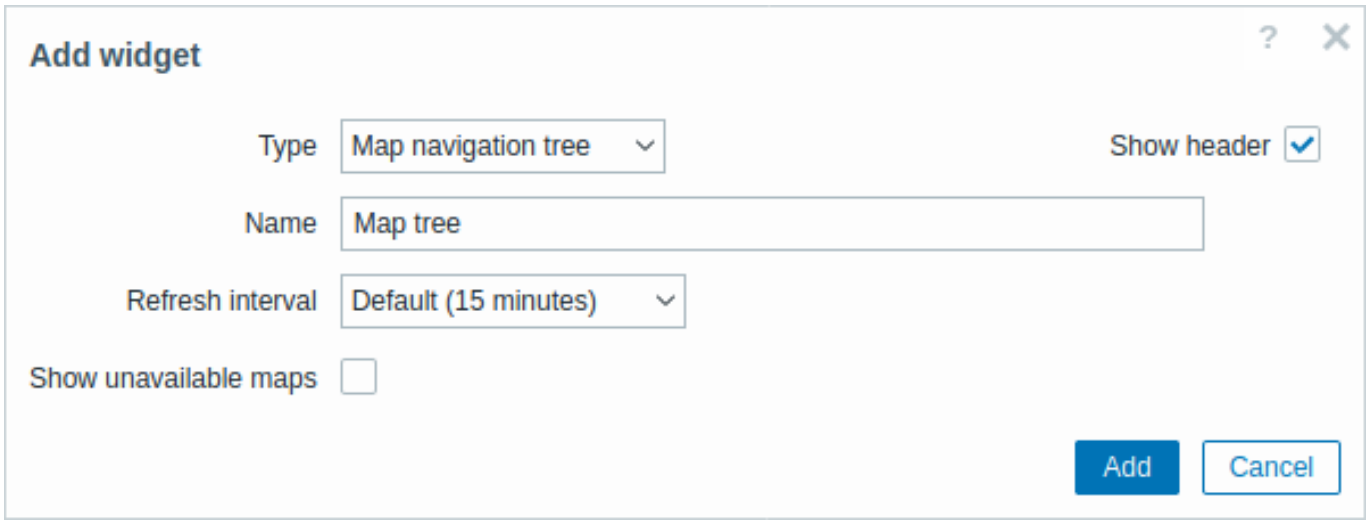


Puede emparejarlo con el widget Mapa para mostrar dinámicamente el mapa seleccionado en el árbol.

El recuento de problemas para el mapa de nivel superior en la jerarquía incluye todos los problemas de los submapas y los suyos propios.

Configuración

Para configurar, seleccione *Árbol de navegación de mapas* como tipo:

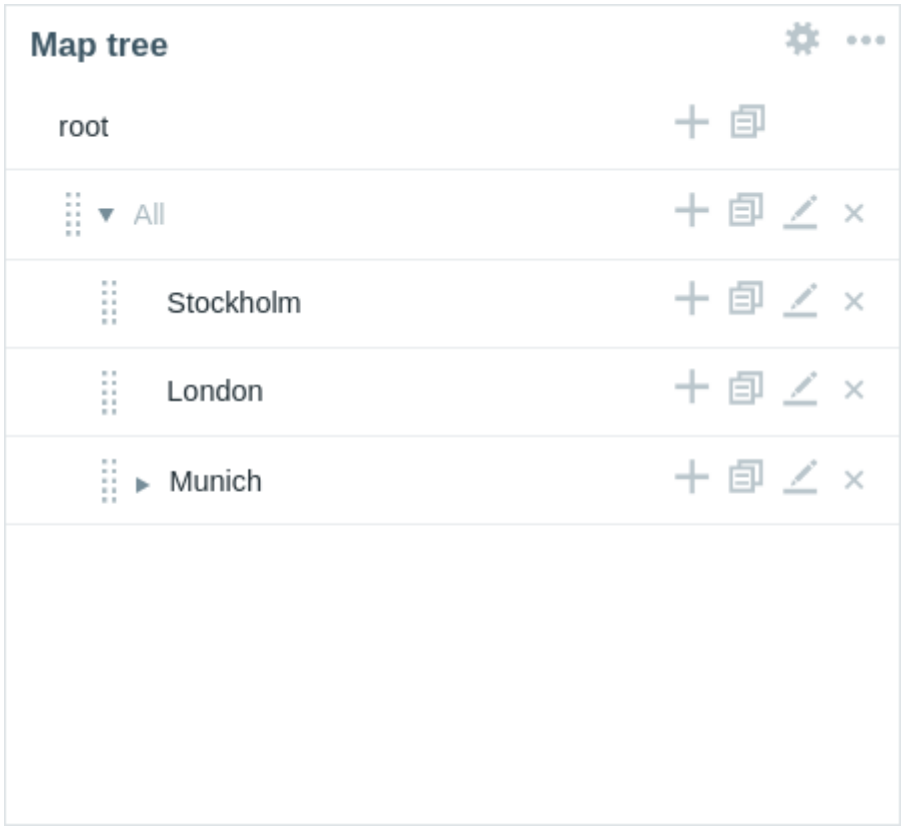


Además de los parámetros que son comunes para todos los widgets, puede establecer las siguientes opciones específicas:

Mostrar mapas no disponibles	Marque esta casilla para mostrar los mapas para los que el usuario no tiene permiso de lectura. Los mapas no disponibles en el árbol de navegación se mostrarán con un icono atenuado. Tenga en cuenta que si esta casilla está marcada, las submapas disponibles se mostrarán incluso si el mapa de nivel superior no está disponible. Si no está marcada, las submapas disponibles de un mapa principal no disponible no se mostrarán en absoluto. El recuento de problemas se calcula en función de los mapas disponibles y los elementos de mapa disponibles.

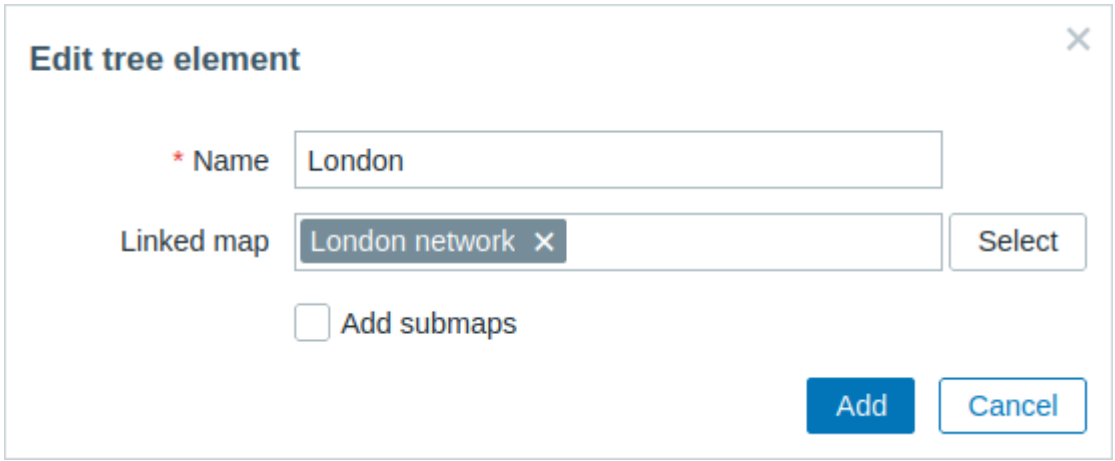
Los elementos del árbol de navegación se muestran en una lista. Puede:

- arrastrar un elemento (incluidos sus elementos secundarios) a un nuevo lugar en la lista;
- expandir o contraer un elemento para mostrar u ocultar sus elementos secundarios;
- añadir un elemento secundario (con o sin un mapa vinculado) a un elemento;
- añadir varios elementos secundarios (con mapas vinculados) a un elemento;
- editar un elemento;
- eliminar un elemento (incluidos sus elementos secundarios).



Configuración de elementos

Para configurar un elemento del árbol de navegación, agregue un nuevo elemento o edite uno existente.



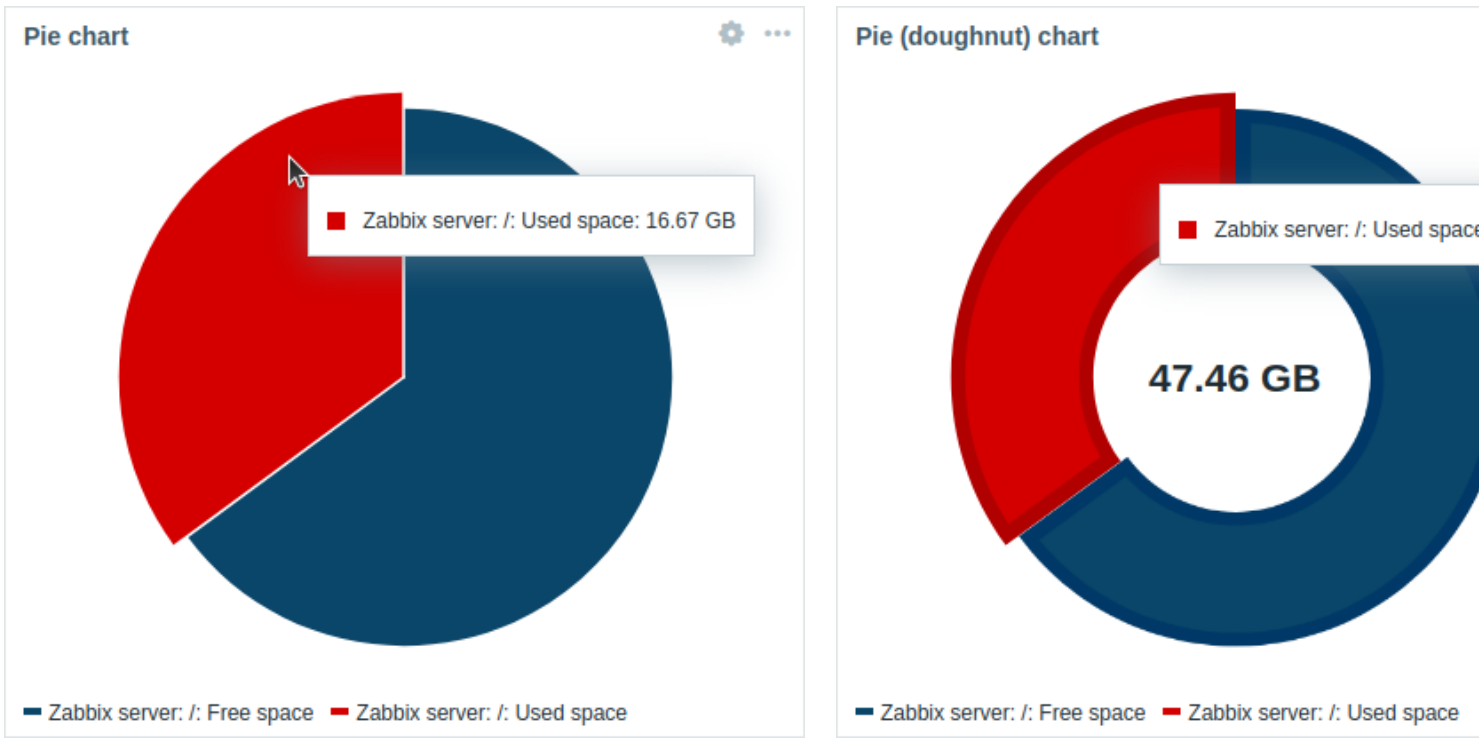
Los siguientes parámetros de configuración del elemento del árbol de navegación están disponibles:

Nombre	Introduzca el nombre del elemento del árbol de navegación.
Mapa vinculado	Seleccione el mapa para vincular al elemento del árbol de navegación. Este campo es autocompletado, por lo que al comenzar a escribir el nombre de un mapa se mostrará un menú desplegable con los mapas coincidentes.
Agregar submapas	Marque esta casilla para agregar los submapas del mapa vinculado como elementos secundarios al elemento del árbol de navegación.

Gráfico circular

Descripción general

El widget *Gráfico circular* muestra datos numéricos de los item como un gráfico circular o de anillo basado en vectores. Este widget puede ayudarle a visualizar cómo contribuyen los item o los host al conjunto de datos general.

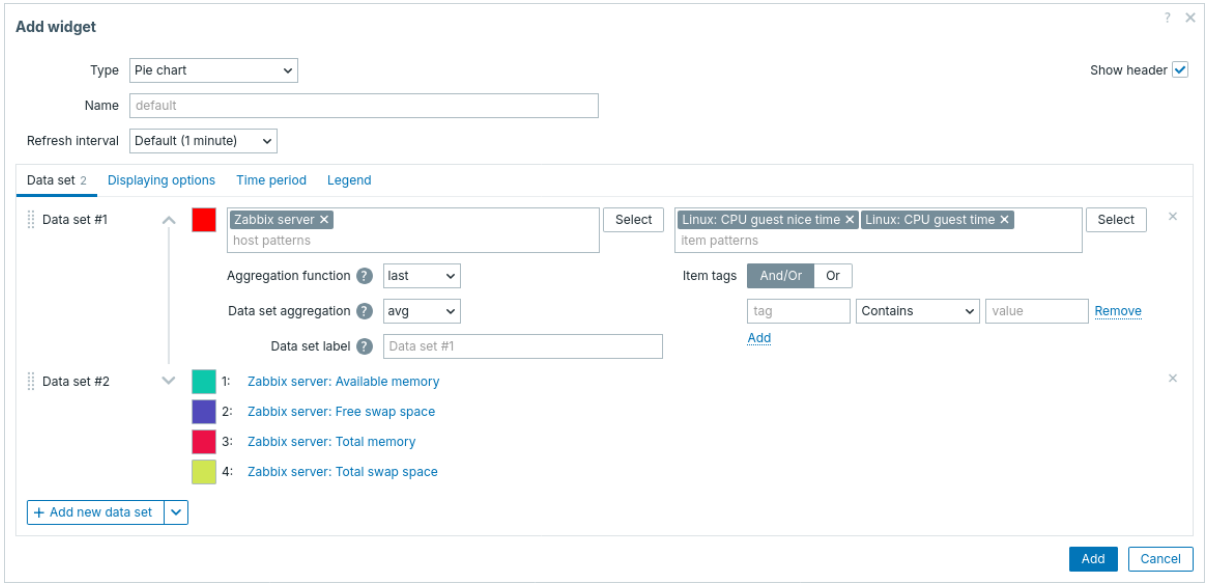


Al pasar el ratón por encima, el sector señalado se expande y muestra su leyenda; al hacer clic en el sector, este permanece expandido y su leyenda visible hasta que se cierra.

La información mostrada en el widget *Gráfico circular* se puede descargar como una imagen PNG seleccionando la opción *Descargar imagen* en el menú *Acciones* del widget ⋯, que aparece al pasar el ratón por encima del widget. Según la configuración del navegador, la imagen se guarda automáticamente en la carpeta `Downloads` o se abre un cuadro de diálogo *Guardar* que permite elegir la ubicación.

Configuración

Para configurar, seleccione *Gráfico circular* como tipo:



Conjunto de datos

La pestaña **Conjunto de datos** permite seleccionar los datos para el gráfico circular mediante la adición de conjuntos de datos. Se pueden añadir dos tipos de conjuntos de datos:

- *Patrones de item*: se muestran los datos de los items coincidentes. Puede elegir un único color base o seleccionar una fila de paleta para asignar colores distintos a cada item coincidente.
- *Lista de items*: se muestran los datos de los items seleccionados. Puede elegir individualmente el color de cada item mediante el selector.

De forma predeterminada, se añade un conjunto de datos de tipo *Patrones de item*.

Conjunto de datos	Para el conjunto de datos **Patrones de item**: Seleccione o introduzca patrones de host e item; los datos de los items que coincidan con los patrones introducidos se mostrarán en el gráfico circular; se pueden mostrar hasta 50 items. Se pueden utilizar patrones comodín para la selección (por ejemplo, ∗ devolverá resultados que coincidan con cero o más caracteres). Para especificar un patrón comodín, introduzca la cadena manualmente y pulse *Enter*. El símbolo comodín siempre se interpreta, por lo que no es posible añadir individualmente, por ejemplo, un item llamado *item** si existen otros items coincidentes (por ejemplo, *item2*, *item3*). Es obligatorio especificar patrones de host e item para los conjuntos de datos de tipo ”Patrones de item”. Consulte también: Detalles de configuración del conjunto de datos. Para el conjunto de datos **Lista de items**: Seleccione los items para el gráfico circular haciendo clic en el botón *Añadir item*. El menú desplegable *Tipo* situado después del nombre del item permite seleccionar el tipo de visualización de cada item: **Normal**: el valor del item se representa proporcionalmente en el gráfico circular (predeterminado); **Total**: el valor del item ocupa todo el gráfico circular. Tenga en cuenta que solo puede existir un item de tipo ”Total” por gráfico circular y que se colocará en primer lugar en la leyenda del gráfico circular. Si un item se establece como ”Total”, el parámetro *Agregación del conjunto de datos* (consulte más abajo) se deshabilitará y se establecerá en ”no utilizado”. También puede seleccionar widgets compatibles como fuente de datos para los items haciendo clic en el botón *Añadir widget*. Se aplican a los widgets las mismas opciones que a los items individuales. Es obligatorio especificar items o widgets para los conjuntos de datos de tipo ”Lista de items”. Consulte también: Detalles de configuración del conjunto de datos. Tenga en cuenta que solo se permiten tipos de item numéricos. Al configurar el widget en un dashboard de template, el parámetro para especificar patrones de host no está disponible y el parámetro para especificar una lista de items permite seleccionar únicamente los items configurados en el template.
Función de agregación	Especifique qué función de agregación se utilizará para cada item del conjunto de datos dentro del *Período de tiempo* seleccionado: **min**: muestra el valor más pequeño; **max**: muestra el valor más grande; **avg**: muestra el valor promedio; **sum**: muestra la suma de los valores; **count**: muestra el número de valores; **first**: muestra el primer valor; **last**: muestra el último valor (predeterminado). La agregación permite mostrar un valor agregado para el intervalo especificado (5 minutos, una hora, un día). Solo se pueden mostrar datos numéricos para *min*, *max*, *avg* y *sum*. Para *count*, los datos no numéricos se convertirán a numéricos.
Agregación del conjunto de datos	Especifique qué función de agregación se utilizará para todo el conjunto de datos: **no utilizado**: no se realiza ninguna agregación; los items se muestran por separado (predeterminado); **min**: muestra el valor más pequeño; **max**: muestra el valor más grande; **avg**: muestra el valor promedio; **sum**: muestra la suma de los valores; **count**: muestra el número de valores. La agregación permite mostrar un valor agregado para el intervalo especificado (5 minutos, una hora, un día), seleccionado en la pestaña *Período de tiempo* o utilizado para todo el dashboard.

Etiqueta del conjunto de datos	Especifique una etiqueta personalizada para el conjunto de datos. La etiqueta se muestra en la configuración del conjunto de datos y en la leyenda del gráfico circular (para los conjuntos de datos agregados). Todos los conjuntos de datos se numeran, incluidos aquellos que tienen una *Etiqueta del conjunto de datos* especificada. Si no se especifica ninguna etiqueta, el conjunto de datos se etiquetará automáticamente según su número (por ejemplo, ”Conjunto de datos n.º 2”, ”Conjunto de datos n.º 3”, etc.). La numeración de los conjuntos de datos se recalcula después de reordenarlos o arrastrarlos. Las etiquetas de conjuntos de datos demasiado largas se acortarán para ajustarse al espacio disponible (por ejemplo, ”Número de proc...”).
Etiquetas de item	Especifique las etiquetas para filtrar los items mostrados en el widget. Es posible incluir o excluir etiquetas y valores de etiqueta específicos. Se pueden establecer varias condiciones. La coincidencia de nombres de etiqueta siempre distingue entre mayúsculas y minúsculas. Hay varios operadores disponibles para cada condición: **Existe**: incluye los nombres de etiqueta especificados; **Es igual a**: incluye los nombres y valores de etiqueta especificados (distingue entre mayúsculas y minúsculas); **Contiene**: incluye los nombres de etiqueta especificados cuyos valores contienen la cadena introducida (coincidencia de subcadena, sin distinguir entre mayúsculas y minúsculas); **No existe**: excluye los nombres de etiqueta especificados; **No es igual a**: excluye los nombres y valores de etiqueta especificados (distingue entre mayúsculas y minúsculas); **No contiene**: excluye los nombres de etiqueta especificados cuyos valores contienen la cadena introducida (coincidencia de subcadena, sin distinguir entre mayúsculas y minúsculas). Hay dos tipos de cálculo para las condiciones: **Y/O**: deben cumplirse todas las condiciones; las condiciones que tienen el mismo nombre de etiqueta se agruparán mediante la condición O; **O**: basta con que se cumpla una condición.

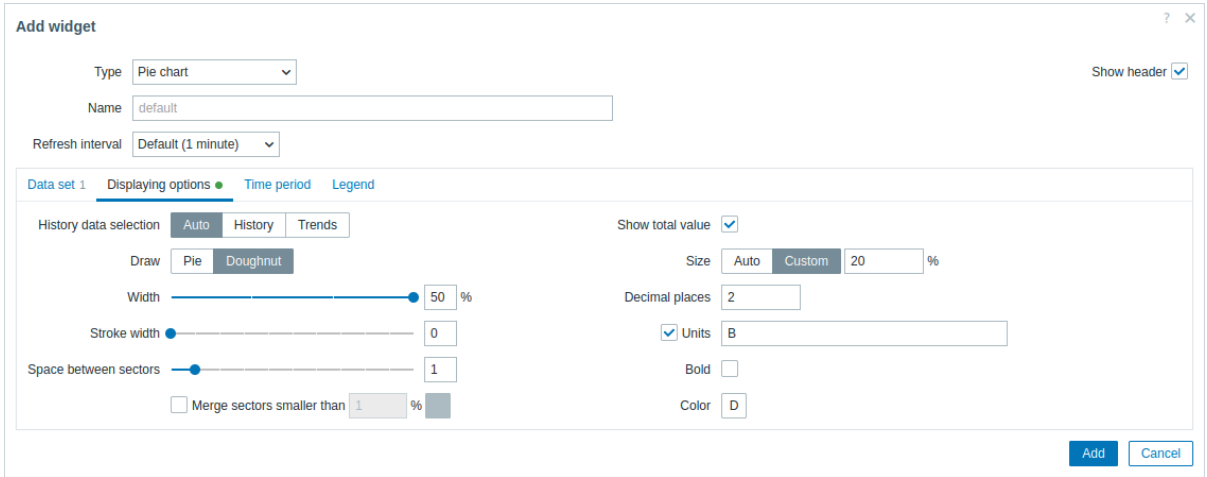
Detalles de la configuración del conjunto de datos

Los conjuntos de datos existentes se muestran en una lista. Puede reorganizar, expandir/colapsar, cambiar colores y clonar estos conjuntos de datos.

Para obtener más información, consulte los detalles de la configuración del conjunto de datos en el widget *Gráfico*. Estos detalles también se aplican al widget *Gráfico circular*.

Opciones de visualización

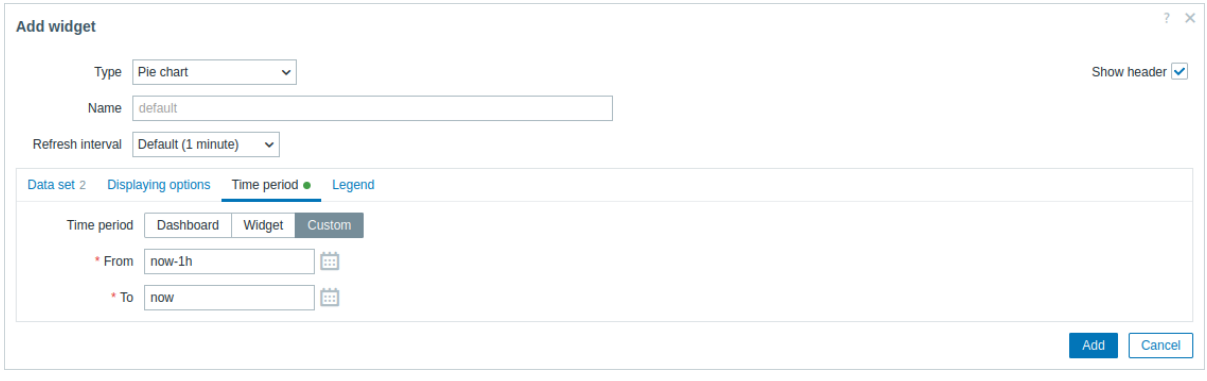
La pestaña **Opciones de visualización** permite definir la selección de datos históricos y las opciones de visualización del gráfico circular:



Selección de datos históricos		Seleccione la fuente de datos: **Auto**: los datos se obtienen según el algoritmo clásico (predeterminado); **History**: datos del historial; **Trends**: datos de tendencias.
Dibujar		Seleccione el estilo de visualización del gráfico circular: **Pie**: un círculo completo (los sectores ocupan el 100 % del radio); **Doughnut**: un círculo con un espacio vacío en el centro (los sectores ocupan hasta el 50 % del radio).
Espacio entre sectores		Seleccione el tamaño del espacio (en unidades de 0 a 10) entre los sectores (el valor predeterminado es ”1”).
Combinar sectores menores que N %		Marque la casilla para combinar los sectores menores que N %. Si está habilitada, seleccione el color de los sectores combinados y el umbral porcentual (N) para combinar los sectores pequeños.
Estilo de dibujo: ***Doughnut***		
	Ancho	Seleccione el ancho del gráfico de anillo: 20, 30, 40 o 50 % (predeterminado) del radio.
	Ancho del borde	Seleccione el ancho del borde del sector del gráfico de anillo (0-10).
	Mostrar valor total	Marque la casilla para mostrar el valor total en el centro del gráfico de anillo.
		Tamaño: Seleccione la opción de tamaño para el valor total: **Auto**: el tamaño del texto se ajusta automáticamente para que quepa de forma legible en el centro del gráfico de anillo; **Custom**: especifique el tamaño del texto como un porcentaje de la altura total del widget. *Posiciones decimales*: Especifique el número de posiciones decimales del valor total (0-6). *Unidades*: Especifique las unidades del valor total. *Negrita*: Marque la casilla para mostrar el valor total en negrita. *Color*: Seleccione el color del valor total.

Período de tiempo

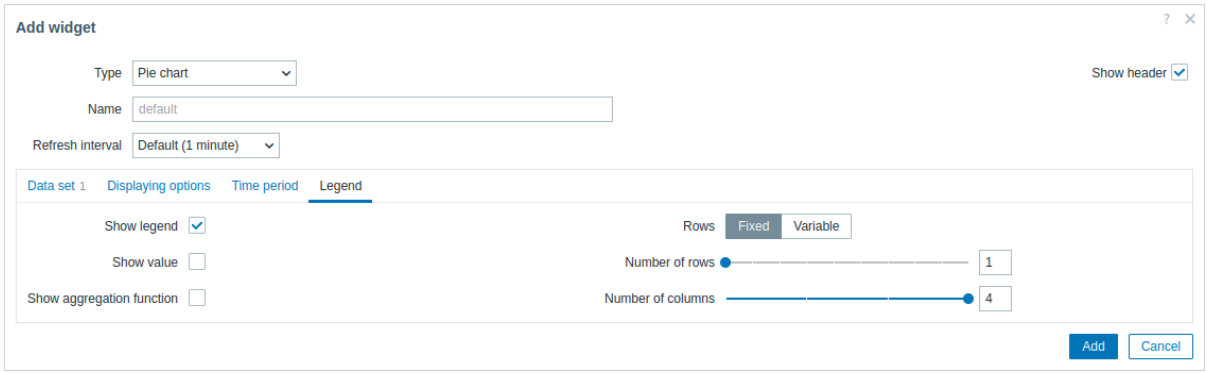
La pestaña **Período de tiempo** permite establecer un período de tiempo personalizado para la configuración de agregación del gráfico circular:



Período de tiempo	Seleccione la fuente de datos para el período de tiempo: **Dashboard**: utiliza el selector de período de tiempo del dashboard; **Widget**: utiliza un widget compatible (establecido en el parámetro *Widget*); **Personalizado**: utiliza un período de tiempo personalizado especificado en los parámetros *Desde* y *Hasta*; si se establece, se mostrará un icono de reloj en la esquina superior derecha del widget, que indicará la hora establecida al pasar el ratón por encima. Tenga en cuenta que, independientemente de la configuración de *Período de tiempo* del widget, los widgets compatibles aún pueden utilizarlo como fuente de datos para el período de tiempo.
Widget	Introduzca o seleccione un widget compatible (*Graph*, *Graph (classic)*, *Graph prototype*) como fuente de datos para el período de tiempo. Este parámetro está disponible si *Período de tiempo* está establecido en "Widget".
Desde	Introduzca o seleccione el inicio del período de tiempo. Se admite la sintaxis de tiempo relativo (`now`, `now/d`, `now/w-1w`, etc.). Este parámetro está disponible si *Período de tiempo* está establecido en "Personalizado".
Hasta	Introduzca o seleccione el final del período de tiempo. Se admite la sintaxis de tiempo relativo (`now`, `now/d`, `now/w-1w`, etc.). Este parámetro está disponible si *Período de tiempo* está establecido en "Personalizado".

Leyenda

La pestaña **Leyenda** permite personalizar la leyenda del gráfico circular:

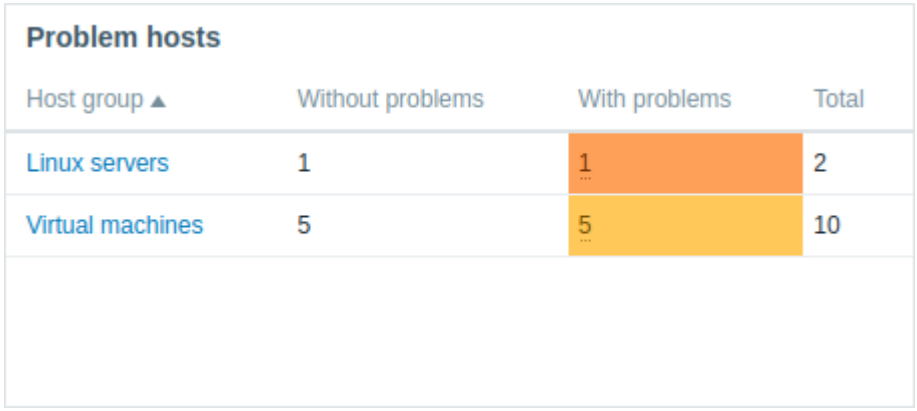


Mostrar leyenda	Desmarque esta casilla para ocultar la leyenda en el gráfico circular (marcada de forma predeterminada).
Mostrar valor	Marque esta casilla para mostrar el valor del item en la leyenda.
Mostrar función de agregación	Marque esta casilla para mostrar la función de agregación en la leyenda.
Filas	Seleccione el modo de visualización de las filas de la leyenda: **Fijas**: el número de filas mostradas está determinado por el valor del parámetro *Número de filas*; **Variables**: el número de filas mostradas está determinado por la cantidad de items configurados, sin superar el valor del parámetro *Número máximo de filas*.
Número de filas/ Número máximo de filas	Si *Filas* está establecido en «Fijas», establezca el número de filas de la leyenda que se mostrarán (1-10). Si *Filas* está establecido en «Variables», establezca el número máximo de filas de la leyenda que se mostrarán (1-10).
Número de columnas	Establezca el número de columnas de la leyenda que se mostrarán (1-4). Este parámetro está disponible si *Mostrar valor* no está marcado.

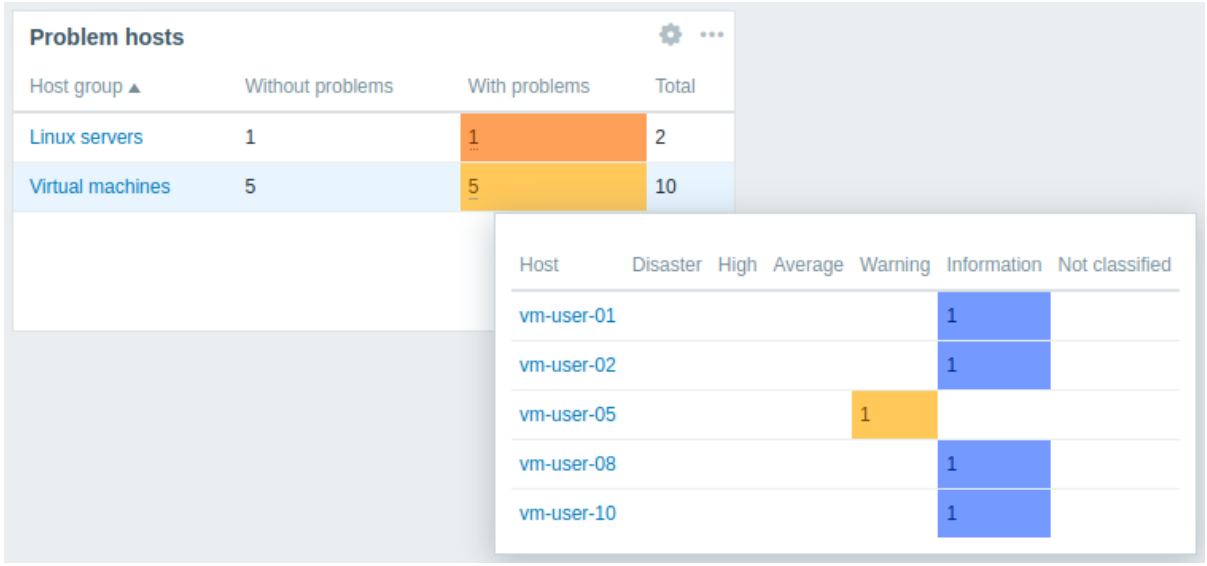
Hosts con problemas

Descripción general

El widget *Hosts con problemas* muestra el número de problemas causa por grupo de equipos y muestra la mayor gravedad de problema en cada grupo. Permite filtrar los problemas por diferentes parámetros (grupos de equipos, equipos, nombres de problemas, etc.).

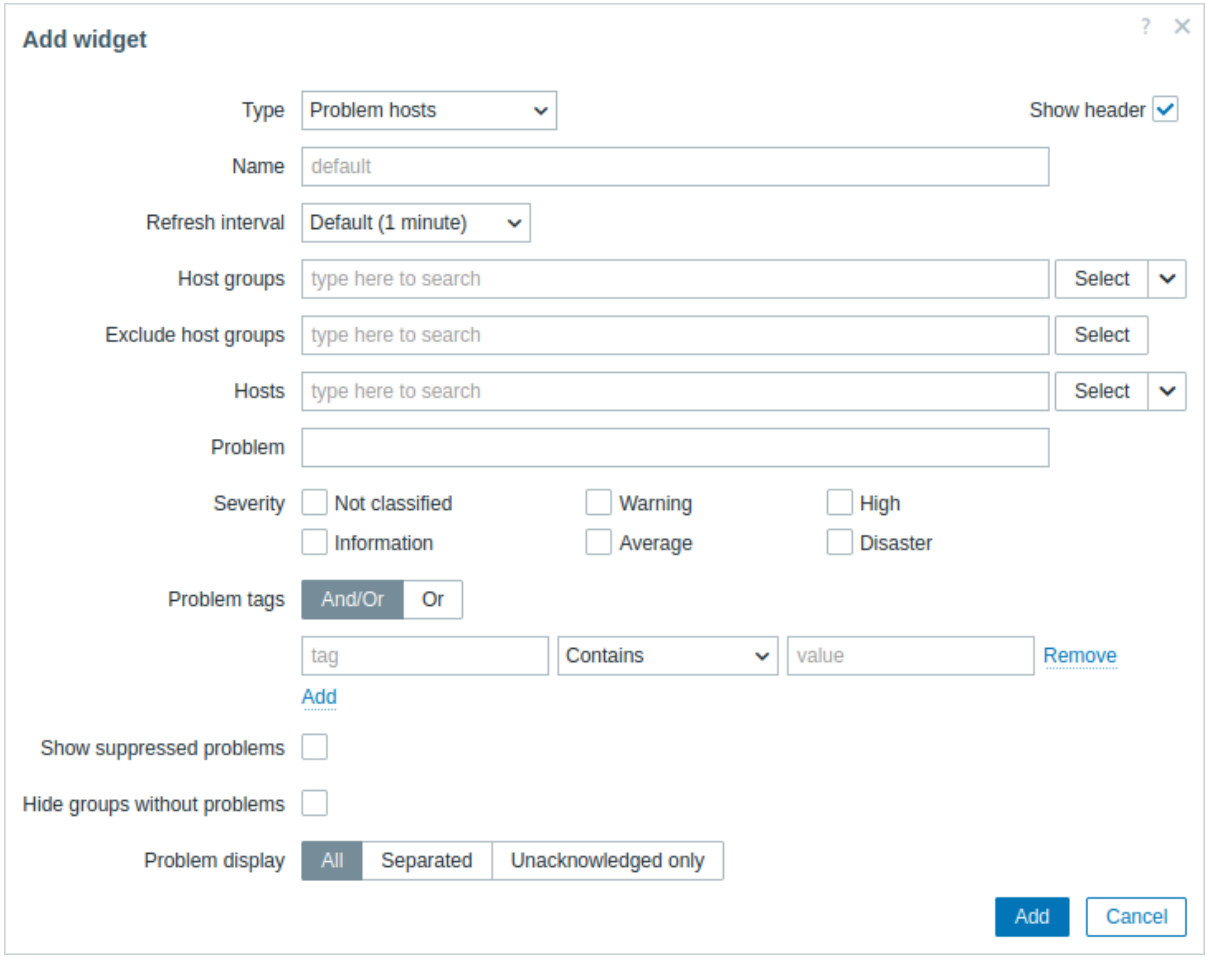


Al pasar el cursor sobre el recuento de problemas se muestra la lista de equipos con problemas:



Configuración

Para configurar, seleccione *Equipos con problemas* como tipo:



Además de los parámetros que son comunes para todos los widgets, puede establecer las siguientes opciones específicas:

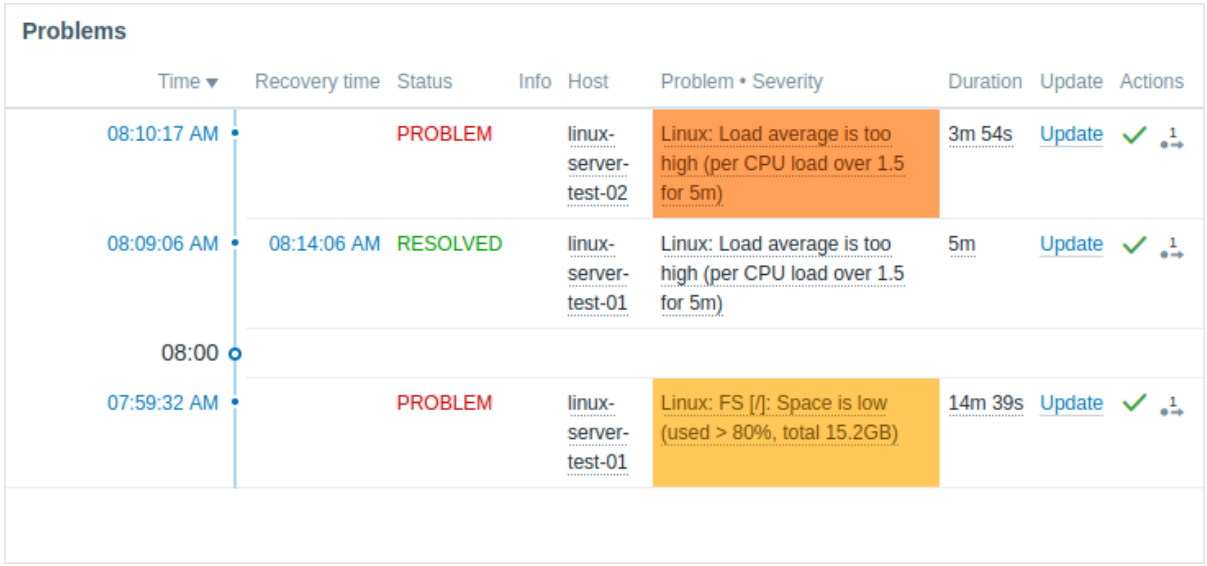
Grupos de equipos	Seleccione los grupos de equipos que se mostrarán en el widget. Alternativamente, seleccione un widget compatible como fuente de datos para los grupos de equipos. Este campo es autocompletable, por lo que al comenzar a escribir el nombre de un grupo se ofrecerá un menú desplegable con los grupos coincidentes. Especificar un grupo de equipos principal selecciona implícitamente todos los grupos de equipos anidados. Los datos de los equipos de estos grupos de equipos se mostrarán en el widget; si no se ingresan grupos de equipos, se mostrarán todos los grupos de equipos. Este parámetro no está disponible al configurar el widget en un dashboard de plantilla.
Excluir grupos de equipos	Seleccione los grupos de equipos que se ocultarán en el widget. Este campo es autocompletable, por lo que al comenzar a escribir el nombre de un grupo se ofrecerá un menú desplegable con los grupos coincidentes. Especificar un grupo de equipos principal selecciona implícitamente todos los grupos de equipos anidados. Los datos de los equipos de estos grupos de equipos no se mostrarán en el widget. Por ejemplo, los equipos 001, 002, 003 pueden estar en el Grupo A y los equipos 002, 003 también en el Grupo B. Si seleccionamos *mostrar* el Grupo A y *excluir* el Grupo B al mismo tiempo, solo se mostrarán los datos del equipo 001 en el dashboard. Este parámetro no está disponible al configurar el widget en un dashboard de plantilla.
Equipos	Seleccione los equipos que se mostrarán en el widget. Alternativamente, seleccione un widget compatible o el dashboard como fuente de datos para los equipos. Este campo es autocompletable, por lo que al comenzar a escribir el nombre de un equipo se ofrecerá un menú desplegable con los equipos coincidentes. Si no se ingresan equipos, se mostrarán todos los equipos. Este parámetro no está disponible al configurar el widget en un dashboard de plantilla.

Problema	Puede limitar el número de equipos con problemas mostrados por el nombre del problema. Si ingresa una cadena aquí, solo se mostrarán aquellos equipos con problemas cuyo nombre contenga la cadena ingresada. Las macros no se expanden.
Severidad	Marque las severidades de los problemas para filtrar los problemas que se mostrarán en el widget. Si no se marca ninguna severidad, se mostrarán todos los problemas.
Etiquetas de problema	Especifique etiquetas de problema para limitar el número de problemas mostrados en el widget. Es posible incluir así como excluir etiquetas y valores de etiquetas específicos. Se pueden establecer varias condiciones. La coincidencia de nombres de etiquetas siempre distingue entre mayúsculas y minúsculas. Hay varios operadores disponibles para cada condición: **Existe** - incluir los nombres de etiqueta especificados; **Igual** - incluir los nombres y valores de etiqueta especificados (distingue mayúsculas y minúsculas); **Contiene** - incluir los nombres de etiqueta especificados donde los valores de etiqueta contengan la cadena ingresada (coincidencia de subcadena, no distingue mayúsculas y minúsculas); **No existe** - excluir los nombres de etiqueta especificados; **No es igual** - excluir los nombres y valores de etiqueta especificados (distingue mayúsculas y minúsculas); **No contiene** - excluir los nombres de etiqueta especificados donde los valores de etiqueta contengan la cadena ingresada (coincidencia de subcadena, no distingue mayúsculas y minúsculas). Hay dos tipos de cálculo para las condiciones: **Y/O** - se deben cumplir todas las condiciones, las condiciones con el mismo nombre de etiqueta se agruparán por la condición *O*; **O** - basta con que se cumpla una condición.
Mostrar problemas suprimidos	Marque la casilla para mostrar los problemas que de otro modo serían suprimidos (no mostrados) debido al mantenimiento del equipo.
Ocultar grupos sin problemas	Marque la opción *Ocultar grupos sin problemas* para ocultar los datos de los grupos de equipos sin problemas en el widget. Este parámetro no está disponible al configurar el widget en un dashboard de plantilla.
Visualización de problemas	Mostrar el recuento de problemas como: **Todos** - se mostrará el recuento total de problemas; **Separados** - se mostrará el recuento de problemas no reconocidos separado como un número del recuento total de problemas; **Solo no reconocidos** - solo se mostrará el recuento de problemas no reconocidos.

Problemas

Descripción general

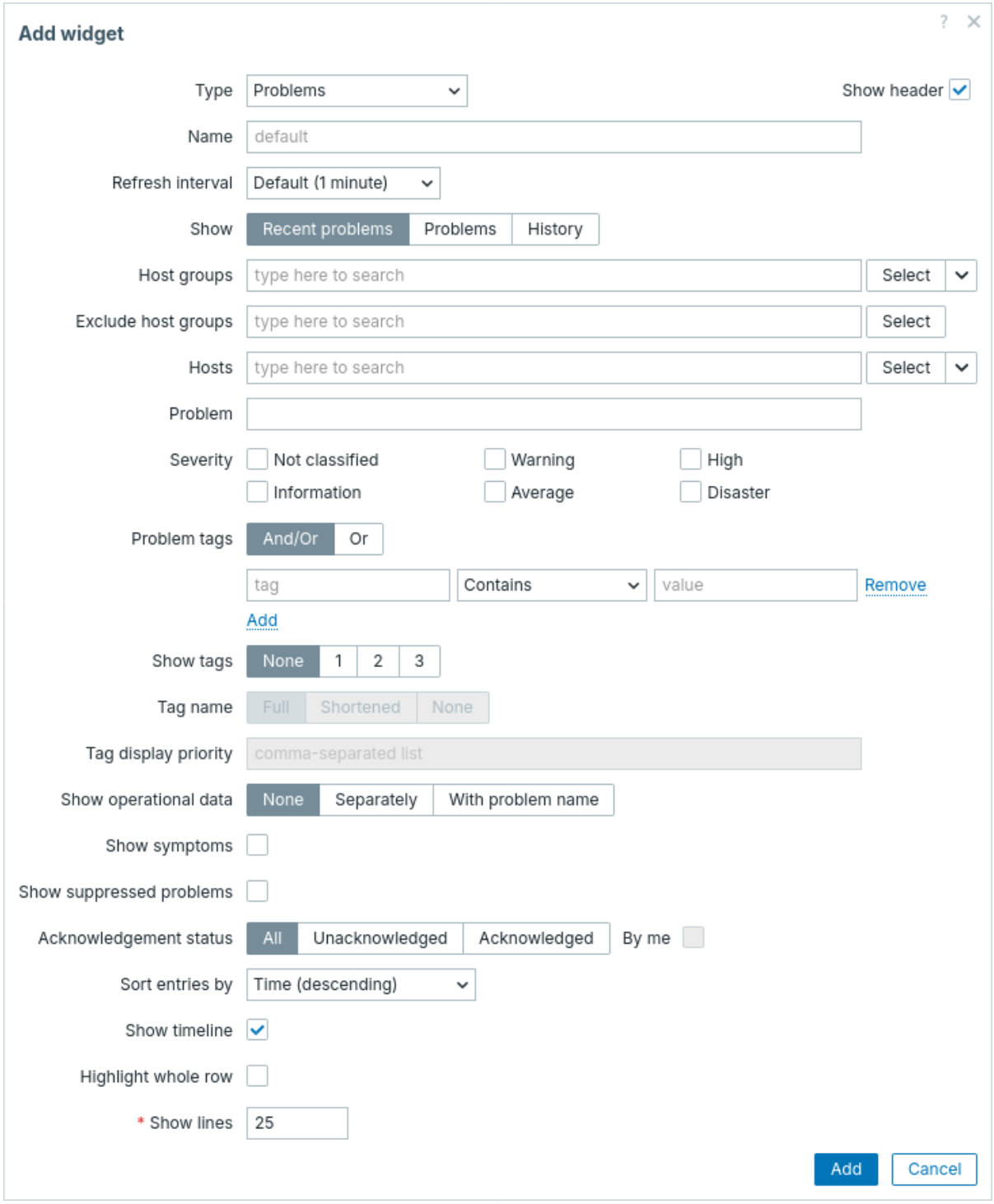
El widget *Problemas* muestra los problemas actuales filtrados por diferentes parámetros (grupos de equipos, equipos, nombres de problemas, etc.), dándole una visión clara de lo que requiere atención.



Muestra los mismos datos que *Supervisión* > *Problemas* y puede mostrar hasta 1000 registros.

Configuración

Para configurar, seleccione *Problems* como tipo:



Puede limitar de varias maneras cuántos problemas se muestran en el widget: por estado del problema, nombre del problema, severidad, grupo de host, host, etiqueta del evento, estado de reconocimiento, etc.

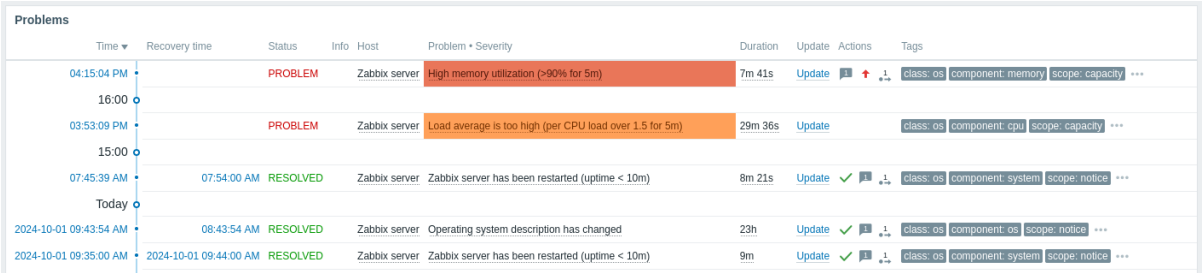
Además de los parámetros comunes para todos los widgets, puede definir las siguientes opciones específicas:

Show	Filtrar por estado del problema: **Recent problems** - se muestran los problemas no resueltos y los resueltos recientemente (predeterminado); **Problems** - se muestran los problemas no resueltos; **History** - se muestra el historial de todos los eventos.

Host groups	Seleccione los grupos de host de los que se mostrarán problemas en el widget. Como alternativa, seleccione un widget compatible como fuente de datos para los grupos de host. Este campo tiene autocompletado, por lo que al empezar a escribir el nombre de un grupo se mostrará una lista desplegable con los grupos coincidentes. Al especificar un grupo de host padre, se seleccionan implícitamente todos los grupos de host anidados. Los problemas de estos grupos de host se mostrarán en el widget; si no se introduce ningún grupo de host, se mostrarán los problemas de todos los grupos de host. Este parámetro no está disponible al configurar el widget en un template dashboard.
Exclude host groups	Seleccione los grupos de host cuyos problemas se ocultarán en el widget. Este campo tiene autocompletado, por lo que al empezar a escribir el nombre de un grupo se mostrará una lista desplegable con los grupos coincidentes. Al especificar un grupo de host padre, se seleccionan implícitamente todos los grupos de host anidados. Los problemas de estos grupos de host no se mostrarán en el widget. Por ejemplo, los hosts 001, 002 y 003 pueden estar en el Grupo A, y los hosts 002 y 003 también en el Grupo B. Si seleccionamos *show* Grupo A y *exclude* Grupo B al mismo tiempo, solo se mostrarán en el widget los problemas del host 001. Este parámetro no está disponible al configurar el widget en un template dashboard.
Hosts	Seleccione los hosts de los que se mostrarán problemas en el widget. Como alternativa, seleccione un widget compatible o el dashboard como fuente de datos para los hosts. Este campo tiene autocompletado, por lo que al empezar a escribir el nombre de un host se mostrará una lista desplegable con los hosts coincidentes. Si no se introduce ningún host, se mostrarán los problemas de todos los hosts. Este parámetro no está disponible al configurar el widget en un template dashboard.
Problem	Puede limitar el número de problemas mostrados por su nombre. Si introduce una cadena aquí, solo se mostrarán los problemas cuyo nombre contenga la cadena introducida. Las macros no se expanden.
Severity	Marque las severidades de problema para filtrar los problemas que se mostrarán en el widget. Si no se marca ninguna severidad, se mostrarán todos los problemas.
Problem tags	Especifique etiquetas de problema para limitar el número de problemas mostrados en el widget. Es posible incluir y excluir etiquetas y valores de etiqueta específicos. Se pueden establecer varias condiciones. La coincidencia del nombre de la etiqueta siempre distingue entre mayúsculas y minúsculas. Hay varios operadores disponibles para cada condición: **Exists** - incluir los nombres de etiqueta especificados; **Equals** - incluir los nombres y valores de etiqueta especificados (distingue mayúsculas y minúsculas); **Contains** - incluir los nombres de etiqueta especificados cuyos valores de etiqueta contengan la cadena introducida (coincidencia de subcadena, no distingue mayúsculas y minúsculas); **Does not exist** - excluir los nombres de etiqueta especificados; **Does not equal** - excluir los nombres y valores de etiqueta especificados (distingue mayúsculas y minúsculas); **Does not contain** - excluir los nombres de etiqueta especificados cuyos valores de etiqueta contengan la cadena introducida (coincidencia de subcadena, no distingue mayúsculas y minúsculas). Hay dos tipos de cálculo para las condiciones: **And/Or** - deben cumplirse todas las condiciones; las condiciones que tengan el mismo nombre de etiqueta se agruparán mediante la condición *Or*; **Or** - basta con que se cumpla una condición. Cuando se filtran, las etiquetas especificadas aquí se mostrarán primero junto con el problema, salvo que la lista *Tag display priority* (véase más abajo) las sobrescriba.

Show tags	Seleccione el número de etiquetas mostradas: **None** - sin columna *Tags*; **1** - la columna *Tags* contiene una etiqueta; **2** - la columna *Tags* contiene dos etiquetas; **3** - la columna *Tags* contiene tres etiquetas. Para ver todas las etiquetas del problema, pase el ratón sobre el icono de los tres puntos.
Tag name	Seleccione el modo de visualización del nombre de la etiqueta: **Full** - los nombres y valores de las etiquetas se muestran completos; **Shortened** - los nombres de las etiquetas se acortan a 3 símbolos, pero los valores de las etiquetas se muestran completos; **None** - solo se muestran los valores de las etiquetas; no se muestran nombres.
Tag display priority	Introduzca la prioridad de visualización de etiquetas para un problema, como una lista de etiquetas separadas por comas. Solo deben usarse nombres de etiqueta, no valores. Ejemplo: `Services,Applications,Application` Las etiquetas de esta lista siempre se mostrarán primero, anulando el orden natural alfabético.
Show operational data	Seleccione el modo de visualización de los datos operativos: **None** - no se muestran datos operativos; **Separately** - los datos operativos se muestran en una columna separada; **With problem name** - añade los datos operativos al nombre del problema, usando paréntesis para los datos operativos. Para más detalles, consulte la descripción del campo *Operational data* en la configuración del trigger.
Show symptoms	Marque la casilla para mostrar en su propia línea los problemas clasificados como síntomas.
Show suppressed problems	Marque la casilla para mostrar los problemas que, de otro modo, estarían suprimidos (no mostrados) debido al mantenimiento del host o a la supresión de problemas individual.
Acknowledgement status	Filtre para mostrar todos los problemas, solo los problemas sin reconocer o solo los problemas reconocidos. Marque la casilla adicional para filtrar los problemas que alguna vez haya reconocido usted.
Sort entries by	Ordene las entradas por: **Time** (descendente o ascendente); **Severity** (descendente o ascendente); **Problem name** (descendente o ascendente); **Host** (descendente o ascendente). Ordenar las entradas por **Host** (descendente o ascendente) no está disponible al configurar el widget en un template dashboard.
Show timeline	Marque la casilla para mostrar una línea de tiempo visual.
Highlight whole row	Marque la casilla para resaltar la línea completa de los problemas no resueltos. Se utiliza el color de severidad del problema para el resaltado. *Highlight whole row* no está disponible en los temas de alto contraste.
Show lines	Especifique el número de líneas de problema que se mostrarán.

Uso del widget



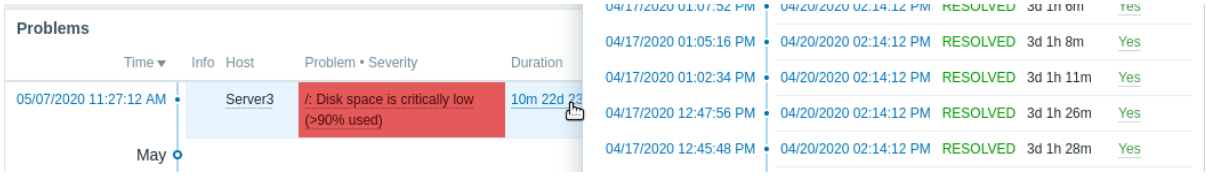
El widget Problems ofrece acceso rápido a información adicional:

- Haga clic en la fecha del problema (en la columna Time) o en la fecha de recuperación (en la columna Recovery Time) para ver los detalles del evento.
- Si el campo Info no está vacío, pase el cursor sobre el icono mostrado para ver detalles adicionales.
- Haga clic en el nombre del host para abrir el menú del host.
- Haga clic en el nombre del problema para abrir el menú del evento.
- Pase el cursor sobre la duración del problema o haga clic en ella para ver el ventana emergente del evento del problema.

- Al hacer clic en *Update* se abre una ventana Actualizar problema.
- Pase el cursor sobre el icono de flecha gris en la columna Actions o haga clic en él para ver la lista de acciones ejecutadas.

Ventana emergente del evento del problema

La ventana emergente del evento del problema incluye la lista de eventos de problema para este trigger y, si está definido, la descripción del trigger y una URL en la que se puede hacer clic.



Para mostrar la ventana emergente del evento del problema:

- Pase el cursor del mouse sobre la duración del problema en la columna *Duration* del widget *Problems*. La ventana emergente desaparece en cuanto retire el mouse de la duración.
- Haga clic en la duración en la columna *Duration* del widget *Problems*. La ventana emergente desaparece solo si vuelve a hacer clic en la duración.

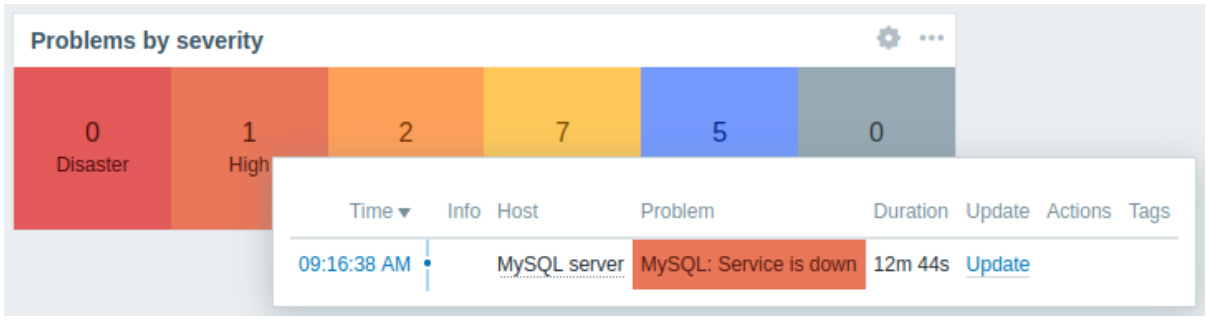
Problemas por severidad

Descripción general

El widget *Problemas por gravedad* muestra el número de problemas causa agrupados por gravedad. Permite filtrar los problemas por diferentes parámetros (grupos de equipos, equipos, nombres de problemas, etc.).

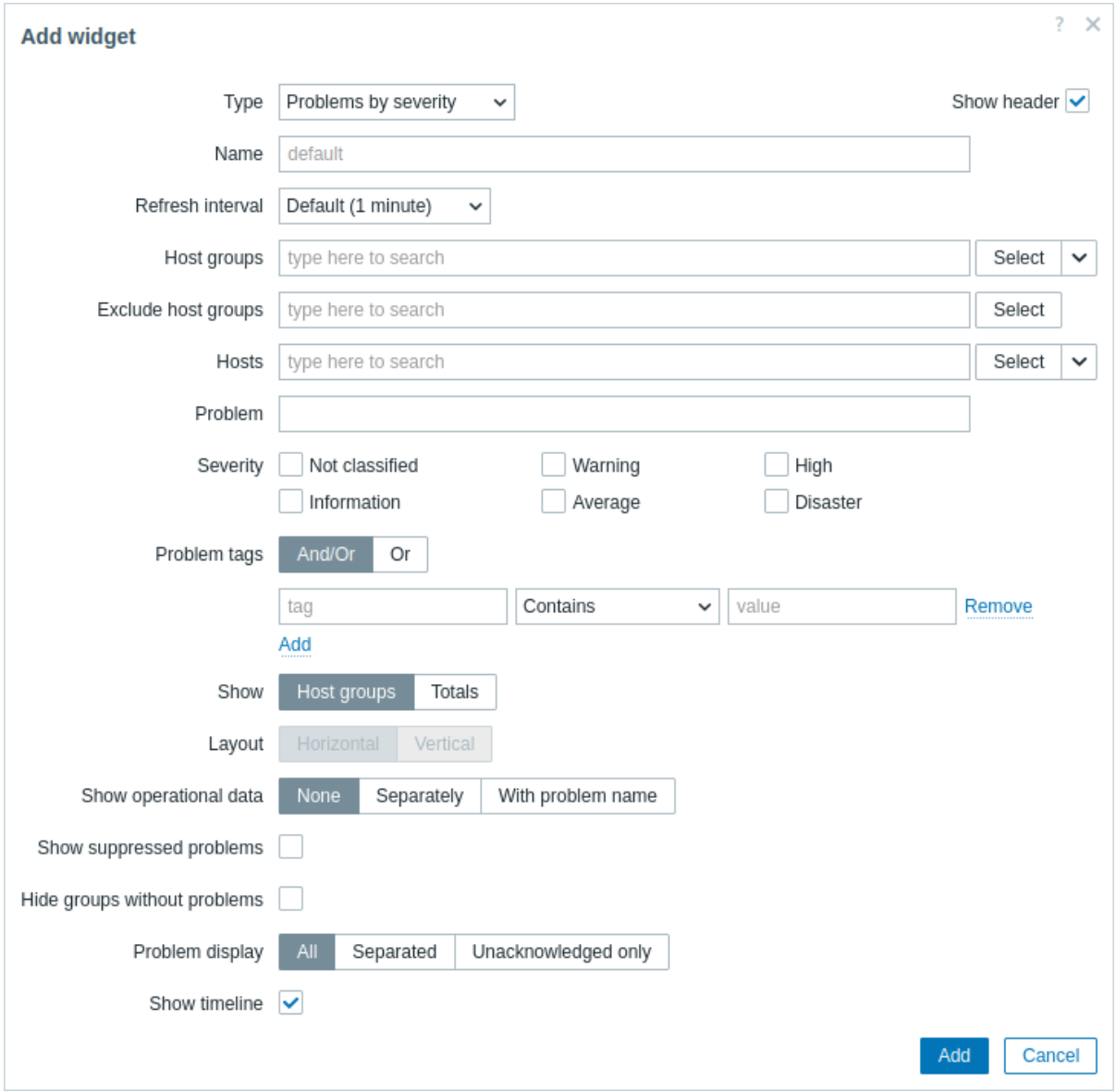


Al pasar el cursor sobre un recuento de problemas se muestra la lista de problemas relacionados:



Configuración

Para configurar, seleccione *Problems by severity* como tipo:



Además de los parámetros comunes para todos los widgets, puede definir las siguientes opciones específicas:

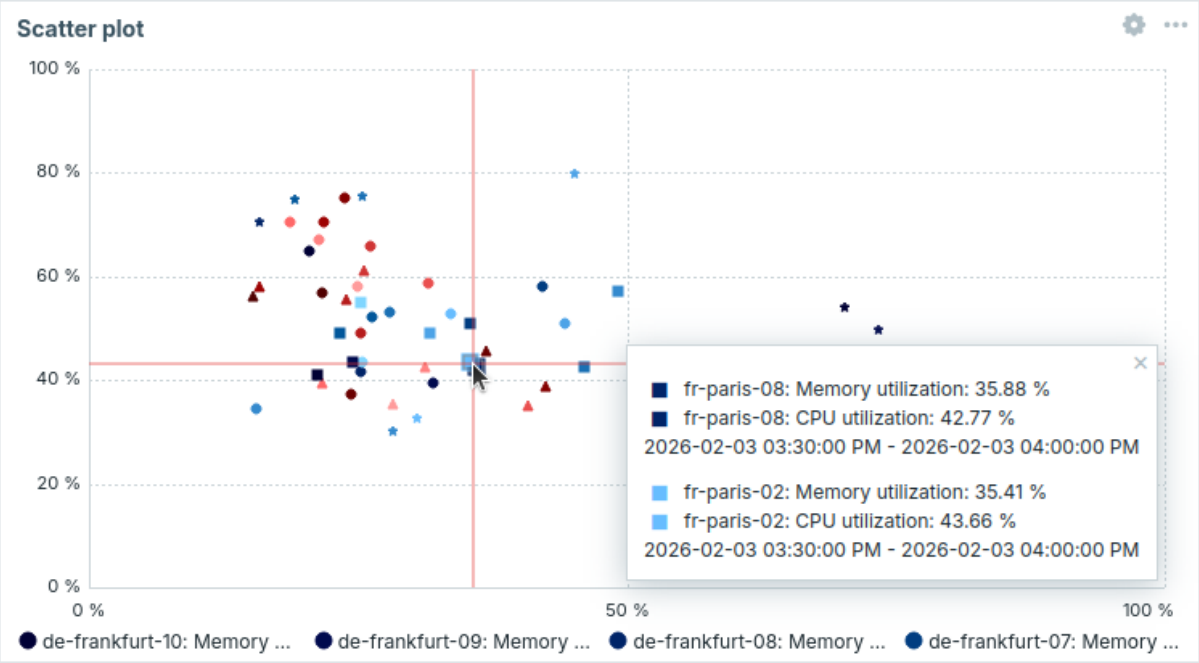
Host groups	Seleccione los host groups que se mostrarán en el widget. Alternativamente, seleccione un widget compatible como fuente de datos para host groups. Este campo tiene autocompletado, por lo que al empezar a escribir el nombre de un grupo se mostrará una lista desplegable con los grupos coincidentes. Especificar un host group padre selecciona implícitamente todos los host groups anidados. Los datos de host de estos host groups se mostrarán en el widget; si no se introduce ningún host group, se mostrarán todos los host groups. Este parámetro no está disponible al configurar el widget en un template dashboard.
Exclude host groups	Seleccione los host groups que se ocultarán en el widget. Este campo tiene autocompletado, por lo que al empezar a escribir el nombre de un grupo se mostrará una lista desplegable con los grupos coincidentes. Especificar un host group padre selecciona implícitamente todos los host groups anidados. Los datos de host de estos host groups no se mostrarán en el widget. Por ejemplo, los hosts 001, 002 y 003 pueden estar en el Grupo A, y los hosts 002 y 003 también en el Grupo B. Si seleccionamos *mostrar* el Grupo A y *excluir* el Grupo B al mismo tiempo, solo se mostrarán los datos del host 001 en el dashboard. Este parámetro no está disponible al configurar el widget en un template dashboard.

Hosts	Seleccione los hosts que se mostrarán en el widget. Alternativamente, seleccione un widget compatible o el dashboard como fuente de datos para hosts. Este campo tiene autocompletado, por lo que al empezar a escribir el nombre de un host se mostrará una lista desplegable con los hosts coincidentes. Si no se introduce ningún host, se mostrarán todos los hosts. Este parámetro no está disponible al configurar el widget en un template dashboard.
Problem	Puede limitar el número de hosts con problemas mostrados por el nombre del problema. Si introduce una cadena aquí, solo se mostrarán los hosts con problemas cuyo nombre contenga la cadena introducida. Las macros no se expanden.
Severity	Marque las severidades de problema para filtrar los problemas que se mostrarán en el widget. Si no se marca ninguna severidad, se mostrarán todos los problemas.
Problem tags	Especifique etiquetas de problema para limitar el número de problemas mostrados en el widget. Es posible incluir y excluir etiquetas y valores de etiqueta específicos. Se pueden establecer varias condiciones. La coincidencia del nombre de la etiqueta siempre distingue entre mayúsculas y minúsculas. Hay varios operadores disponibles para cada condición: **Exists** - incluye los nombres de etiqueta especificados; **Equals** - incluye los nombres y valores de etiqueta especificados (distingue entre mayúsculas y minúsculas); **Contains** - incluye los nombres de etiqueta especificados cuyos valores de etiqueta contienen la cadena introducida (coincidencia de subcadena, no distingue entre mayúsculas y minúsculas); **Does not exist** - excluye los nombres de etiqueta especificados; **Does not equal** - excluye los nombres y valores de etiqueta especificados (distingue entre mayúsculas y minúsculas); **Does not contain** - excluye los nombres de etiqueta especificados cuyos valores de etiqueta contienen la cadena introducida (coincidencia de subcadena, no distingue entre mayúsculas y minúsculas). Hay dos tipos de cálculo para las condiciones: **And/Or** - todas las condiciones deben cumplirse; las condiciones que tengan el mismo nombre de etiqueta se agruparán mediante la condición *Or*; **Or** - basta con que se cumpla una condición.
Show	Seleccione la opción de visualización: **Host groups** - muestra los problemas por host group; **Totals** - muestra un total de problemas para todos los host groups seleccionados en bloques de color correspondientes a la severidad del problema. Este parámetro no está disponible al configurar el widget en un template dashboard, y solo se mostrará un total de problemas.
Layout	Seleccione la opción de diseño: **Horizontal** - los bloques de color de los totales se mostrarán horizontalmente; **Vertical** - los bloques de color de los totales se mostrarán verticalmente. Este parámetro está disponible si *Show* está configurado como ”Totals”.
Show operational data	Mostrar los datos operativos como: **None** - no se mostrarán datos operativos **Separately** - los datos operativos se mostrarán como una línea separada **With problem name** - los datos operativos se mostrarán junto con el nombre del problema. Para más detalles, consulte la descripción del campo *Operational data* en la configuración de trigger.
Show suppressed problems	Marque la casilla para mostrar los problemas que, de otro modo, estarían suprimidos (no mostrados) debido al mantenimiento del host.
Hide groups without problems	Marque la opción *Hide groups without problems* para ocultar en el widget los datos de los host groups sin problemas. Este parámetro no está disponible al configurar el widget en un template dashboard.
Problem display	Mostrar el recuento de problemas como: **All** - se mostrará el recuento completo de problemas; **Separated** - se mostrará el recuento de problemas no reconocidos separado como un número del recuento total de problemas; **Unacknowledged only** - solo se mostrará el recuento de problemas no reconocidos.
Show timeline	Marque la casilla para mostrar una línea de tiempo visual.

Gráfico de dispersión

Descripción general

El widget *Gráfico de dispersión* muestra la relación entre dos métricas mediante la representación de puntos de datos individuales en los ejes X e Y. Esto ayuda a revelar patrones, agrupaciones, correlaciones y valores atípicos en el conjunto de datos.



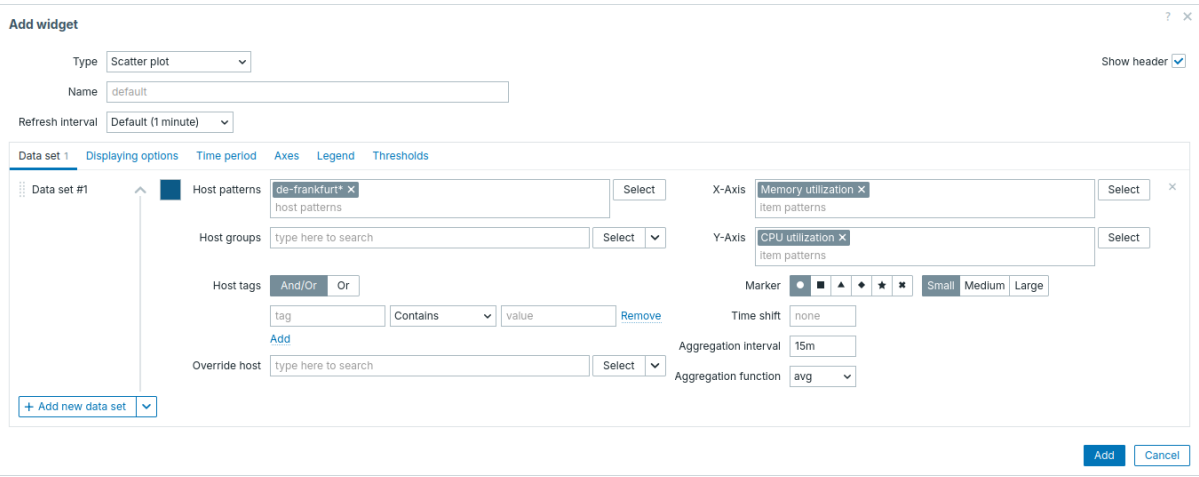
Al pasar el cursor sobre un punto de datos, se muestra una información emergente con una lista de los items de ese punto de datos y de otros puntos situados dentro de un radio de 6 px; primero se ordenan por los valores del eje X del gráfico de dispersión en orden ascendente y, después, por los valores del eje Y en orden descendente.

Al seleccionar un item en la información emergente, sus datos se transmiten a otros widgets.

La información mostrada en el widget *Gráfico de dispersión* se puede descargar como una imagen PNG seleccionando la opción *Descargar imagen* en el menú *Acciones* del widget ..., que aparece al pasar el cursor sobre el widget. Según la configuración del navegador, la imagen se guarda automáticamente en la carpeta `Downloads` o se abre un cuadro de diálogo *Guardar*, que permite elegir la ubicación.

Configuración

Para configurar, seleccione *Gráfico de dispersión* como tipo:



Conjunto de datos

La pestaña **Conjunto de datos** permite seleccionar datos para el gráfico de dispersión mediante la adición de conjuntos de datos. Se pueden añadir dos tipos de conjuntos de datos:

- *Patrones de item*: los hosts con items coincidentes se muestran como puntos de datos. Puede elegir un color base único o seleccionar una fila de paleta para asignar colores distintos a cada host con los items coincidentes.
- *Lista de items*: los items seleccionados se muestran como puntos de datos, donde cada punto representa un par de items (X e Y). Puede elegir individualmente el color de cada conjunto de datos mediante el selector.

Los puntos de datos se crean únicamente para los pares de hosts/items que tienen al menos un valor en ambos ejes.

De forma predeterminada, se añade un conjunto de datos de tipo *Patrones de item*.

Conjunto de datos	Para el conjunto de datos **Patrones de item**: Seleccione o introduzca patrones de host e item; los datos de los items que coincidan con estos patrones se mostrarán en el gráfico de dispersión. Tenga en cuenta que, si el número de puntos de datos supera los 8k, se debe aumentar `memory_limit` en el archivo de configuración `php.ini`. El límite de memoria recomendado para 20k puntos de datos es de 512 MB. Se pueden utilizar patrones comodín para la selección (por ejemplo, ∗ devolverá resultados que coincidan con cero o más caracteres). Para especificar un patrón comodín, introduzca la cadena manualmente y pulse *Enter*. El símbolo comodín siempre se interpreta, por lo que no es posible añadir individualmente, por ejemplo, un item denominado *item** si existen otros items coincidentes (por ejemplo, *item2*, *item3*). Es obligatorio especificar patrones de host e item para los conjuntos de datos de tipo ”Patrones de item”. Consulte también: Detalles de configuración del conjunto de datos. Para el conjunto de datos **Lista de items**: Seleccione items para el gráfico de dispersión haciendo clic en el botón *Añadir item*. También puede seleccionar widgets compatibles como fuente de datos para los items haciendo clic en el botón *Añadir widget*. Es obligatorio especificar items o widgets para los conjuntos de datos de tipo ”Lista de items”. Consulte también: Detalles de configuración del conjunto de datos. Tenga en cuenta que solo se permiten tipos de item numéricos. Al configurar el widget en un dashboard de template, el parámetro para especificar patrones de host no está disponible, y el parámetro para especificar una lista de items solo permite seleccionar los items configurados en el template.
Grupos de hosts	Seleccione grupos de hosts. Como alternativa, seleccione un widget compatible como fuente de datos para los grupos de hosts. Este campo ofrece autocompletado, por lo que al comenzar a escribir el nombre de un grupo se mostrará una lista desplegable con los grupos coincidentes. Especificar un grupo de hosts principal selecciona implícitamente todos los grupos de hosts anidados. Los datos de los hosts de estos grupos de hosts se mostrarán en el widget; si no se introducen grupos de hosts, se mostrarán todos los grupos de hosts. Este parámetro solo está disponible para los conjuntos de datos de tipo *Patrones de item* y no está disponible al configurar el widget en un dashboard de template.

Etiquetas de host	Especifique etiquetas para filtrar los hosts mostrados en el widget. Es posible incluir y excluir etiquetas y valores de etiqueta específicos. Se pueden establecer varias condiciones. La coincidencia de nombres de etiqueta siempre distingue entre mayúsculas y minúsculas. Hay varios operadores disponibles para cada condición: **Existe**: incluye los nombres de etiqueta especificados; **Es igual a**: incluye los nombres y valores de etiqueta especificados (distingue entre mayúsculas y minúsculas); **Contiene**: incluye los nombres de etiqueta especificados cuyos valores contienen la cadena introducida (coincidencia de subcadena, sin distinguir entre mayúsculas y minúsculas); **No existe**: excluye los nombres de etiqueta especificados; **No es igual a**: excluye los nombres y valores de etiqueta especificados (distingue entre mayúsculas y minúsculas); **No contiene**: excluye los nombres de etiqueta especificados cuyos valores contienen la cadena introducida (coincidencia de subcadena, sin distinguir entre mayúsculas y minúsculas). Hay dos tipos de cálculo para las condiciones: **Y/O**: se deben cumplir todas las condiciones; las condiciones que tienen el mismo nombre de etiqueta se agruparán mediante la condición O; **O**: basta con que se cumpla una condición. Este parámetro solo está disponible para los conjuntos de datos de tipo *Patrones de item* y no está disponible al configurar el widget en un dashboard de template.
Anular host	Seleccione un widget compatible o el selector de host del dashboard como fuente de datos para los hosts. Este parámetro no está disponible al configurar el widget en un dashboard de template.
Marcador	Elija el tipo y el tamaño del marcador para los puntos de datos de los hosts en el conjunto de datos.
Desplazamiento temporal	Especifique el desplazamiento temporal si es necesario. Puede utilizar sufijos de tiempo en este campo. Se permiten valores negativos.
Función de agregación	Especifique qué función de agregación se utilizará para cada item del conjunto de datos dentro del *Intervalo de agregación* seleccionado: **min**: muestra el valor más pequeño; **max**: muestra el valor más grande; **avg**: muestra el valor promedio; **sum**: muestra la suma de los valores; **count**: muestra el número de valores (si existe al menos un valor en ambos ejes); **first**: muestra el primer valor; **last**: muestra el último valor. La agregación permite mostrar un valor agregado para el intervalo especificado (5 minutos, una hora, un día), en lugar de todos los valores. Solo se pueden mostrar datos numéricos para *min*, *max*, *avg* y *sum*. Para *count*, los datos no numéricos se convertirán a numéricos.

Intervalo de agregación	Especifique el intervalo para agregar los valores. Puede utilizar sufijos de tiempo en este campo. Un valor numérico sin sufijo se considerará en segundos. Tenga en cuenta que, si el widget está configurado para mostrar datos históricos basados en tendencias (*Selección de datos históricos* está establecido en *Tendencias* o *Automático*), se recomienda utilizar un intervalo de agregación que sea múltiplo de 1 hora (por ejemplo, 3600, 60m, 1h, 3h, etc.). Las tendencias almacenan valores agregados por hora, por lo que utilizar un intervalo de agregación que no sea múltiplo de 1 hora (por ejemplo, 100s, 7min, 15min, 90min, etc.) puede producir resultados difíciles de interpretar. Es obligatorio especificar un intervalo de agregación. El intervalo de agregación determina el intervalo de tiempo y las coordenadas de un punto de datos; por ejemplo, si los items del eje X tienen 4 valores y los items del eje Y tienen 1 valor, las coordenadas reflejan la agregación de 4 valores en X y 1 en Y dentro del intervalo especificado.

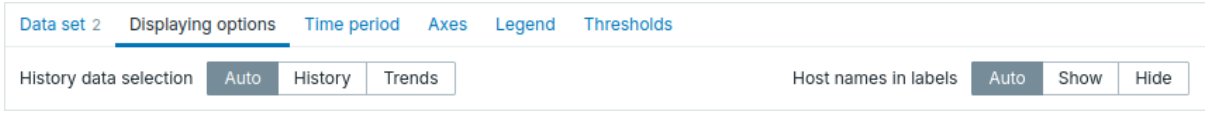
Detalles de la configuración del conjunto de datos

Los conjuntos de datos existentes se muestran en una lista. Puede reorganizar, expandir/colapsar, cambiar colores y clonar estos conjuntos de datos.

Para obtener más información, consulte los detalles de la configuración del conjunto de datos en el widget *Gráfico*. Estos detalles también se aplican al widget *Gráfico de dispersión*.

Opciones de visualización

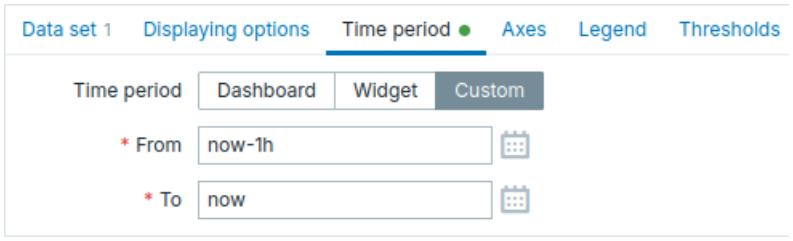
La pestaña **Opciones de visualización** permite definir la selección de datos históricos:



Selección de datos históricos	Establezca el origen de los datos del gráfico de dispersión: **Auto**: los datos se obtienen según el algoritmo del gráfico clásico (predeterminado). **History**: datos del historial. **Trends**: datos de las tendencias.
Nombres de host en las etiquetas	Elija si desea mostrar los nombres de los host en la leyenda del gráfico y en la información sobre herramientas que aparece al pasar el cursor sobre los datos del gráfico: **Auto**: los nombres de los host se muestran solo si hay más de un host en todos los conjuntos de datos (predeterminado). **Show**: se muestran los nombres de los host. **Hide**: se ocultan los nombres de los host. Este parámetro no está disponible al configurar el widget en un dashboard de template.

Período de tiempo

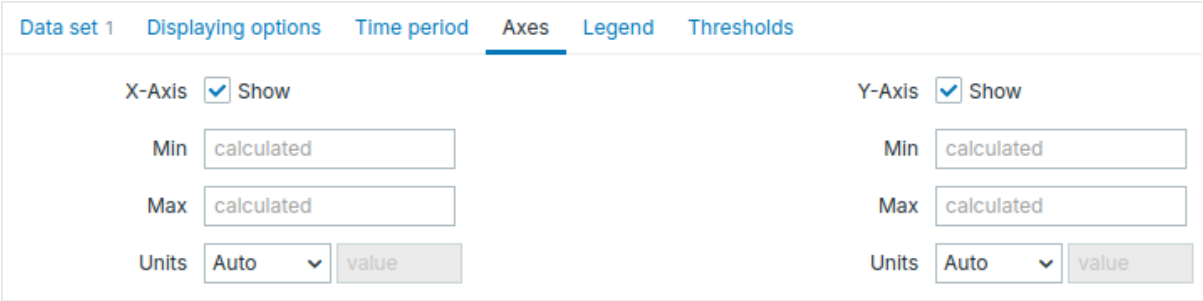
La pestaña **Período de tiempo** permite establecer el período de tiempo para el que se mostrarán los datos en el gráfico de dispersión:



Período de tiempo	Seleccione la fuente de datos para el período de tiempo: **Dashboard**: utiliza el selector de período de tiempo del dashboard; **Widget**: utiliza un widget compatible (establecido en el parámetro *Widget*); **Personalizado**: utiliza un período de tiempo personalizado establecido en los parámetros *Desde* y *Hasta*; si se establece, se mostrará un icono de reloj en la esquina superior derecha del widget, que indicará la hora establecida al pasar el ratón por encima. Tenga en cuenta que, independientemente de la configuración de *Período de tiempo* del widget, los widgets compatibles aún pueden utilizarlo como fuente de datos para el período de tiempo.
Widget	Introduzca o seleccione un widget compatible como fuente de datos para el período de tiempo. Este parámetro está disponible si *Período de tiempo* está establecido en "Widget".
Desde	Introduzca o seleccione el inicio del período de tiempo. Se admite la sintaxis de tiempo relativo (`now`, `now/d`, `now/w-1w`, etc.). Este parámetro está disponible si *Período de tiempo* está establecido en "Personalizado".
Hasta	Introduzca o seleccione el final del período de tiempo. Se admite la sintaxis de tiempo relativo (`now`, `now/d`, `now/w-1w`, etc.). Este parámetro está disponible si *Período de tiempo* está establecido en "Personalizado".

Ejes

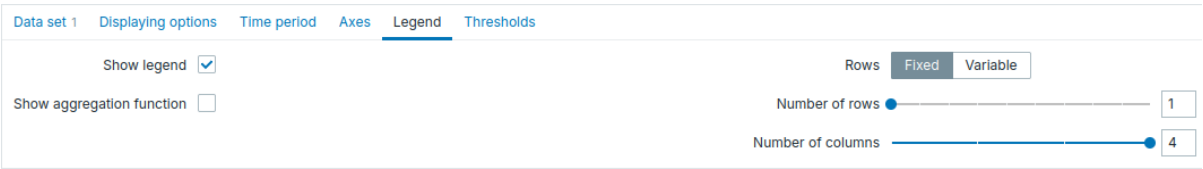
La pestaña **Ejes** permite personalizar cómo se muestran los ejes:



Eje X/Eje Y	Desmarque esta casilla para ocultar el eje X/eje Y (marcada de forma predeterminada).
Mín.	Establezca el valor mínimo del eje correspondiente.
Máx.	Establezca el valor máximo del eje correspondiente.
Unidades	Seleccione la unidad para los valores de los ejes del gráfico de dispersión en la lista desplegable: **Automático**: los valores de los ejes se muestran utilizando la unidad del primer item del conjunto de datos; **Estático**: los valores de los ejes se muestran utilizando la unidad especificada en el campo de entrada *valor*; si el campo se deja en blanco, solo se muestran valores numéricos.

Leyenda

La pestaña **Leyenda** permite personalizar la leyenda del gráfico de dispersión:

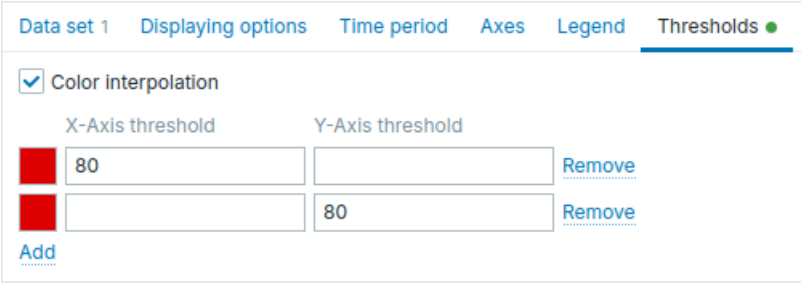


Mostrar leyenda	Desmarque esta casilla para ocultar la leyenda en el gráfico de dispersión (marcada de forma predeterminada).
Mostrar función de agregación	Marque esta casilla para mostrar la función de agregación en la leyenda.
Filas	Seleccione el modo de visualización de las filas de la leyenda: **Fijo**: el número de filas mostradas está determinado por el valor del parámetro *Número de filas*; **Variable**: el número de filas mostradas está determinado por la cantidad de items configurados, sin superar el valor del parámetro *Número máximo de filas*.

Número de filas/ Número máximo de filas	Si *Filas* está establecido en «Fijo», establezca el número de filas de leyenda que se mostrarán (1-10). Si *Filas* está establecido en «Variable», establezca el número máximo de filas de leyenda que se mostrarán (1-10).
Número de columnas	Establezca el número de columnas de leyenda que se mostrarán (1-4). Este parámetro está disponible si *Mostrar mín./prom./máx.* no está marcado.

Umbrales

La pestaña **Umbrales** permite configurar los umbrales que determinan los colores de los marcadores de los host:

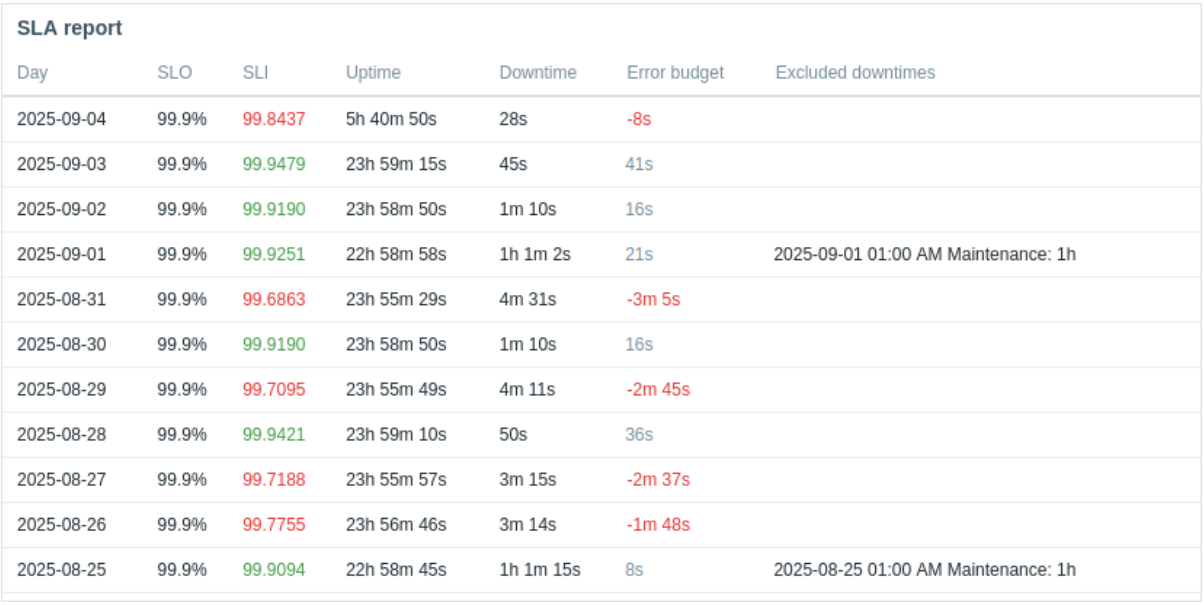


Interpolación de colores	Marque la casilla para habilitar una transición gradual entre los colores de los umbrales para los marcadores de los host. Este parámetro se aplica si se configuran dos o más umbrales para el eje X, el eje Y o ambos ejes simultáneamente.
Umbral del eje X/eje Y	Haga clic en *Añadir* para añadir un umbral, seleccione un color para el umbral en el selector de colores y especifique un valor numérico. Los umbrales se pueden aplicar a un solo eje, a ambos ejes por separado o a ambos ejes simultáneamente. Por ejemplo, un único umbral con X=80 e Y=80 coloreará los marcadores cuando se cumplan ambas condiciones. La lista de umbrales se ordenará en orden ascendente al guardarla. Se admiten sufijos (por ejemplo, ”1d”, ”2w”, ”4K”, ”8G”). Se admiten las asignaciones de valores.

Informe de SLA

Descripción general

El widget *Informe SLA* muestra informes SLA, ayudándole a supervisar el rendimiento del servicio frente a los objetivos definidos.

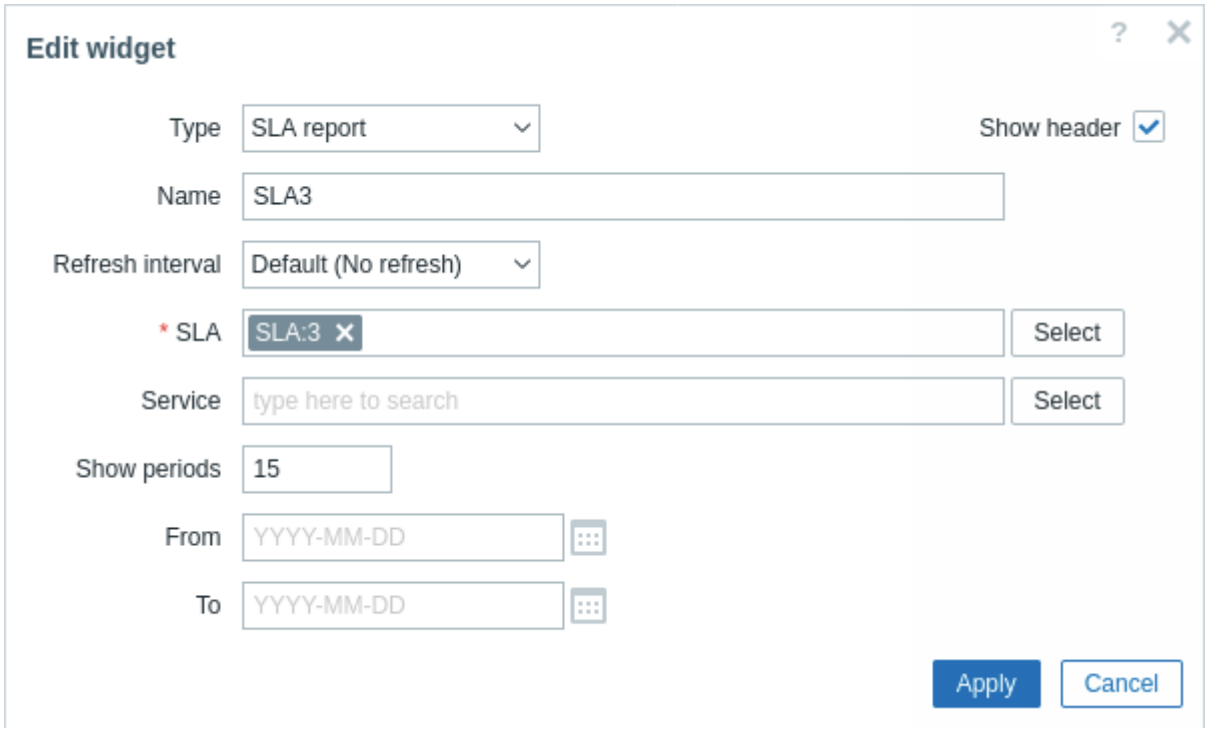

SLA report

Day	SLO	SLI	Uptime	Downtime	Error budget	Excluded downtimes
2025-09-04	99.9%	99.8437	5h 40m 50s	28s	-8s	
2025-09-03	99.9%	99.9479	23h 59m 15s	45s	41s	
2025-09-02	99.9%	99.9190	23h 58m 50s	1m 10s	16s	
2025-09-01	99.9%	99.9251	22h 58m 58s	1h 1m 2s	21s	2025-09-01 01:00 AM Maintenance: 1h
2025-08-31	99.9%	99.6863	23h 55m 29s	4m 31s	-3m 5s	
2025-08-30	99.9%	99.9190	23h 58m 50s	1m 10s	16s	
2025-08-29	99.9%	99.7095	23h 55m 49s	4m 11s	-2m 45s	
2025-08-28	99.9%	99.9421	23h 59m 10s	50s	36s	
2025-08-27	99.9%	99.7188	23h 55m 57s	3m 15s	-2m 37s	
2025-08-26	99.9%	99.7755	23h 56m 46s	3m 14s	-1m 48s	
2025-08-25	99.9%	99.9094	22h 58m 45s	1h 1m 15s	8s	2025-08-25 01:00 AM Maintenance: 1h

Muestra los mismos datos que *Servicios* > *Informe SLA*.

Configuración

Para configurar, seleccione *Informe SLA* como tipo:



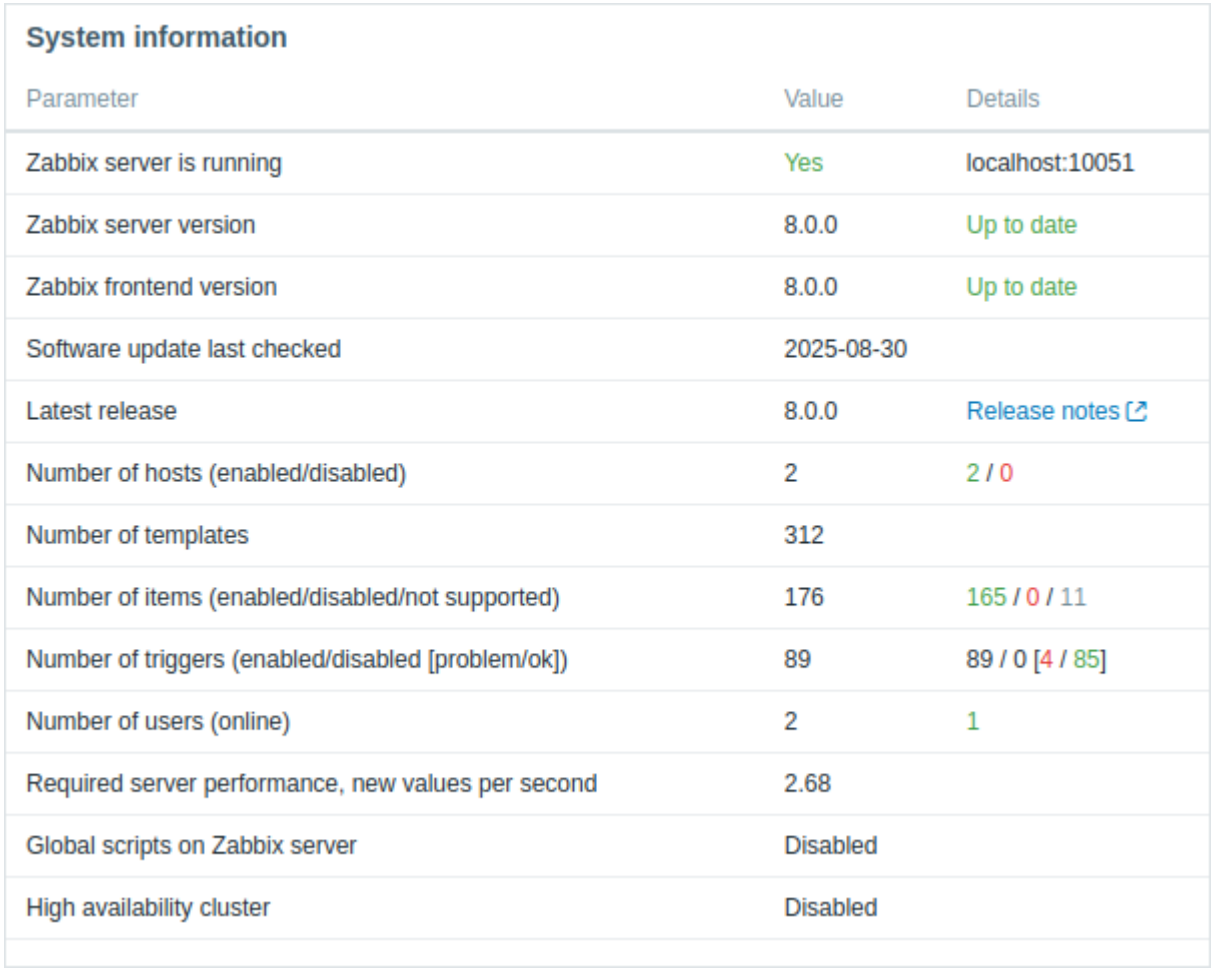
Además de los parámetros que son comunes para todos los widgets, puede establecer las siguientes opciones específicas:

SLA	Seleccione el SLA para el informe.
Servicio	Seleccione el servicio para el informe.
Mostrar periodos	Establezca cuántos periodos se mostrarán en el widget (20 por defecto, 100 como máximo).
Desde	Seleccione la fecha de inicio para el informe. Se admiten fechas relativas: `now`, `now/d`, `now/w-1w`, etc.; modificadores de fecha admitidos: d, w, M, y.
Hasta	Seleccione la fecha de finalización para el informe. Se admiten fechas relativas: `now`, `now/d`, `now/w-1w`, etc.; modificadores de fecha admitidos: d, w, M, y.

Información del sistema

Resumen

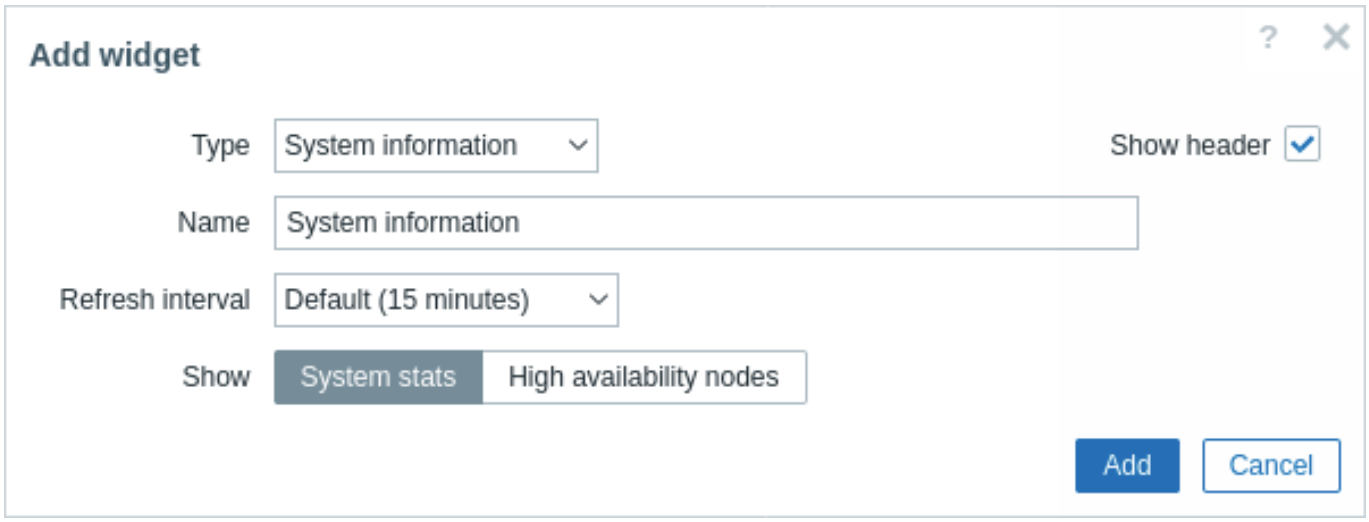
El widget *System information* muestra un resumen de los datos clave del server Zabbix y del sistema y, cuando la alta disponibilidad está habilitada, detalles de los nodos de alta disponibilidad.



Muestra los mismos datos que *Reports* > *System information*, excepto *Zabbix server ID*.

Configuración

Para configurar, seleccione *Información del sistema* como tipo:



Además de los parámetros comunes a todos los widgets, puede establecer las siguientes opciones específicas:

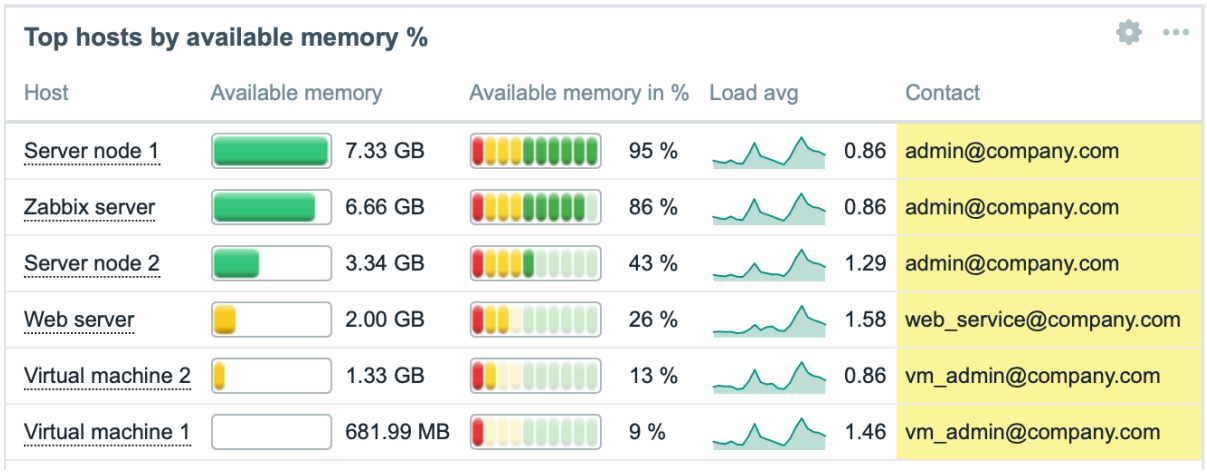
Mostrar	Seleccione qué desea mostrar: **Estadísticas del sistema**: muestra un resumen de los datos principales del server de Zabbix y del sistema; **Nodos de alta disponibilidad**: muestra el estado de los nodos de alta disponibilidad (si el clúster de alta disponibilidad está habilitado).

Mostrar detalles de la comprobación de actualización del software	Marque la casilla para mostrar los detalles de la comprobación de actualización del software de Zabbix. Esta opción solo está disponible si la comprobación de actualización del software está habilitada en la configuración del server de Zabbix y se ha seleccionado "Estadísticas del sistema" en el campo *Mostrar*.

Hosts principales

Descripción general

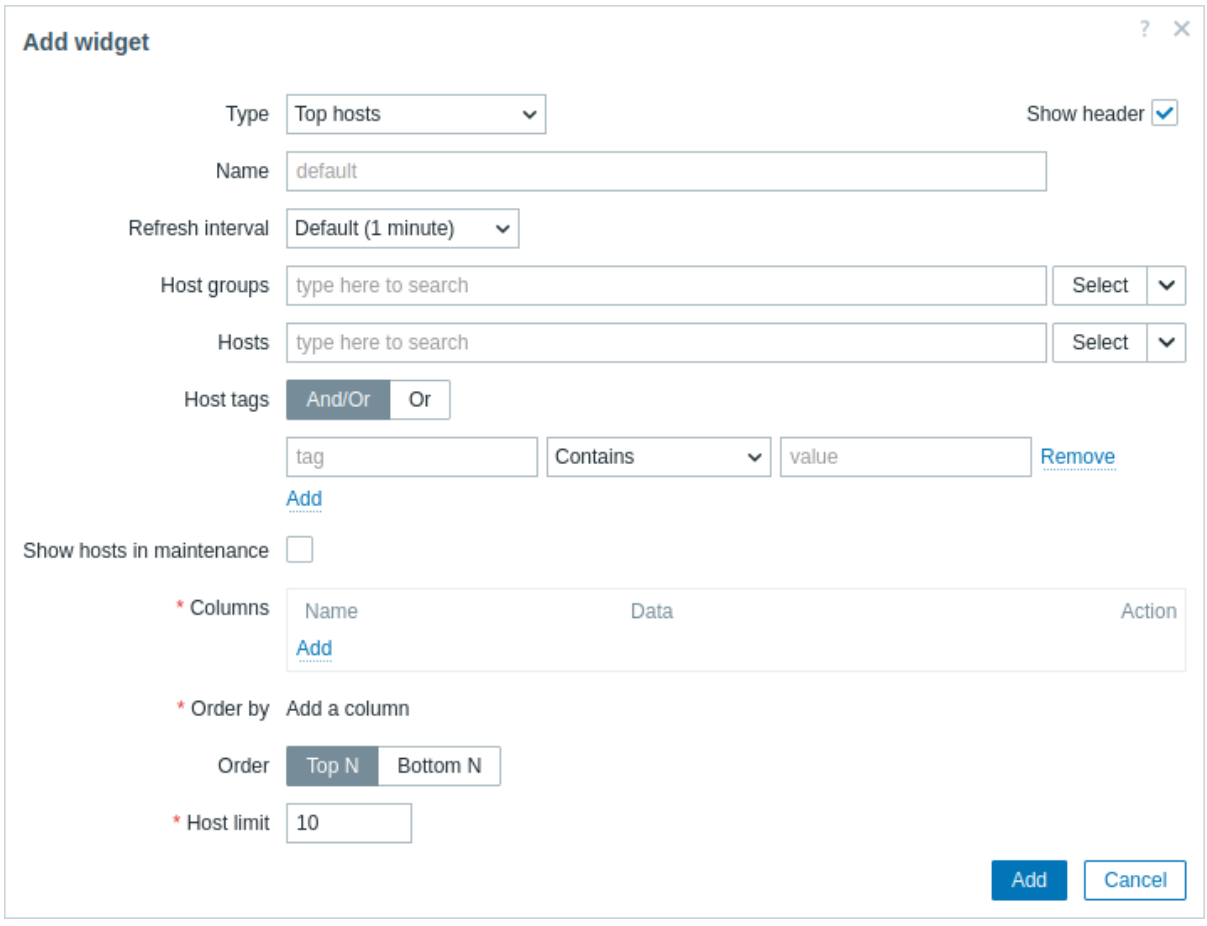
El widget *Principales equipos* muestra los valores más altos/bajos para los elementos seleccionados de varios equipos.



El widget admite la visualización de hasta 1000 equipos simultáneamente.

Configuración

Para configurar, seleccione *Equipos principales* como tipo:



Además de los parámetros que son comunes para todos los widgets, puede configurar las siguientes opciones específicas:

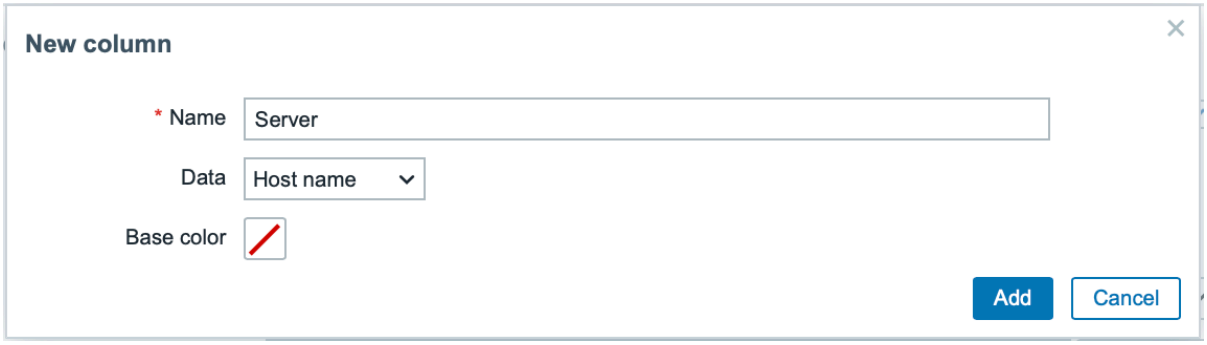
Grupos de equipos	Seleccione los grupos de equipos que se mostrarán en el widget. Alternativamente, seleccione un widget compatible como fuente de datos para los grupos de equipos. Este campo es autocompletado, por lo que al comenzar a escribir el nombre de un grupo se mostrará un desplegable con los grupos coincidentes. Este parámetro no está disponible al configurar el widget en un dashboard de plantilla.
Equipos	Seleccione los equipos que se mostrarán en el widget. Alternativamente, seleccione un widget compatible o el dashboard como fuente de datos para los equipos. Este campo es autocompletado, por lo que al comenzar a escribir el nombre de un equipo se mostrará un desplegable con los equipos coincidentes. Este parámetro no está disponible al configurar el widget en un dashboard de plantilla.
Etiquetas de equipo	Especifique etiquetas para limitar el número de equipos mostrados en el widget. Es posible incluir así como excluir etiquetas y valores de etiquetas específicos. Se pueden establecer varias condiciones. La coincidencia del nombre de la etiqueta siempre distingue entre mayúsculas y minúsculas. Hay varios operadores disponibles para cada condición: **Existe** - incluir los nombres de etiqueta especificados; **Igual a** - incluir los nombres y valores de etiqueta especificados (distingue mayúsculas y minúsculas); **Contiene** - incluir los nombres de etiqueta especificados donde los valores de etiqueta contienen la cadena introducida (coincidencia de subcadena, no distingue mayúsculas y minúsculas); **No existe** - excluir los nombres de etiqueta especificados; **No igual a** - excluir los nombres y valores de etiqueta especificados (distingue mayúsculas y minúsculas); **No contiene** - excluir los nombres de etiqueta especificados donde los valores de etiqueta contienen la cadena introducida (coincidencia de subcadena, no distingue mayúsculas y minúsculas). Hay dos tipos de cálculo para las condiciones: **Y/O** - se deben cumplir todas las condiciones, las condiciones con el mismo nombre de etiqueta se agruparán por la condición O; **O** - basta con que se cumpla una condición. Este parámetro no está disponible al configurar el widget en un dashboard de plantilla.
Mostrar equipos en mantenimiento	Marque esta casilla para que los equipos en mantenimiento también se muestren (en este caso, se mostrará un icono de mantenimiento junto al nombre del equipo). Por defecto, no está marcada.
Columnas	Agregue columnas de datos para mostrar. El orden de las columnas determina su visualización de izquierda a derecha. Las columnas se pueden reordenar arrastrándolas hacia arriba o hacia abajo usando el manejador antes del nombre de la columna.
Ordenar por	Especifique la columna de la lista de *Columnas* definida que se utilizará para el orden *Top N* o *Bottom N*.
Orden	Especifique el orden de las filas: **Top N** - en orden descendente según el valor agregado de *Ordenar por*; **Bottom N** - en orden ascendente según el valor agregado de *Ordenar por*.
Límite de equipos	Número de filas de equipos a mostrar (1-1000). Este parámetro no está disponible al configurar el widget en un dashboard de plantilla.

Configuración de columnas

Es posible añadir columnas de tres tipos de datos: nombre de host, valor de item o texto. La lista de parámetros de columna disponibles depende del tipo de dato de la columna y, para el tipo valor de item, del formato del valor.

Columna de nombre de equipo

La columna de nombre de equipo se utiliza para mostrar el nombre del equipo.

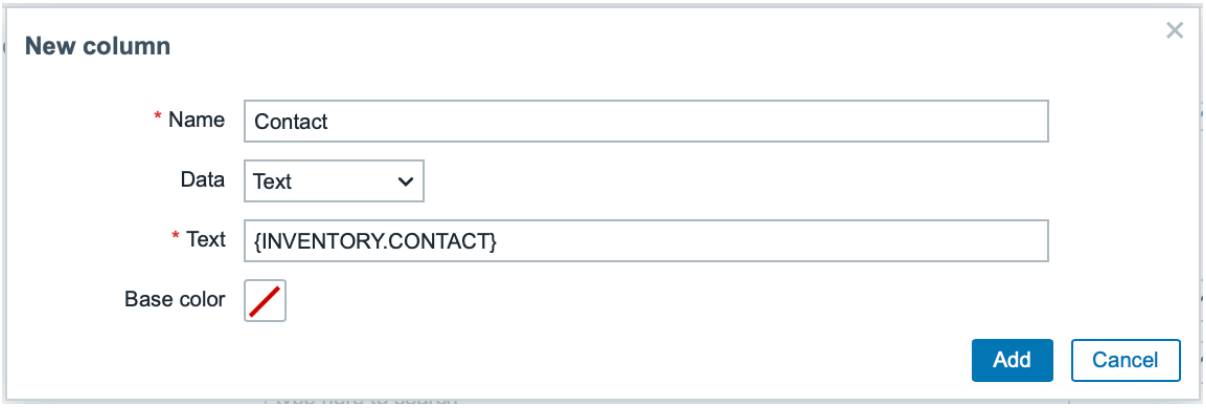


Parámetros soportados:

Nombre	Nombre de la columna.
Datos	Tipo de dato a mostrar en la columna, seleccione **Nombre de equipo**.
Color base	Color de fondo de la columna.

Columna de texto

La columna de texto se utiliza para mostrar cualquier cadena de texto especificada.

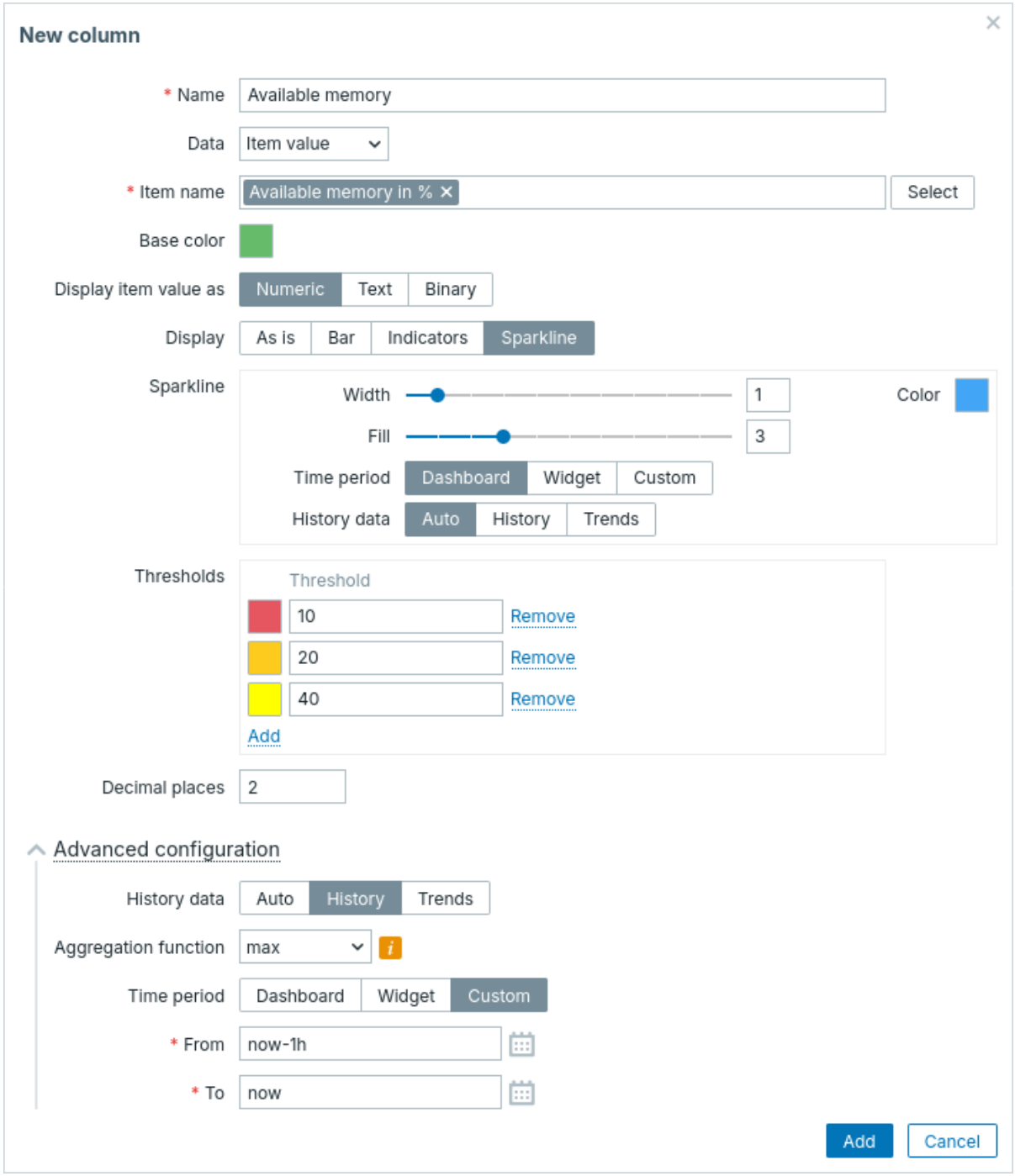


Parámetros soportados:

Nombre	Nombre de la columna.
Datos	Tipo de datos a mostrar en la columna, seleccione **Texto**.
Color base	Color de fondo de la columna.
Texto	Introduzca la cadena a mostrar. Puede contener macros de host e inventario macros.

Columna de valor del item

La columna de valor del item se utiliza para mostrar el valor del item especificado; admite varios formatos y opciones de visualización de valores.



Parámetros compatibles:

Name	Nombre de la columna.
Data	Tipo de datos que se mostrarán en la columna; seleccione **Item value**.
Item name	Seleccione un item; su nombre se utiliza para buscar y mostrar items con el mismo nombre en todos los hosts seleccionados. Si un host tiene varios items de este tipo, el widget muestra el item cuya clave aparece primero alfabéticamente (por ejemplo, *CPU utilization* con la clave `proc.cpu.util` en lugar de `system.cpu.util`). Al configurar el widget en un dashboard de template, solo se pueden seleccionar los items configurados en el template.
Base color	Color de fondo de la columna; color de relleno si los datos de *Item value* se muestran como barra o indicadores. El color predeterminado puede sustituirse por un color personalizado si el valor del item supera uno de los umbrales especificados.
Display item value as	Formato para mostrar el valor del item: **Numeric**, **Text** o **Binary**. La opción seleccionada determina qué parámetros adicionales estarán disponibles. Consulte la lista de parámetros de cada formato a continuación.
Advanced configuration	

History data	Tomar los datos del historial o de las tendencias: **Auto** - selección automática; **History** - tomar los datos del historial; **Trends** - tomar los datos de las tendencias. Este parámetro solo está disponible para valores numéricos de items. Para valores de texto y binarios, los datos siempre se tomarán del historial.
Aggregation function	Especifique qué función de agregación se utilizará dentro del *Time period* seleccionado: **min** - mostrar el valor más pequeño; **max** - mostrar el valor más grande; **avg** - mostrar el valor promedio; **count** - mostrar el número de valores; **sum** - mostrar la suma de los valores; **first** - mostrar el primer valor; **last** - mostrar el último valor. Si se selecciona **not used**, se muestra el valor más reciente (sin agregación). La agregación permite mostrar un valor agregado para el intervalo especificado (5 minutos, una hora, un día), en lugar del valor más reciente. Solo se pueden mostrar datos numéricos para *min*, *max*, *avg* y *sum*. Para *count*, los datos no numéricos se convertirán a numéricos.
Time period	Especifique el período que se utilizará para agregar los valores. Seleccione la fuente de datos para el período: **Dashboard** - utilizar el selector de período del dashboard; **Widget** - utilizar un widget compatible (establecido en el parámetro *Widget*); **Custom** - utilizar un período personalizado establecido en los parámetros *From* y *To*; si se establece, se mostrará un icono de reloj en la esquina superior derecha del widget, que indicará el tiempo establecido al pasar el ratón por encima. Tenga en cuenta que, independientemente de la configuración de *Time period* del widget, los widgets compatibles aún pueden utilizarlo como fuente de datos para el período. Este parámetro no está disponible si *Aggregation function* está establecido en "not used".
Widget	Introduzca o seleccione un widget compatible como fuente de datos para el período. Este parámetro está disponible si *Time period* está establecido en "Widget".
From	Introduzca o seleccione el inicio del período. Se admite la sintaxis de tiempo relativo (`now`, `now/d`, `now/w-1w`, etc.). Este parámetro está disponible si *Time period* está establecido en "Custom".
To	Introduzca o seleccione el final del período. Se admite la sintaxis de tiempo relativo (`now`, `now/d`, `now/w-1w`, etc.). Este parámetro está disponible si *Time period* está establecido en "Custom".

Parámetros de **valor numérico del item**:

Display	Defina cómo se mostrará el valor: **As is** - como texto normal; **Bar** - como una barra sólida con relleno de color; **Indicators** - como una barra segmentada con relleno de color; **Sparkline** - gráfico de líneas pequeño.
Thresholds	Especifique los valores de umbral en los que debe cambiar el color de fondo o de relleno. La lista se ordenará en orden ascendente al guardarla. En los sparklines, los umbrales solo se aplican al último valor del item.
Decimal places	Especifique cuántas posiciones decimales se mostrarán con el valor.
Bar/Indicators	
Min	Valor mínimo.
Max	Valor máximo.
Sparkline	
Width	Establezca el grosor de la línea del gráfico mediante el control deslizante o introduciendo manualmente un valor entre 0 y 10.
Color	Seleccione el color de la línea y del relleno.
Fill	Establezca el nivel de transparencia del color de relleno mediante el control deslizante o introduciendo manualmente un valor entre 0 y 10.

Time period	Especifique el período cuyos valores se incluirán en el gráfico sparkline. Seleccione la fuente de datos para el período: **Dashboard** - utilizar el selector de período del dashboard **Widget** - utilizar un widget compatible (establecido en el parámetro *Widget*); **Custom** - utilizar un período personalizado establecido en los parámetros *From* y *To*; si se establece, se mostrará un icono de reloj en la esquina superior derecha del widget, que indicará el tiempo establecido al pasar el ratón por encima. Tenga en cuenta que, independientemente de la configuración de *Time period* del widget, los widgets compatibles aún pueden utilizarlo como fuente de datos para el período.
Widget	Introduzca o seleccione un widget compatible como fuente de datos para el período. Este parámetro está disponible si *Time period* está establecido en "Widget".
From	Introduzca o seleccione el inicio del período. Se admite la sintaxis de tiempo relativo (`now`, `now/d`, `now/w-1w`, etc.). Este parámetro está disponible si *Time period* está establecido en "Custom".
To	Introduzca o seleccione el final del período. Se admite la sintaxis de tiempo relativo (`now`, `now/d`, `now/w-1w`, etc.). Este parámetro está disponible si *Time period* está establecido en "Custom".
History data	Tomar los datos del historial o de las tendencias: **Auto** - selección automática; **History** - tomar los datos del historial; **Trends** - tomar los datos de las tendencias.

Parámetros de **valor de texto del item**:

Highlights	Especifique las expresiones regulares cuya coincidencia debe provocar el cambio del color de fondo o de relleno.

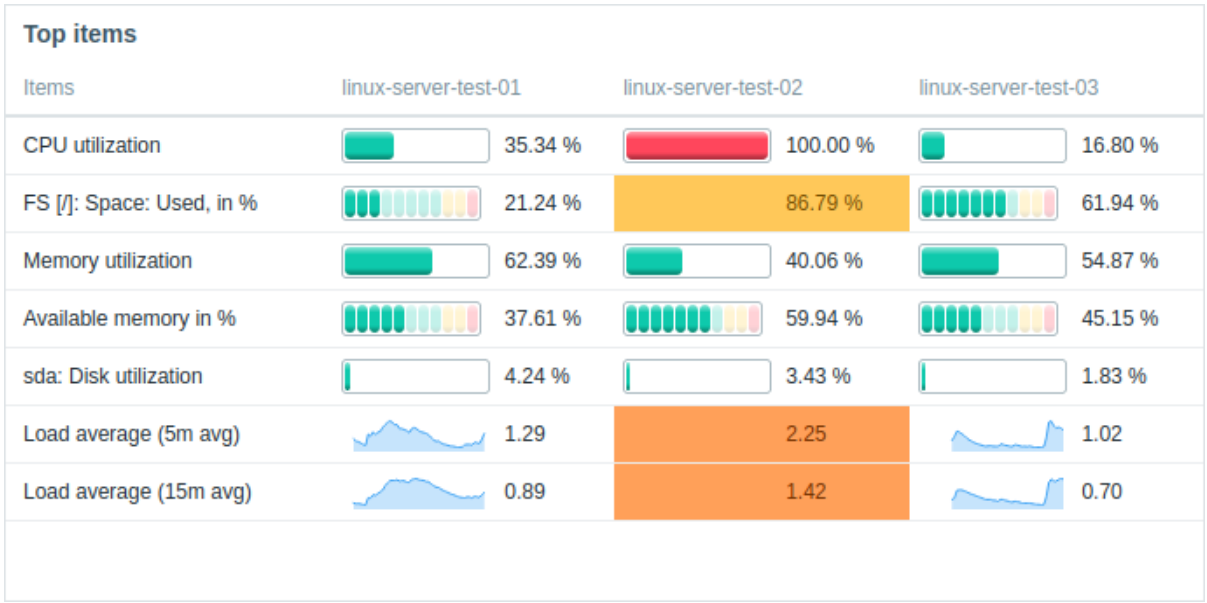
Parámetros de **valor binario del item**:

Show thumbnail	Especifique si se debe crear y mostrar una miniatura de la imagen que contiene datos binarios o mostrar un hipervínculo *Show* que dirija a la imagen a tamaño completo en la columna de valor.

Principales items

Descripción general

El widget *Elementos principales* muestra los valores más altos/bajos para los elementos seleccionados, proporcionando una visión rápida de su rendimiento.

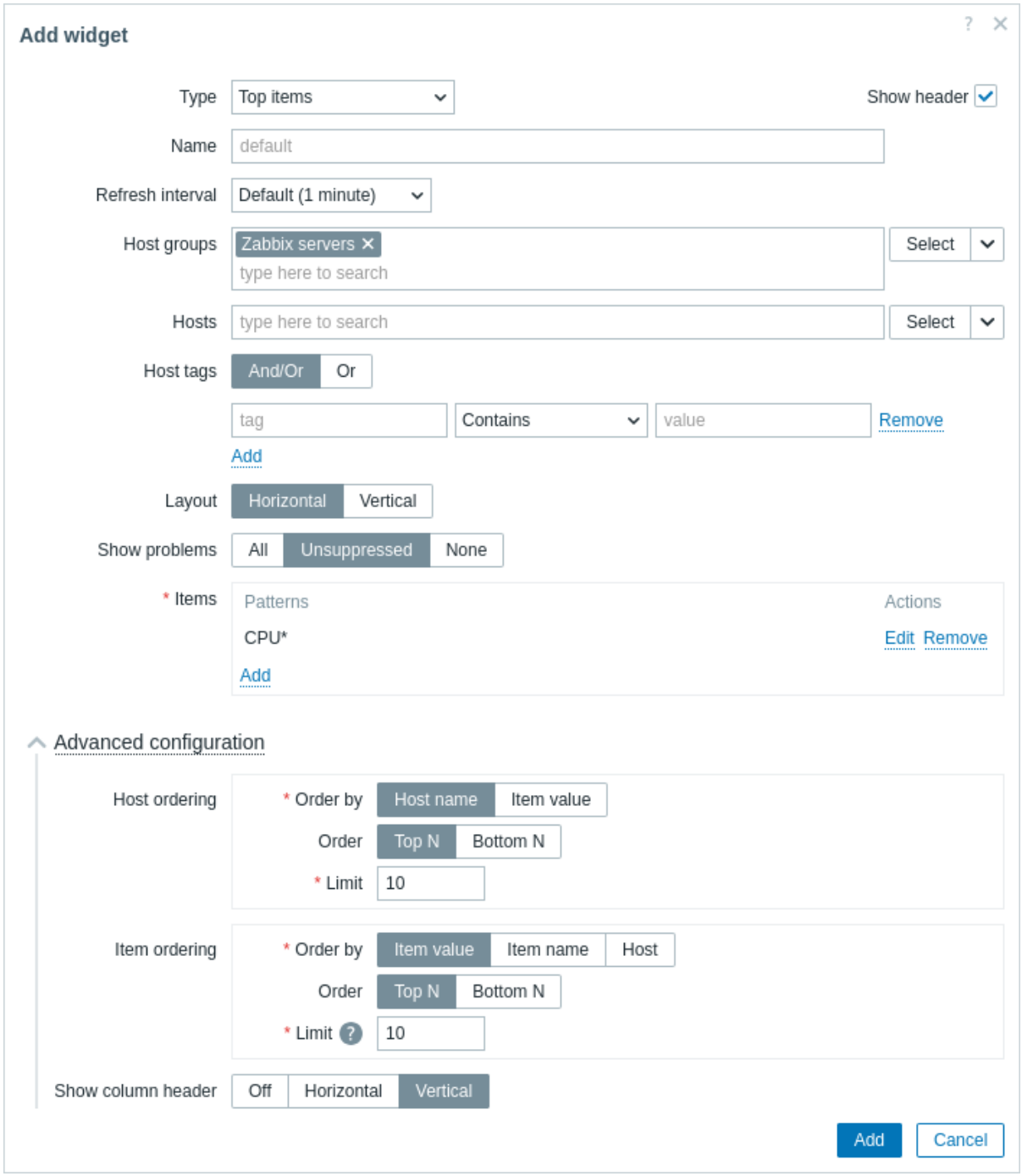


Por defecto, el widget muestra los valores de las últimas 24 horas. Este período puede cambiarse en *Administración* > *General* > *Interfaz gráfica* (*Período máximo de visualización de historial*).

Al hacer clic en el valor del elemento se abre el menú del elemento.

Configuración

Para configurar, seleccione *Top items* como tipo:



Además de los parámetros comunes para todos los widgets, puede definir las siguientes opciones específicas:

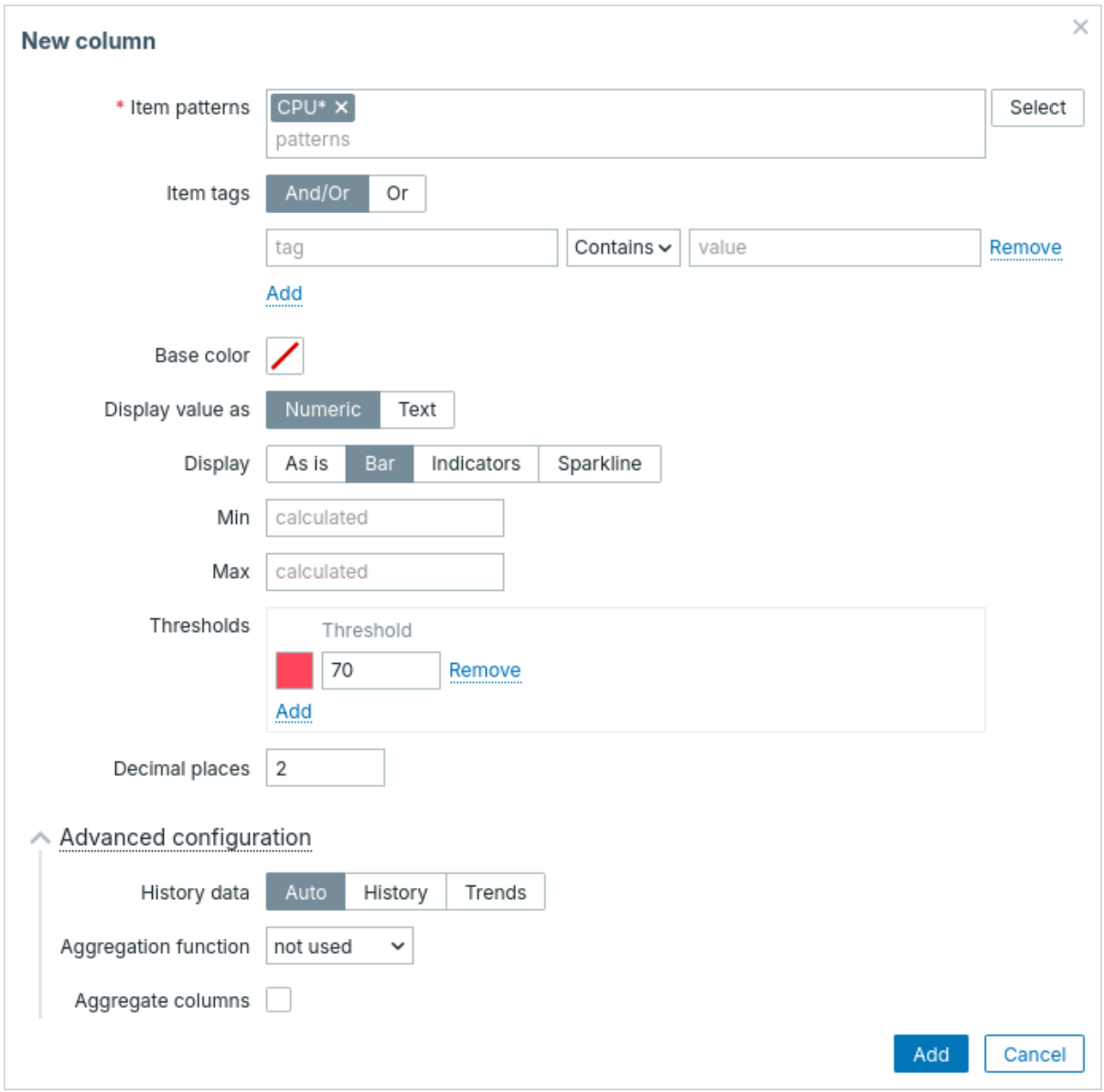
Host groups	Seleccione grupos de host. Como alternativa, seleccione un widget compatible como fuente de datos para los grupos de host. Este campo tiene autocompletado, por lo que al empezar a escribir el nombre de un grupo se mostrará una lista desplegable con los grupos que coincidan. Este parámetro no está disponible al configurar el widget en un template dashboard.

Hosts	Seleccione hosts. Como alternativa, seleccione un widget compatible o el dashboard como fuente de datos para los hosts. Este campo tiene autocompletado, por lo que al empezar a escribir el nombre de un host se mostrará una lista desplegable con los hosts que coincidan. Este parámetro no está disponible al configurar el widget en un template dashboard.
Host tags	Especifique etiquetas para limitar la cantidad de datos de host mostrados en el widget. Es posible incluir y excluir etiquetas y valores de etiqueta específicos. Se pueden establecer varias condiciones. La coincidencia del nombre de la etiqueta siempre distingue entre mayúsculas y minúsculas. Hay varios operadores disponibles para cada condición: **Exists** - incluir los nombres de etiqueta especificados; **Equals** - incluir los nombres y valores de etiqueta especificados (distingue entre mayúsculas y minúsculas); **Contains** - incluir los nombres de etiqueta especificados cuyos valores de etiqueta contengan la cadena introducida (coincidencia de subcadena, sin distinción entre mayúsculas y minúsculas); **Does not exist** - excluir los nombres de etiqueta especificados; **Does not equal** - excluir los nombres y valores de etiqueta especificados (distingue entre mayúsculas y minúsculas); **Does not contain** - excluir los nombres de etiqueta especificados cuyos valores de etiqueta contengan la cadena introducida (coincidencia de subcadena, sin distinción entre mayúsculas y minúsculas). Hay dos tipos de cálculo para las condiciones: **And/Or** - todas las condiciones deben cumplirse; las condiciones que tengan el mismo nombre de etiqueta se agruparán mediante la condición *Or*; **Or** - basta con que se cumpla una condición.
Layout	Seleccione la opción de diseño: **Horizontal** - los nombres de host se mostrarán a la izquierda; **Vertical** - los nombres de host se mostrarán en la parte superior.
Show problems	Filtre qué problemas mostrar según su estado (todos, no suprimidos, ninguno). Los problemas se indican con valores de item coloreados, anulando la configuración de visualización de la columna item. Los colores se basan en la severidad del problema, que se puede personalizar en *Administration* > *General* > *Trigger displaying options*. Las severidades de los problemas se pueden cambiar al actualizar problemas.
Items	Añada patrones de item o items específicos para su visualización (consulte Column configuration).
Advanced configuration	
Host ordering	Seleccione las opciones de ordenación para la columna/fila de host.
Order by	Ordene los hosts por: **Host name** - los hosts se ordenarán por nombre de host; **Item value** - los hosts se ordenarán por el valor de los items seleccionados.
Order	Seleccione si desea mostrar los valores más altos o más bajos: **TopN** - los N valores superiores; **BottomN** - los N valores inferiores. El valor de N se selecciona en el campo *Limit*.
Limit	Introduzca el límite de hosts que se pueden mostrar (rango 1-1000; valor predeterminado 10). Este valor se convertirá en el valor de N en el campo *Order*.

Item ordering		Seleccione las opciones de ordenación para la columna/fila de item.
	Order by	Ordene los items por: **Item value** - los items se ordenarán por el valor del item; **Item name** - los items se ordenarán por el nombre del item; **Host** - los items se ordenarán por el patrón de host seleccionado.
	Order	Seleccione si desea mostrar los valores más altos o más bajos: **TopN** - los N valores superiores; **BottomN** - los N valores inferiores. El valor de N se selecciona en el campo *Limit*.
	Limit	Introduzca el límite de items que se pueden mostrar (rango 1-1000; valor predeterminado 10). Este valor se convertirá en el valor de N en el campo *Order*.
Show column header		Seleccione las opciones de visualización del encabezado de columna: **Off** - no mostrar el encabezado de columna; **Horizontal** - mostrar el texto horizontalmente en el encabezado; **Vertical** - mostrar el texto verticalmente en el encabezado.

Configuración de columnas

Para configurar las columnas de métricas, haga clic en *Agregar* en el parámetro *Métricas*:



Parámetros comunes de columna:

Patrones de métricas	Especifique uno o varios patrones de métricas, utilizando el carácter comodín. Alternativamente, seleccione las métricas. Al configurar el widget en un dashboard de plantilla, solo se pueden seleccionar métricas configuradas en la plantilla.
Etiquetas de métricas	Especifique etiquetas para limitar la cantidad de datos de métricas mostrados en el widget. Es posible incluir así como excluir etiquetas y valores de etiquetas específicos. Se pueden establecer varias condiciones. La coincidencia de nombres de etiquetas siempre distingue entre mayúsculas y minúsculas. Hay varios operadores disponibles para cada condición: **Existe** - incluye los nombres de etiquetas especificados; **Igual** - incluye los nombres y valores de etiquetas especificados (distingue mayúsculas y minúsculas); **Contiene** - incluye los nombres de etiquetas especificados donde los valores de las etiquetas contienen la cadena introducida (coincidencia de subcadena, no distingue mayúsculas y minúsculas); **No existe** - excluye los nombres de etiquetas especificados; **No es igual** - excluye los nombres y valores de etiquetas especificados (distingue mayúsculas y minúsculas); **No contiene** - excluye los nombres de etiquetas especificados donde los valores de las etiquetas contienen la cadena introducida (coincidencia de subcadena, no distingue mayúsculas y minúsculas). Hay dos tipos de cálculo para las condiciones: **Y/O** - se deben cumplir todas las condiciones, las condiciones con el mismo nombre de etiqueta se agruparán por la condición *O*; **O** - basta con que se cumpla una condición.
Color base	Seleccione el color de fondo de la columna o el color de relleno si *Mostrar* está configurado en ”Barra” o ”Indicadores”. Tenga en cuenta que el color base puede ser sobrescrito por colores de umbral o de resaltado.
Mostrar valor como	Formato para mostrar el valor de la métrica: **Numérico** o **Texto**. La opción seleccionada determina qué parámetros adicionales estarán disponibles. Consulte la lista de parámetros para cada formato a continuación.

Parámetros de columna para valores numéricos:

Display	Defina cómo se debe mostrar el valor: **As is**: como texto normal; **Bar**: como una barra sólida rellena de color; **Indicators**: como una barra segmentada rellena de color; **Sparkline**: como un minigráfico de líneas.
Min	Valor mínimo para la visualización de barras/indicadores.
Max	Valor máximo para la visualización de barras/indicadores.
Width	Establezca el grosor de la línea del gráfico mediante el control deslizante o introduciendo manualmente un valor en el rango de 0 a 10. Este parámetro solo se aplica a la visualización de minigráficos.
Fill	Establezca el nivel de transparencia del color de relleno mediante el control deslizante o introduciendo manualmente un valor en el rango de 0 a 10. Este parámetro solo se aplica a la visualización de minigráficos.
Color	Seleccione el color de la línea y del relleno. Este parámetro solo se aplica a la visualización de minigráficos.
Consulte *Advanced configuration* para obtener una descripción de los campos relacionados con el período de tiempo y la selección de datos históricos.	

Thresholds	Especifique los valores de umbral a partir de los cuales debe cambiar el color de fondo/relleno. La lista se ordenará en orden ascendente al guardarla. En los minigráficos, los umbrales solo se aplican al último valor del item.
Decimal places	Especifique cuántas posiciones decimales se mostrarán con el valor.
Advanced configuration	
History data	Tome los datos del historial o de las tendencias: **Auto**: selección automática; **History**: tomar los datos del historial; **Trends**: tomar los datos de las tendencias. Este parámetro solo está disponible para valores numéricos de item. Para valores de texto, los datos siempre se tomarán del historial.
Aggregation function	Especifique qué función de agregación se utilizará dentro del *Time period* seleccionado: **min**: mostrar el valor más pequeño; **max**: mostrar el valor más grande; **avg**: mostrar el valor promedio; **count**: mostrar el número de valores; **sum**: mostrar la suma de los valores; **first**: mostrar el primer valor; **last**: mostrar el último valor. Si se selecciona **not used**, se mostrará el valor más reciente (sin agregación). La agregación permite mostrar un valor agregado para el intervalo especificado (5 minutos, una hora, un día), en lugar del valor más reciente. Solo se pueden mostrar datos numéricos para *min*, *max*, *avg* y *sum*. Para *count*, los datos no numéricos se convertirán a valores numéricos.
Time period	Especifique el período de tiempo que se utilizará para agregar los valores. Seleccione la fuente de datos para el período de tiempo: **Dashboard**: utilizar el selector de período de tiempo del dashboard; **Widget**: utilizar un widget compatible (establecido en el parámetro *Widget*); **Custom**: utilizar un período de tiempo personalizado establecido en los parámetros *From* y *To*; si se establece, se mostrará un icono de reloj en la esquina superior derecha del widget, que indicará la hora establecida al pasar el ratón por encima. Tenga en cuenta que, independientemente de la configuración de *Time period* del widget, los widgets compatibles aún pueden utilizarlo como fuente de datos para el período de tiempo. Este parámetro no está disponible si *Aggregation function* está establecido en "not used".
Widget	Introduzca o seleccione un widget compatible como fuente de datos para el período de tiempo. Este parámetro está disponible si *Time period* está establecido en "Widget".
From	Introduzca o seleccione el inicio del período de tiempo. Se admite la sintaxis de tiempo relativo (`now`, `now/d`, `now/w-1w`, etc.). Este parámetro está disponible si *Time period* está establecido en "Custom".
To	Introduzca o seleccione el final del período de tiempo. Se admite la sintaxis de tiempo relativo (`now`, `now/d`, `now/w-1w`, etc.). Este parámetro está disponible si *Time period* está establecido en "Custom".
Aggregate columns	Al seleccionarlo, agrupa todos los items que coincidan con los patrones de item de la columna en una única columna/fila combinada. Los valores de los items coincidentes se agregarán mediante la *Column aggregation function* seleccionada y se mostrarán bajo *Combined column name*. Valor predeterminado: desactivado (sin seleccionar).Este parámetro solo se muestra cuando *Display value as* es **Numeric**.
Column aggregation function	Elija cómo se combinarán los valores de los patrones de item agrupados. Opciones disponibles: **min**, **max**, **avg**, **count**, **sum** (predeterminado). Este parámetro solo se muestra cuando *Aggregate columns* está seleccionado.
Combined column name	Introduzca el nombre que se mostrará para la columna agregada. Este campo es obligatorio cuando *Aggregate columns* está seleccionado y se convierte en el encabezado de la columna para los valores combinados.

Parámetros de columna para valores de texto:

Highlights	Especifique las expresiones regulares cuya coincidencia debe provocar el cambio del color de fondo/relleno.

Principales triggers

Descripción general

El widget *Principales disparadores* muestra los disparadores con el mayor número de problemas.

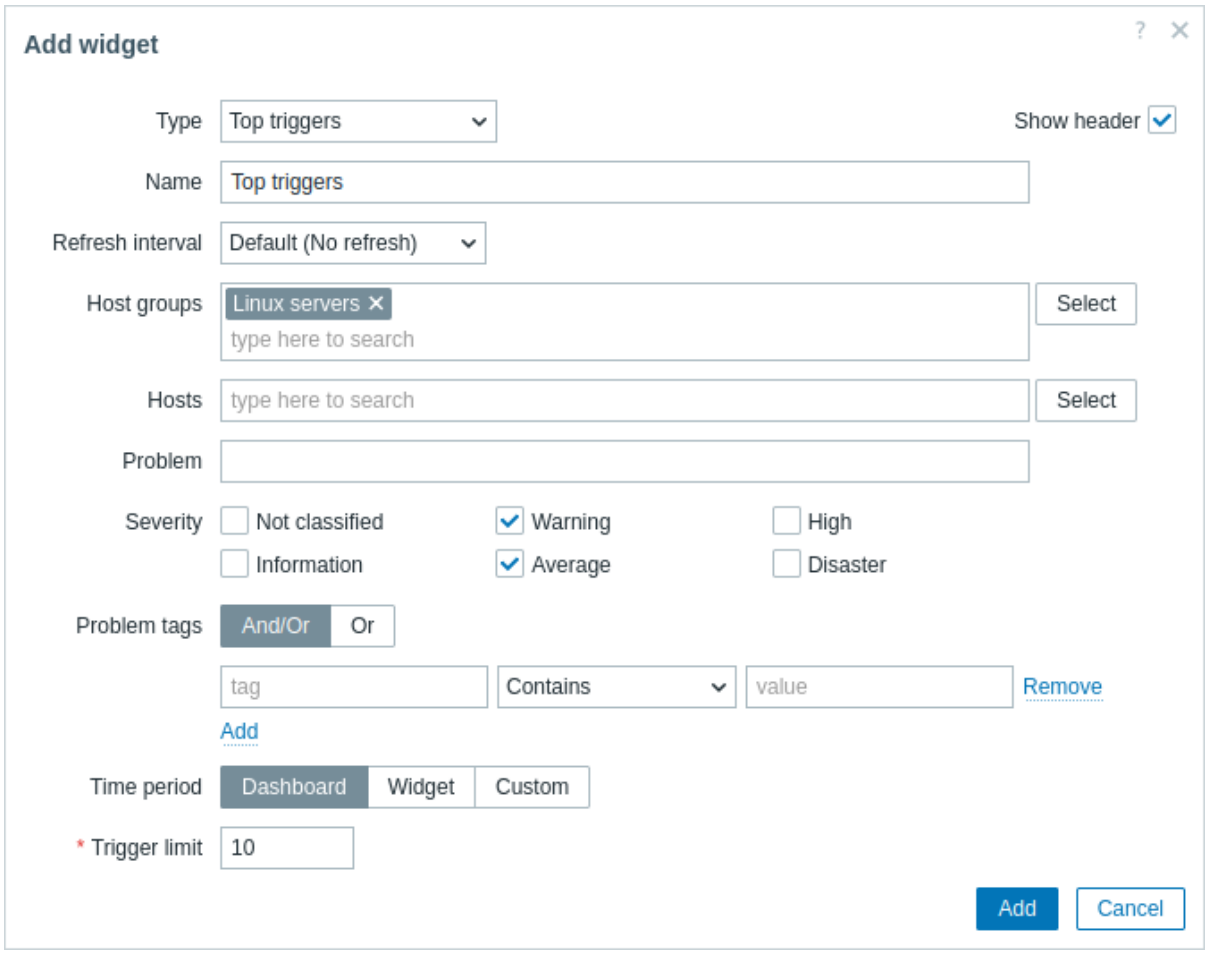
Top triggers

Host	Trigger	Severity	Number of problems
Zabbix server	Interface enp0s3: Link down	Average	2
Zabbix server	Load average is too high	Average	2
Zabbix server	Zabbix agent is not available	Average	2
Zabbix server	Zabbix server: More than 100 items having missing data for more than 10 minutes	Warning	2
Zabbix server	Zabbix server: Utilization of escalator processes is high	Average	2

Muestra los mismos datos que *Informes > Principales 100 disparadores* y puede mostrar hasta 1000 registros.

Configuración

Para configurar, seleccione *Principales disparadores* como tipo de widget:



Además de los parámetros que son comunes para todos los widgets, puede establecer las siguientes opciones específicas:

Grupos de equipos	Seleccione los grupos de equipos cuyos problemas se mostrarán en el widget. Este campo es autocompletable, por lo que al comenzar a escribir el nombre de un grupo se ofrecerá un menú desplegable con los grupos coincidentes. Al especificar un grupo de equipos principal, se seleccionan implícitamente todos los grupos de equipos anidados. Los problemas de estos grupos de equipos se mostrarán en el widget; si no se ingresan grupos de equipos, se mostrarán los problemas de todos los grupos de equipos. Este parámetro no está disponible al configurar el widget en un panel de plantilla.
Equipos	Seleccione los equipos cuyos problemas se mostrarán en el widget. Este campo es autocompletable, por lo que al comenzar a escribir el nombre de un equipo se ofrecerá un menú desplegable con los equipos coincidentes. Si no se ingresan equipos, se mostrarán los problemas de todos los equipos. Este parámetro no está disponible al configurar el widget en un panel de plantilla.
Problema	Puede ver los disparadores solo para problemas concretos. Para ello, introduzca la cadena que debe coincidir en el nombre del problema. Las macros no se expanden.
Gravedad	Marque las gravedades de los disparadores para filtrar los disparadores que se mostrarán en el widget. Si no se marca ninguna gravedad, se mostrarán todos los disparadores.
Etiquetas de problema	Especifique las etiquetas de los problemas que se mostrarán en el widget. Es posible incluir o excluir etiquetas y valores de etiquetas específicos. Se pueden establecer varias condiciones. La coincidencia de nombres de etiquetas siempre distingue entre mayúsculas y minúsculas. Hay varios operadores disponibles para cada condición: **Existe**: incluye los nombres de etiquetas especificados; **Igual**: incluye los nombres y valores de etiquetas especificados (distingue mayúsculas y minúsculas); **Contiene**: incluye los nombres de etiquetas especificados donde los valores de las etiquetas contienen la cadena introducida (coincidencia de subcadena, no distingue mayúsculas y minúsculas); **No existe**: excluye los nombres de etiquetas especificados; **No igual**: excluye los nombres y valores de etiquetas especificados (distingue mayúsculas y minúsculas); **No contiene**: excluye los nombres de etiquetas especificados donde los valores de las etiquetas contienen la cadena introducida (coincidencia de subcadena, no distingue mayúsculas y minúsculas). Hay dos tipos de cálculo para las condiciones: **Y/O**: se deben cumplir todas las condiciones, las condiciones con el mismo nombre de etiqueta se agruparán por la condición *O*; **O**: basta con que se cumpla una condición.
Período de tiempo	Seleccione la fuente de datos para el período de tiempo: **Panel**: utiliza el selector de período de tiempo del panel; **Widget**: utiliza un widget compatible (establecido en el parámetro *Widget*); **Personalizado**: utiliza un período de tiempo personalizado establecido en los parámetros *Desde* y *Hasta*; si se establece, se mostrará un icono de reloj en la esquina superior derecha del widget, indicando el tiempo establecido al pasar el ratón. Tenga en cuenta que, independientemente de la configuración de *Período de tiempo* del widget, los widgets compatibles aún pueden usarlo como fuente de datos para el período de tiempo.
Widget	Introduzca o seleccione un widget compatible como fuente de datos para el período de tiempo. Este parámetro está disponible si *Período de tiempo* está establecido en ”Widget”.
Desde	Introduzca o seleccione el inicio del período de tiempo. Se admite la sintaxis de tiempo relativo (`now`, `now/d`, `now/w-1w`, etc.). Este parámetro está disponible si *Período de tiempo* está establecido en ”Personalizado”.
Hasta	Introduzca o seleccione el final del período de tiempo. Se admite la sintaxis de tiempo relativo (`now`, `now/d`, `now/w-1w`, etc.). Este parámetro está disponible si *Período de tiempo* está establecido en ”Personalizado”.
Límite de disparadores	Establezca el número de disparadores que se mostrarán. Rango de valores posible: 1-1000.

Resumen de trigger

Overview

El widget *Trigger overview* muestra los estados actuales de los triggers de los host como una tabla de bloques de colores. Proporciona una vista visual rápida del estado y la actividad de los triggers en distintos host.



El widget utiliza varios elementos visuales para mostrar los detalles de los triggers:

- Los colores de los bloques para los triggers en PROBLEM coinciden con las severidades de los problemas.
- Los bloques parpadeantes indican cambios recientes en el estado del trigger (dentro de los últimos 2 minutos).
- Las flechas grises indican triggers con dependencias (pase el cursor sobre ellas para ver los detalles).
- Los iconos de casilla de verificación indican problemas reconocidos; para que aparezca este icono, todos los problemas activos o resueltos del trigger deben estar reconocidos.

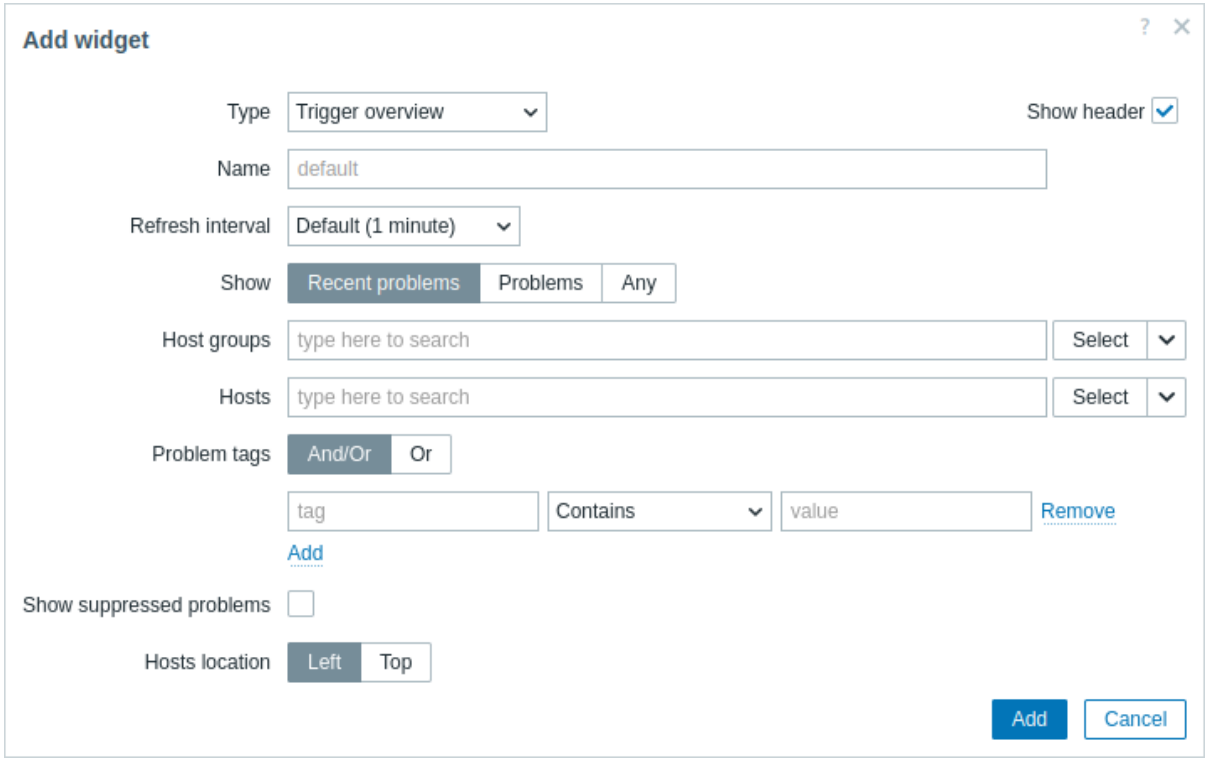
Note:
Los colores de severidad de los triggers se pueden ajustar en *Administration* > *General*. Las severidades de los triggers se pueden modificar al actualizar problemas.

Al hacer clic en un bloque, se muestran enlaces a información relacionada con el trigger (problemas, configuración del trigger, etc.).

De forma predeterminada, se muestran hasta 50 registros (configurable en *Administration* > *General* > *GUI*, en *Max number of columns and rows in overview tables*). Si existen más triggers, se mostrará un mensaje para que refine los criterios de filtrado. Este límite se aplica antes de cualquier filtrado adicional (por ejemplo, por etiquetas).

Configuración

Para configurar, seleccione *Resumen de disparadores* como tipo:



Además de los parámetros que son comunes para todos los widgets, puede establecer las siguientes opciones específicas:

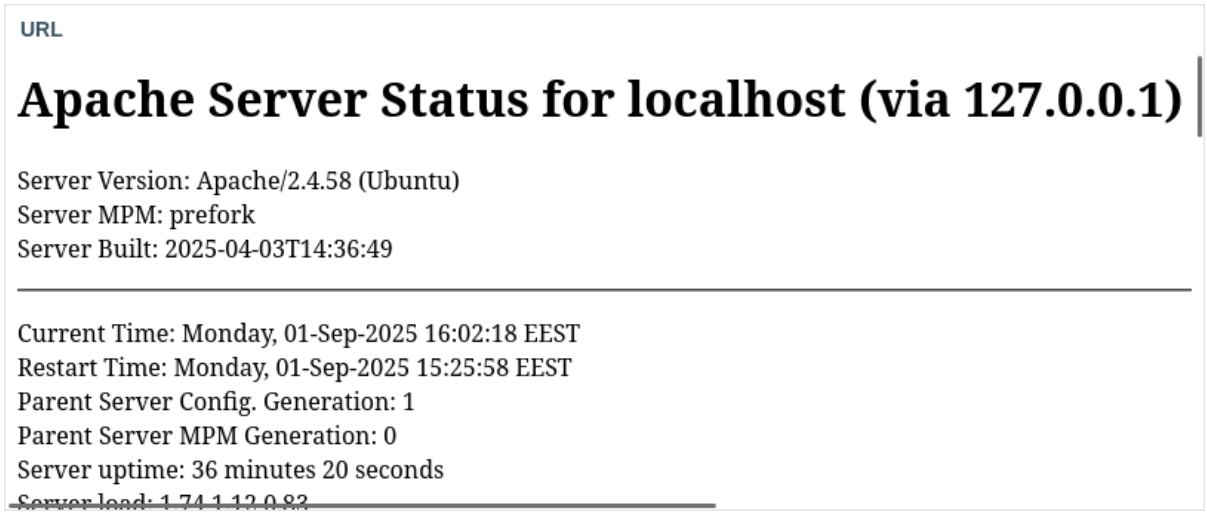
Mostrar	Filtrar disparadores por estado: **Problemas recientes** - *(por defecto)* muestra los disparadores que recientemente han estado o aún están en estado de PROBLEMA (resueltos y no resueltos); **Problemas** - muestra los disparadores que están en estado de PROBLEMA (no resueltos); **Cualquiera** - muestra todos los disparadores.
Grupos de equipos	Seleccione grupos de equipos. Alternativamente, seleccione un widget compatible como fuente de datos para los grupos de equipos. Este campo es autocompletado, por lo que al comenzar a escribir el nombre de un grupo se ofrecerá un desplegable con los grupos coincidentes. Este parámetro no está disponible al configurar el widget en un dashboard de plantilla.
Equipos	Seleccione equipos. Alternativamente, seleccione un widget compatible o el dashboard como fuente de datos para los equipos. Este campo es autocompletado, por lo que al comenzar a escribir el nombre de un equipo se ofrecerá un desplegable con los equipos coincidentes. Este parámetro no está disponible al configurar el widget en un dashboard de plantilla.

Etiquetas de problema	Especifique etiquetas para filtrar los disparadores mostrados en el widget. Es posible incluir así como excluir etiquetas y valores de etiqueta específicos. Se pueden establecer varias condiciones. La coincidencia de nombres de etiqueta siempre distingue entre mayúsculas y minúsculas. **Nota:** Si el parámetro *Mostrar* está configurado en 'Cualquiera', se mostrarán todos los disparadores incluso si se especifican etiquetas. Sin embargo, mientras que los cambios recientes de estado del disparador (mostrados como bloques parpadeantes) se actualizarán para todos los disparadores, los detalles del estado del disparador (color de severidad del problema y si el problema está reconocido) solo se actualizarán para los disparadores que coincidan con las etiquetas especificadas. Hay varios operadores disponibles para cada condición: **Existe** - incluye los nombres de etiqueta especificados; **Igual** - incluye los nombres y valores de etiqueta especificados (distingue mayúsculas y minúsculas); **Contiene** - incluye los nombres de etiqueta especificados donde los valores de etiqueta contienen la cadena introducida (coincidencia de subcadena, no distingue mayúsculas y minúsculas); **No existe** - excluye los nombres de etiqueta especificados; **No es igual** - excluye los nombres y valores de etiqueta especificados (distingue mayúsculas y minúsculas); **No contiene** - excluye los nombres de etiqueta especificados donde los valores de etiqueta contienen la cadena introducida (coincidencia de subcadena, no distingue mayúsculas y minúsculas). Hay dos tipos de cálculo para las condiciones: **Y/O** - se deben cumplir todas las condiciones, las condiciones con el mismo nombre de etiqueta se agruparán por la condición *O*; **O** - basta con que se cumpla una condición.
Mostrar problemas suprimidos	Marque la casilla para mostrar los problemas que de otro modo serían suprimidos (no mostrados) debido al mantenimiento del equipo.
Disposición	Seleccione la opción de disposición: **Horizontal** - los nombres de los equipos se mostrarán a la izquierda; **Vertical** - los nombres de los equipos se mostrarán en la parte superior.

URL

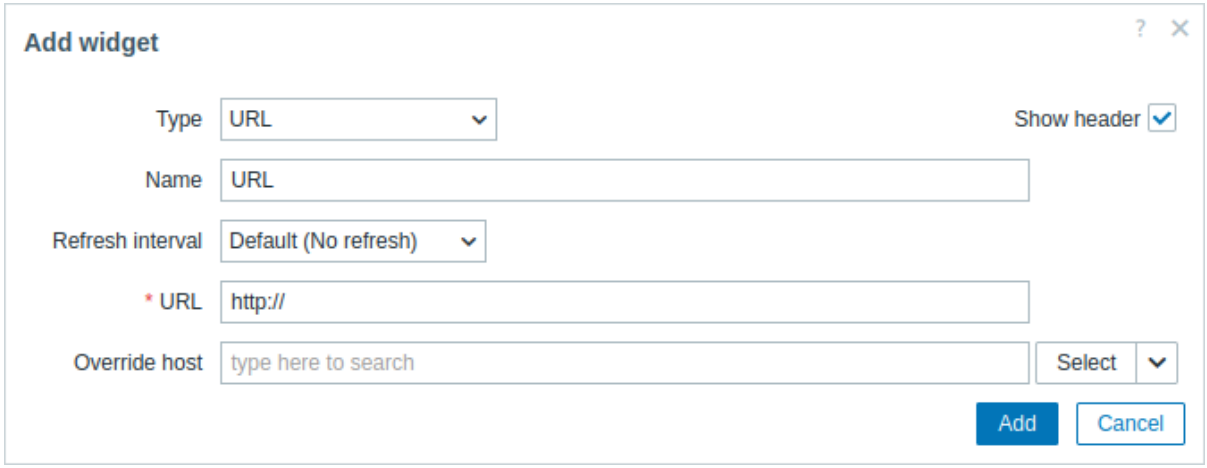
Descripción general

El widget *URL* muestra el contenido recuperado de una URL.



Configuración

Para configurar, seleccione *URL* como tipo:



Además de los parámetros comunes para todos los widgets, puede definir las siguientes opciones específicas:

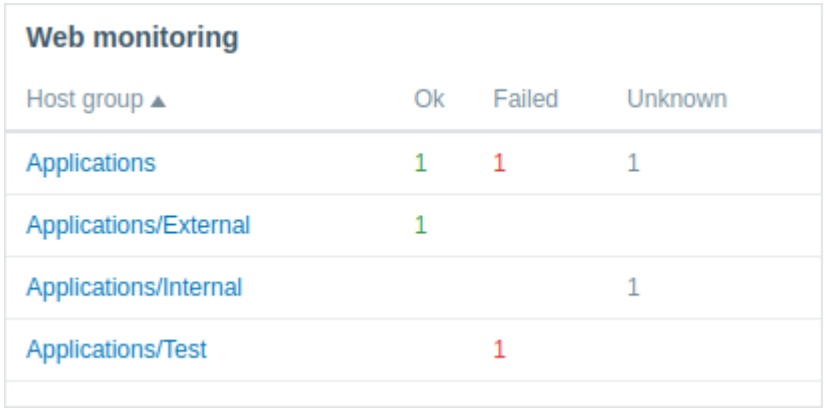
URL	Introduzca la URL que se mostrará (hasta 2048 caracteres). Las URL externas deben comenzar con `http://` o `https://`. Las URL internas admiten rutas relativas (por ejemplo, `zabbix.php?action=report.status`). Se admiten macros `{HOST.*}`.
Override host	Seleccione un widget compatible o el selector de host del dashboard como origen de datos para los hosts. Este parámetro no está disponible al configurar el widget en un template dashboard.

Attention:
Es posible que los navegadores no carguen una página HTTP configurada en el widget si se accede al frontend de Zabbix mediante HTTPS. Además, puede que necesite modificar el encabezado HTTP de respuesta X-Frame-Options para mostrar contenido externo al frontend de Zabbix.

Supervisión web

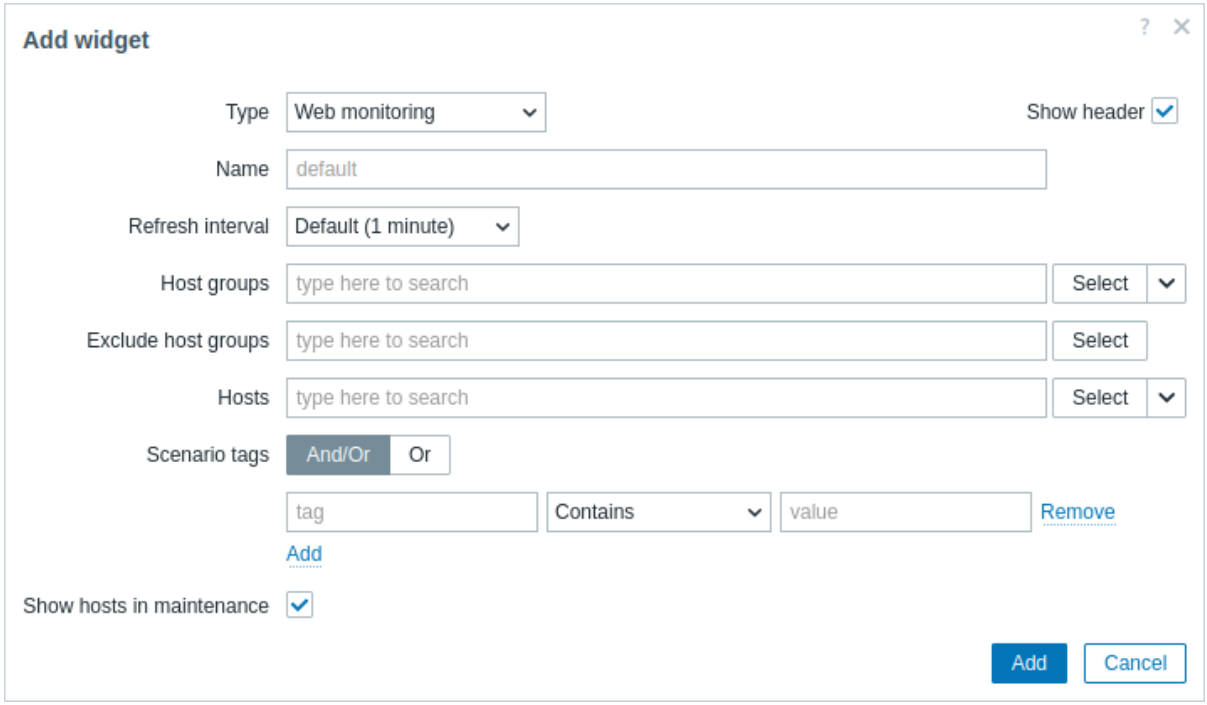
Descripción general

El widget de *Supervisión web* muestra el estado de los escenarios web activos.



Al hacer clic en el nombre de un grupo de equipos, se abre *Supervisión* > *Equipos* > *Escenarios web*, donde puede ver los detalles del escenario.

Configuración



Además de los parámetros que son comunes para todos los widgets, puede establecer las siguientes opciones específicas:

Grupos de equipos	Seleccione los grupos de equipos que se mostrarán en el widget. Alternativamente, seleccione un widget compatible como fuente de datos para los grupos de equipos. Este campo es autocompletable, por lo que al comenzar a escribir el nombre de un grupo se mostrará un desplegable con los grupos coincidentes. Especificar un grupo de equipos principal selecciona implícitamente todos los grupos de equipos anidados. Los datos de los equipos de estos grupos se mostrarán en el widget; si no se introducen grupos de equipos, se mostrarán todos los grupos. Este parámetro no está disponible al configurar el widget en un panel de plantilla.
Excluir grupos de equipos	Seleccione los grupos de equipos que se ocultarán en el widget. Este campo es autocompletable, por lo que al comenzar a escribir el nombre de un grupo se mostrará un desplegable con los grupos coincidentes. Especificar un grupo de equipos principal selecciona implícitamente todos los grupos de equipos anidados. Los datos de los equipos de estos grupos no se mostrarán en el widget. Por ejemplo, los equipos 001, 002, 003 pueden estar en el Grupo A y los equipos 002, 003 también en el Grupo B. Si seleccionamos *mostrar* el Grupo A y *excluir* el Grupo B al mismo tiempo, solo se mostrarán los datos del equipo 001 en el panel. Este parámetro no está disponible al configurar el widget en un panel de plantilla.
Equipos	Seleccione los equipos que se mostrarán en el widget. Alternativamente, seleccione un widget compatible o el panel como fuente de datos para los equipos. Este campo es autocompletable, por lo que al comenzar a escribir el nombre de un equipo se mostrará un desplegable con los equipos coincidentes. Si no se introducen equipos, se mostrarán todos los equipos. Este parámetro no está disponible al configurar el widget en un panel de plantilla.

Etiquetas de escenarios	Especifique etiquetas para limitar el número de escenarios web mostrados en el widget. Es posible incluir o excluir etiquetas y valores de etiquetas específicos. Se pueden establecer varias condiciones. La coincidencia de nombres de etiquetas siempre distingue entre mayúsculas y minúsculas. Hay varios operadores disponibles para cada condición: **Existe**: incluye los nombres de etiquetas especificados; **Igual**: incluye los nombres y valores de etiquetas especificados (distingue mayúsculas y minúsculas); **Contiene**: incluye los nombres de etiquetas especificados donde los valores de las etiquetas contienen la cadena introducida (coincidencia de subcadena, no distingue mayúsculas y minúsculas); **No existe**: excluye los nombres de etiquetas especificados; **No es igual**: excluye los nombres y valores de etiquetas especificados (distingue mayúsculas y minúsculas); **No contiene**: excluye los nombres de etiquetas especificados donde los valores de las etiquetas contienen la cadena introducida (coincidencia de subcadena, no distingue mayúsculas y minúsculas). Hay dos tipos de cálculo para las condiciones: **Y/O**: se deben cumplir todas las condiciones, las condiciones con el mismo nombre de etiqueta se agruparán por la condición *O*; **O**: basta con que se cumpla una condición.
Mostrar equipos en mantenimiento	Incluye los equipos que están en mantenimiento en las estadísticas. Este parámetro se etiqueta como *Mostrar datos en mantenimiento* al configurar el widget en un panel de plantilla.

Monitoreo

Descripción general

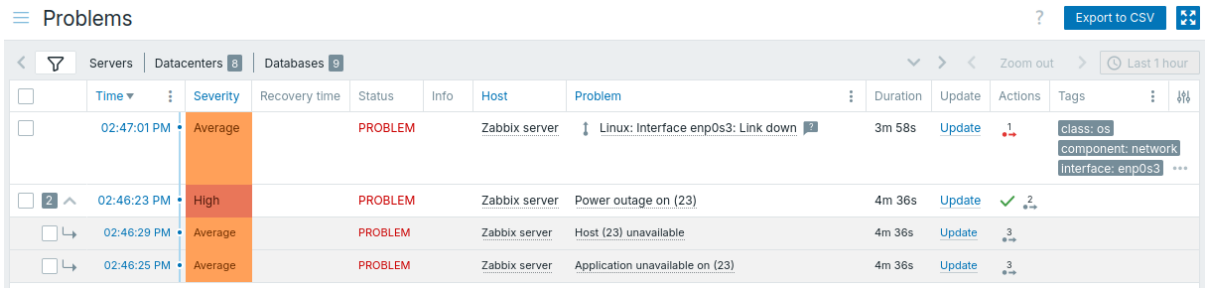
El menú Monitoring se centra en mostrar datos. Toda la información que Zabbix esté configurado para recopilar, visualizar y procesar se mostrará en las distintas secciones del menú *Monitoring*.

Problemas

Resumen

En *Monitoring* > *Problems* puedes ver qué problemas tienes actualmente. Los problemas son aquellos triggers que están en el estado "Problem".

De forma predeterminada, todos los problemas nuevos se clasifican como problemas de causa. Es posible reclasificar manualmente ciertos problemas como problemas de síntoma del problema de causa. Para más detalles, consulta eventos de causa y síntoma.

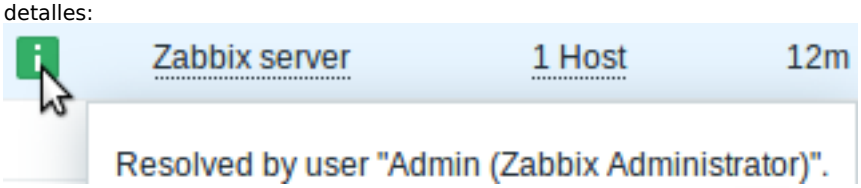
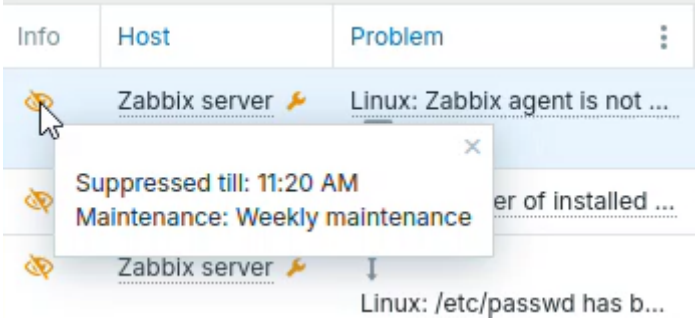


Los detalles del problema se muestran en una tabla personalizable. El botón *Customize table* en el encabezado de la tabla te permite mostrar, ocultar o reordenar columnas, y restablecer el diseño al valor predeterminado. También están disponibles estas opciones:

- *Compact view* - reduce la altura de las filas para mostrar más problemas en pantalla.
- *Highlight full row* - usa el color de severidad del problema para resaltar los problemas no resueltos (no disponible en temas de alto contraste).

También puedes cambiar el tamaño de cada columna arrastrando su borde derecho; haz doble clic en el borde para restablecerlo. Todas las personalizaciones se aplican de inmediato y siempre se recuerdan para el usuario actual dentro de la pestaña activa de filter (predeterminada o guardada).

Las siguientes columnas están disponibles:

Column	Description
Checkbox	Se muestran casillas de verificación para seleccionar problemas. Los iconos junto a las casillas tienen el siguiente significado:  - el número de eventos de síntoma para el problema de causa;  - expandir para mostrar eventos de síntoma;  - contraer para ocultar eventos de síntoma;  - este es un evento de síntoma.
Time	Se muestra la hora de inicio del problema. Al hacer clic en el botón *Options* del encabezado de la columna, puedes acceder a la opción *Show timeline*, que alterna la línea de tiempo visual y la agrupación de problemas (no disponible cuando *Compact view* está habilitado).
Severity	Se muestra la severidad del problema. La severidad del problema se basa originalmente en la severidad del trigger subyacente; sin embargo, después de que ocurre el evento puede actualizarse mediante la pantalla *Update problem* screen. El color de la severidad del problema se usa como fondo de la celda durante el tiempo del problema.
Recovery time	Se muestra la hora de resolución del problema. Esta columna no está disponible cuando *Show* en el filter está configurado en *Problems*.
Status	Se muestra el estado del problema: **Problem** - problema no resuelto **Resolved** - problema resuelto recientemente. Puedes ocultar los problemas resueltos recientemente usando el filtro. Los problemas nuevos y los resueltos recientemente parpadean durante 2 minutos. Los problemas resueltos se muestran durante 5 minutos en total. Ambos valores se pueden configurar en *Administration* > *General* > *Trigger displaying options*. Esta columna no está disponible cuando *Show* en el filter está configurado en *Problems*.
Info	Se muestra un icono verde de información si un problema se cierra por correlación global o manualmente al actualizar el problema. Al pasar el cursor sobre el icono se muestran más detalles:  Si se muestra un problema suprimido, se muestra el siguiente icono: . Cuando el evento está suprimido por mantenimiento del host, el icono se muestra en naranja: . Si se aplican tanto el mantenimiento como la supresión manual, la supresión manual tiene prioridad en el color del icono. Al pasar el cursor sobre el icono se muestran más detalles: 
Host	Se muestra el host del problema. Al hacer clic en el nombre del host se abre el host menu.

Column	Description
Problem	Se muestra el nombre del problema. El nombre del problema se basa en el nombre del trigger subyacente del problema. Las macros en el nombre del trigger se resuelven en el momento en que ocurre el problema y los valores resueltos ya no se actualizan. Al hacer clic en el nombre del problema se abre el event menu. Al pasar el cursor sobre el icono después del nombre del problema se mostrará la descripción del trigger (para aquellos problemas que la tengan). Al hacer clic en el botón *Options* del encabezado de la columna, puedes acceder a dos opciones: - *Show operational data* muestra los datos operativos después del nombre del problema entre paréntesis (solo si el campo *Operational data* no está vacío en la trigger configuration). - *Show trigger expression* muestra la expresión del trigger del problema debajo del nombre del problema (no disponible cuando *Compact view* está habilitado).
Operational data	Se muestran datos dinámicos en tiempo real para el problema. El contenido depende del campo *Operational data* en la trigger configuration: si se define una cadena con macros, se muestra aquí; si el campo se deja vacío, en su lugar se muestran los últimos valores de todos los items de la expresión del trigger.
Duration	Se muestra la duración del problema. Consulta también: Duración negativa del problema
Update	Haz clic en el enlace *Update* para ir a la pantalla de actualización del problema, donde se pueden realizar varias acciones sobre el problema, incluido comentar y reconocer el problema.
Actions	Se muestra el historial de actividades sobre el problema mediante iconos simbólicos: - el problema ha sido reconocido. Este icono siempre se muestra primero. - se han hecho comentarios. También se muestra el número de comentarios. - la severidad del problema ha aumentado (por ejemplo, Information > Warning). - la severidad del problema ha disminuido (por ejemplo, Warning > Information). - la severidad del problema ha cambiado, pero ha vuelto al nivel original (por ejemplo, Warning > Information > Warning). - el problema ha sido suprimido. - la supresión del problema se ha eliminado. - el problema ha sido suprimido debido al mantenimiento del host. - la supresión del problema se ha eliminado después del mantenimiento del host. - se han realizado acciones. También se muestra el número de acciones. - se han realizado acciones, al menos una está en curso. También se muestra el número de acciones. - se han realizado acciones, al menos una ha fallado. También se muestra el número de acciones. Al pasar el ratón sobre los iconos, se muestran ventanas emergentes con detalles sobre la actividad. Consulta ver detalles para obtener más información sobre los iconos usados en la ventana emergente para las acciones realizadas.
Tags	Se muestran los tags del problema. Además, también pueden mostrarse tags de un sistema externo de gestión de incidencias (consulta la opción *Process tags* al configurar webhooks). Al pasar el cursor o hacer clic en el icono de tres puntos después de los tags mostrados, se abre una ventana emergente con todos los tags. Al hacer clic en el botón *Options* del encabezado de la columna, puedes establecer el número de tags mostrados (1-3), elegir cómo se muestran los nombres de los tags (completos, abreviados o ninguno), establecer la prioridad de visualización de los tags o duplicar la columna (las columnas duplicadas se pueden renombrar o eliminar).

Column	Description
Tag value	Se muestran los valores de los tags del problema. Al hacer clic en el botón *Options* del encabezado de la columna, puedes cambiar el nombre de la columna, elegir qué valores de tag mostrar o duplicar la columna (las columnas duplicadas se pueden renombrar o eliminar).
Custom text	Se muestra texto personalizado, valores de macros o una combinación de ambos. Al hacer clic en el botón *Options* del encabezado de la columna, puedes cambiar el nombre de la columna, introducir el texto personalizado o las macros, o duplicar la columna (las columnas duplicadas se pueden renombrar o eliminar). Macros compatibles: {HOST.*} (excepto {HOST.METADATA}), {INVENTORY.*}, {ITEM.*} (excepto {ITEM.STATE.ERROR}), {TRIGGER.ID}, {EVENT.ID} y user macros.

Duración negativa del problema

En realidad, es posible en algunas situaciones comunes tener una duración negativa del problema, es decir, cuando la hora de resolución del problema es anterior a la hora de creación del problema, por ejemplo:

- Si algún host es monitorizado por un proxy y ocurre un error de red, lo que provoca que no se reciban datos del proxy durante un tiempo, el trigger nodata(/host/key) será activado por el server. Cuando se restablece la conexión, el server recibirá datos del item desde el proxy con una hora del pasado. Entonces, el problema nodata(/host/key) se resolverá y tendrá una duración negativa;
- Cuando los datos del item que resuelven el evento del problema son enviados por Zabbix sender y contienen una marca de tiempo anterior a la hora de creación del problema, también se mostrará una duración negativa del problema.

Note:
La duración negativa del problema no afecta de ninguna manera al cálculo de SLA ni al informe de disponibilidad de un trigger en particular; ni reduce ni amplía el tiempo del problema.

Opciones de edición masiva

Los botones debajo de la lista ofrecen algunas opciones de edición masiva:

- *Actualización masiva* - actualiza los problemas seleccionados navegando a la pantalla de actualización de problemas

Para usar esta opción, marque las casillas de verificación delante de los problemas correspondientes y, a continuación, haga clic en el botón *Actualización masiva*.

Botones

El botón de la derecha ofrece la siguiente opción:

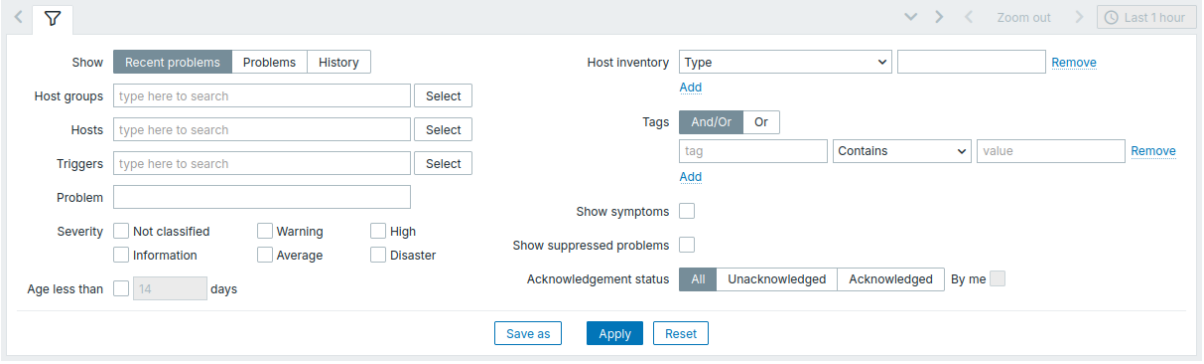
Export to CSV	Exportar el contenido de todas las páginas a un archivo CSV.

Los botones del modo de vista, al ser comunes para todas las secciones, se describen en la página Monitoring.

Uso del filtro

Puede usar el filtro para mostrar solo los problemas que le interesan. Para un mejor rendimiento de búsqueda, los datos se buscan con las macros sin resolver.

El filtro se encuentra encima de la tabla. Las configuraciones favoritas del filtro se pueden guardar como pestañas y luego acceder a ellas rápidamente haciendo clic en las pestañas encima del filtro.

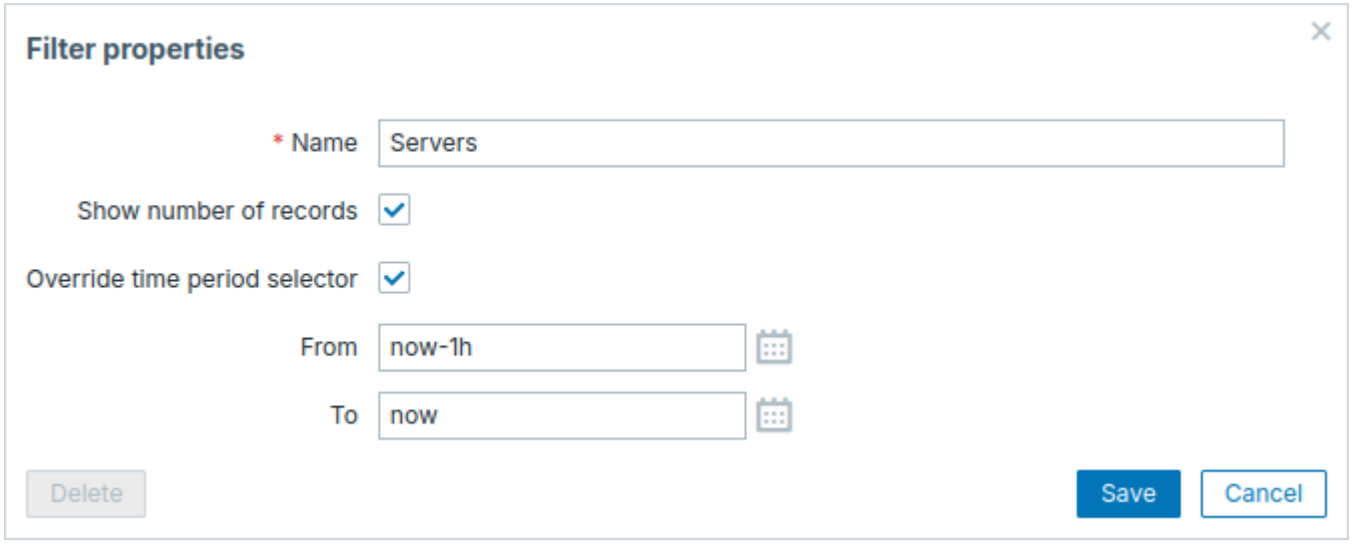


Parameter	Description
Show	Filtrar por estado del problema: **Recent problems** - se muestran los problemas no resueltos y los resueltos recientemente (predeterminado) **Problems** - se muestran los problemas no resueltos **History** - se muestra el historial de todos los eventos
Host groups	Filtrar por uno o varios grupos de host. Especificar un grupo de host padre selecciona implícitamente todos los grupos de host anidados.
Hosts	Filtrar por uno o varios host.
Triggers	Filtrar por uno o varios trigger.
Problem	Filtrar por nombre del problema.
Severity	Filtrar por la severidad del trigger (problema).
Age less than	Filtrar por la antigüedad del problema.
Host inventory	Filtrar por tipo y valor del inventario.
Tags	Filtrar por nombre y valor de event tag. Es posible incluir y excluir etiquetas y valores de etiqueta específicos. Se pueden establecer varias condiciones. La coincidencia del nombre de la etiqueta siempre distingue entre mayúsculas y minúsculas. Hay varios operadores disponibles para cada condición: **Exists** - incluir los nombres de etiqueta especificados **Equals** - incluir los nombres y valores de etiqueta especificados (distingue entre mayúsculas y minúsculas) **Contains** - incluir los nombres de etiqueta especificados cuando los valores de etiqueta contengan la cadena introducida (coincidencia de subcadena, no distingue entre mayúsculas y minúsculas) **Does not exist** - excluir los nombres de etiqueta especificados **Does not equal** - excluir los nombres y valores de etiqueta especificados (distingue entre mayúsculas y minúsculas) **Does not contain** - excluir los nombres de etiqueta especificados cuando los valores de etiqueta contengan la cadena introducida (coincidencia de subcadena, no distingue entre mayúsculas y minúsculas) Hay dos tipos de cálculo para las condiciones: **And/Or** - deben cumplirse todas las condiciones; las condiciones que tengan el mismo nombre de etiqueta se agruparán mediante la condición Or **Or** - basta con que se cumpla una condición Al filtrar, las etiquetas especificadas aquí se mostrarán primero junto con el problema, a menos que se anulen mediante la lista de *Tag display priority* (véase más abajo).
Show symptoms	Marque la casilla para mostrar en su propia línea los problemas clasificados como síntomas.
Show suppressed problems	Marque la casilla para mostrar los problemas que, de otro modo, estarían suprimidos y ocultos debido al mantenimiento del host o suprimidos manualmente por un usuario.
Acknowledgement status	Filtrar para mostrar todos los problemas, solo los problemas no reconocidos o solo los problemas reconocidos. Marque la casilla adicional para excluir aquellos problemas que alguna vez hayan sido reconocidos por usted.

Pestañas para filtros favoritos

Los conjuntos de parámetros de filtro que se usan con frecuencia se pueden guardar en pestañas.

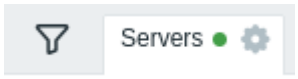
Para guardar un nuevo conjunto de parámetros de filtro, abra la pestaña principal y configure los ajustes del filtro; luego haga clic en el botón *Save as*. En una nueva ventana emergente, defina las *Filter properties*.



Parameter	Description
Name	El nombre del filtro que se mostrará en la lista de pestañas.
Show number of records	Marque esta opción si desea que se muestre el número de problemas junto al nombre de la pestaña.
Override time period selector	Marque esta opción para establecer un período de tiempo predeterminado específico para este conjunto de filtros. Si se establece, solo podrá cambiar el período de tiempo de esta pestaña actualizando los ajustes del filtro. Para las pestañas sin un período de tiempo personalizado, el intervalo de tiempo se puede cambiar haciendo clic en el botón selector de tiempo en la esquina superior derecha (el nombre del botón depende del intervalo de tiempo seleccionado: This week, Last 30 minutes, Yesterday, etc.). Esta opción solo está disponible para filtros en *Monitoring* > *Problems*.
From/To	Inicio y fin del período de tiempo en sintaxis de tiempo absoluta (`Y-m-d H:i:s`) o relativa (`now-1d`). Disponible si está marcada la opción *Set custom time period*.

Una vez guardado, el filtro se crea como una pestaña de filtro con nombre y se activa inmediatamente.

Para editar las propiedades del filtro de un filtro existente, haga clic en el símbolo de engranaje junto al nombre de la pestaña activa.



Notas:

- Para ocultar el área de filtros, haga clic en el nombre de la pestaña actual. Haga clic de nuevo en el nombre de la pestaña activa para volver a abrir el área de filtros.
- Se admite la navegación con teclado: use las flechas para cambiar entre pestañas y pulse *Enter* para abrir.
- Los botones izquierdo/derecho sobre el filtro se pueden usar para cambiar entre filtros guardados. Como alternativa, el botón que apunta hacia abajo abre un menú desplegable con todos los filtros guardados y puede hacer clic en el que necesite.
- Si se han cambiado los ajustes de un filtro guardado (pero no se han guardado), se muestra un punto verde después del nombre del filtro. Para actualizar el filtro según los nuevos ajustes, haga clic en el botón *Update*, que se muestra en lugar del botón *Save as*.
- El diseño de las columnas de la tabla se guarda por pestaña de filtro. Para personalizar el diseño de un filtro guardado, abra primero esa pestaña de filtro y luego personalice las columnas.
- Las pestañas de filtro, los ajustes del filtro y el diseño de las columnas de la tabla para cada pestaña se recuerdan en el perfil del usuario y se restauran cuando el usuario vuelve a abrir la página.

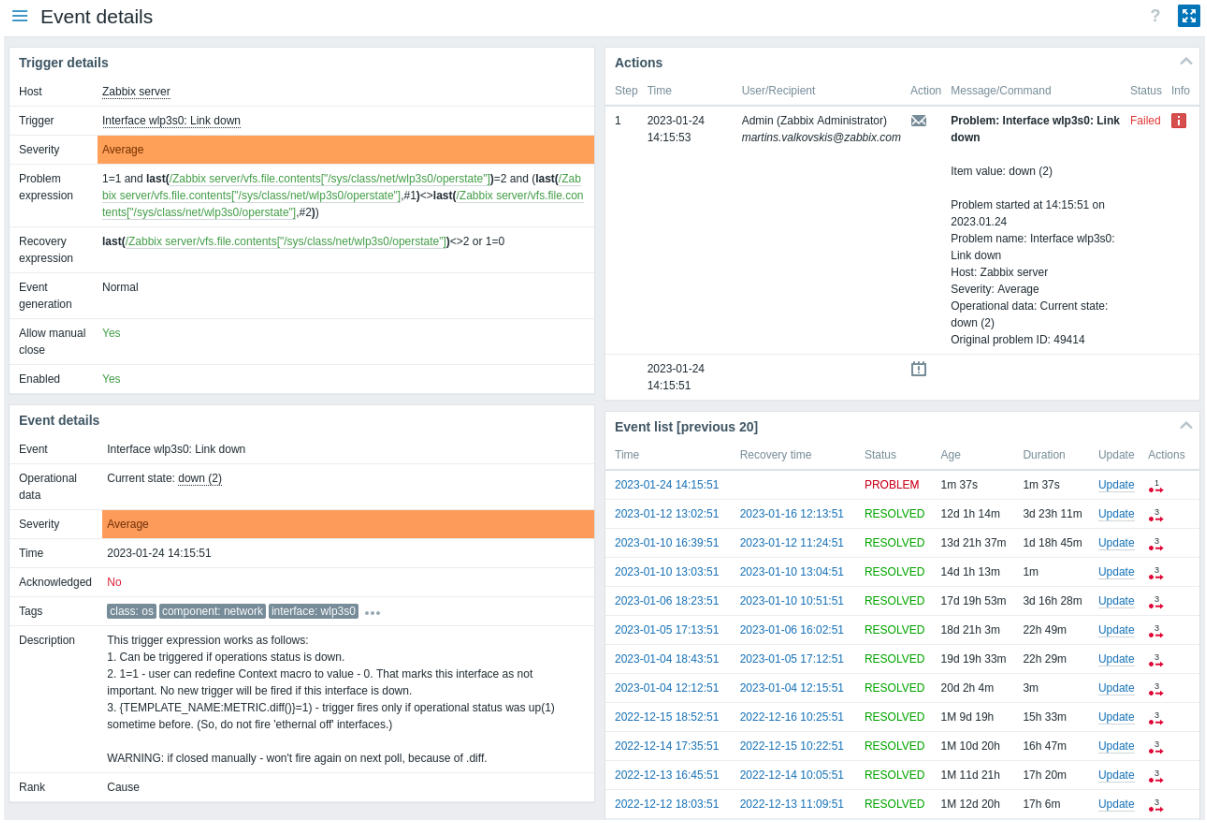
Note:
Para compartir filtros, copie y envíe a otras personas la URL de un filtro activo. Después de abrir esta URL, otros usuarios podrán guardar este conjunto de parámetros como un filtro permanente en su cuenta de Zabbix.
 Véase también: Page parameters.

Botones de filtro

Apply	Aplicar los criterios de filtrado especificados (sin guardar).
Reset	Restablecer el filtro actual y volver a los parámetros guardados de la pestaña actual. En la pestaña principal, esto limpiará el filtro.
Save as	Guardar los parámetros actuales del filtro en una nueva pestaña. Solo disponible en la pestaña principal.
Update	Reemplazar los parámetros de la pestaña con los parámetros especificados actualmente. No disponible en la pestaña principal.

Visualización de detalles

Las horas de inicio del problema y de recuperación en *Monitoring* > *Problems* son enlaces. Al hacer clic en ellos se abren más detalles del evento.



En los detalles del evento, puede ver si el evento fue suprimido debido a mantenimiento o suprimido manualmente por un usuario.

Tenga en cuenta que la severidad del problema puede diferir para el trigger y el evento del problema, si se ha actualizado para el evento del problema mediante la pantalla *Update problem* screen.

En la lista de acciones, se utilizan los siguientes iconos para indicar el tipo de actividad:

- - se generó el evento del problema.
- - se ha enviado un mensaje.
- - evento del problema reconocido.
- - evento del problema no reconocido.
- - se ha añadido un comentario.
- - la severidad del problema se ha incrementado (por ejemplo, Information > Warning).
- - la severidad del problema se ha reducido (por ejemplo, Warning > Information).

-  - la severidad del problema se ha cambiado, pero ha vuelto al nivel original (por ejemplo, Warning > Information > Warning).
-  - se ha ejecutado un comando remoto.
-  - el evento del problema se ha recuperado.
-  - el problema se ha cerrado manualmente.
-  - el problema ha sido suprimido.
-  - el problema ha dejado de estar suprimido.
-  - el problema ha sido suprimido debido al mantenimiento del host.
-  - el problema ha dejado de estar suprimido después del mantenimiento del host.
-  - el problema se ha convertido en un problema de síntoma.
-  - el problema se ha convertido en un problema de causa.

Causa y problemas de síntomas

Descripción general

Por defecto, todos los nuevos problemas se clasifican como problemas causa. Es posible reclasificar manualmente ciertos problemas como problemas síntoma del problema causa.

Por ejemplo, un corte de energía puede ser la causa raíz real de que algún equipo no sea accesible o de que algún servicio esté caído. En este caso, los problemas ”el equipo no es accesible” y ”el servicio está caído” deben clasificarse como problemas síntoma de ”corte de energía”, que es el problema causa.

La jerarquía causa-síntoma solo admite dos niveles. Un problema que ya es un síntoma no puede tener problemas síntoma ”subordinados”; cualquier problema asignado como síntoma a un problema síntoma se convertirá en síntoma del mismo problema causa.

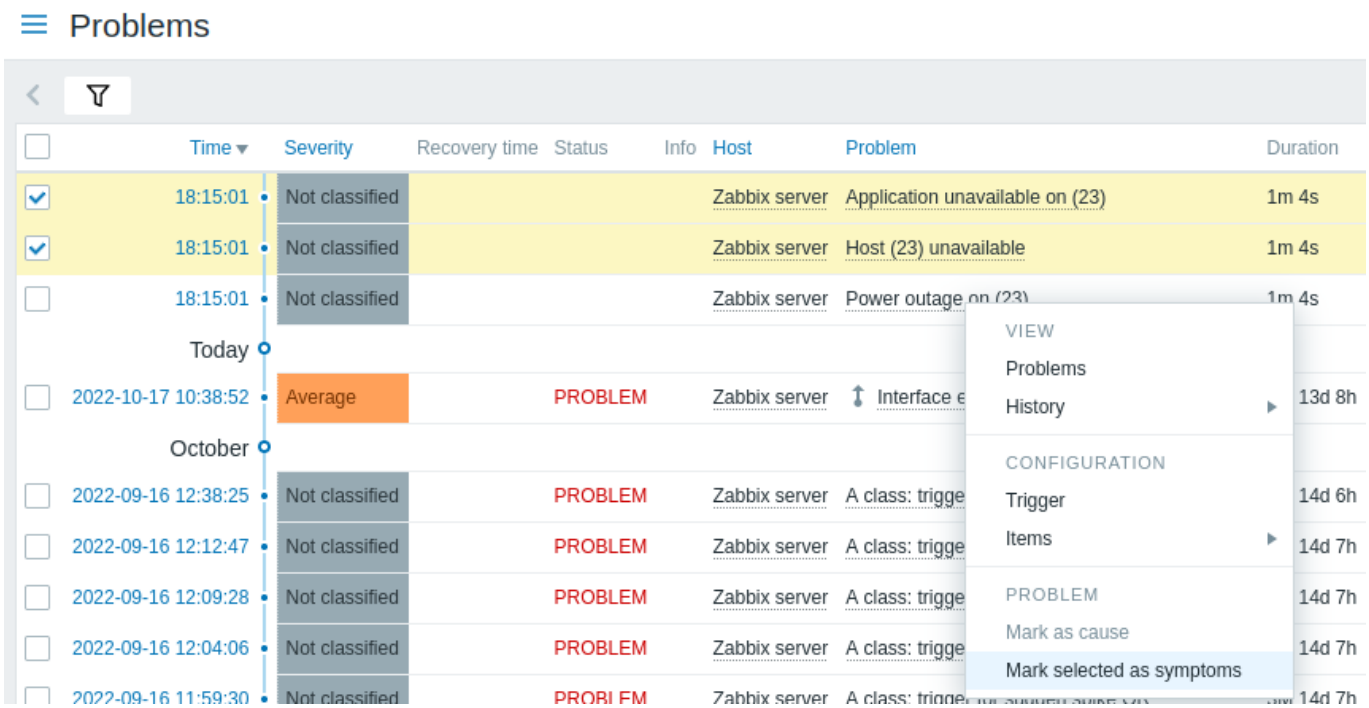
Solo los problemas causa se cuentan en los totales de problemas en mapas, widgets del panel como *Problemas por gravedad* o *Equipos con problemas*, etc. Sin embargo, la clasificación de problemas no afecta a los servicios.

Un problema síntoma solo puede estar vinculado a un problema causa. Los problemas síntoma no se resuelven automáticamente si el problema causa se resuelve o se cierra.

Configuración

Para reclasificar un problema como problema síntoma, primero selecciónelo en la lista de problems. Se puede seleccionar uno o varios problemas.

Luego vaya al problema causa y, en su menú contextual, haga clic en la opción *Mark selected as symptoms*.



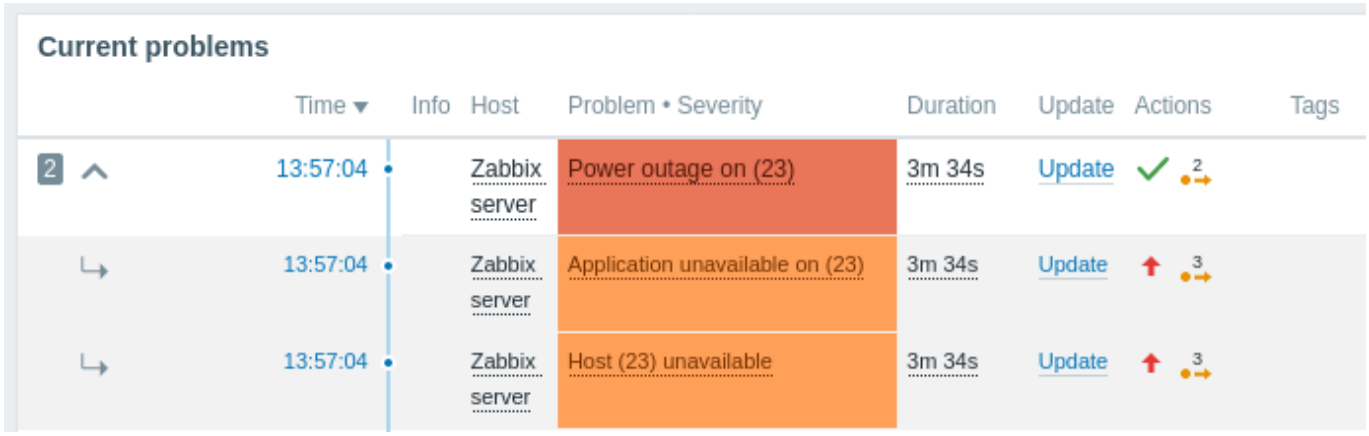
Después de eso, los problemas seleccionados serán actualizados por el server a problemas síntoma del problema causa.

Mientras se actualiza el estado del problema, se muestra de una de estas dos maneras:

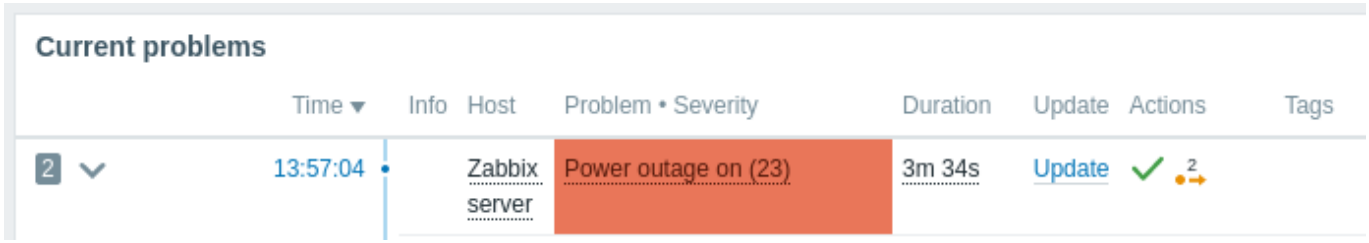
- Se muestra un estado parpadeante "UPDATING" en la columna Status;
- Un icono parpadeante o en la columna Info (esto aplica si en el filtro solo se seleccionan *Problems* y, por lo tanto, no se muestra la columna Status).

Visualización

Los problemas de síntoma se muestran debajo del problema causa y se marcan en consecuencia en *Monitoring* > *Problems* (y en el widget del panel *Problems*) - con un icono, una fuente más pequeña y un fondo diferente.



En la vista contraída, solo se ve el problema causa; la existencia de problemas de síntoma se indica mediante el número al comienzo de la línea y el icono para expandir la vista.



También es posible mostrar adicionalmente los problemas de síntoma con fuente normal y en su propia línea. Para ello, seleccione *Show symptoms* en la configuración del filtro o en la configuración del widget.

Revertir a problema causa

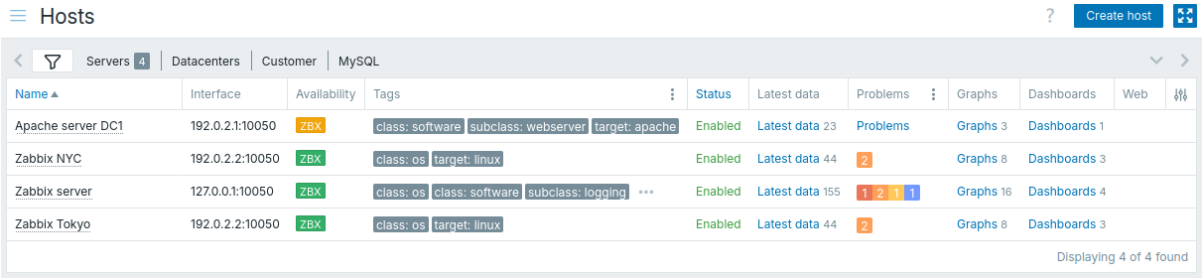
Un problema síntoma puede revertirse a un problema causa. Para hacerlo, puede:

- hacer clic en la opción *Marcar como causa* en el menú contextual del problema síntoma;
- marcar la opción *Convertir a causa* en la pantalla de actualización de problemas y hacer clic en *Actualizar* (esta opción también funcionará si se seleccionan varios problemas).

Hosts

Resumen

La sección *Monitoring > Hosts* muestra una lista completa de los hosts monitorizados con información detallada sobre la interfaz del host, disponibilidad, etiquetas, problemas actuales, estado (habilitado/deshabilitado) y enlaces para navegar fácilmente a los últimos datos del host, el historial de problemas, gráficos, dashboards y escenarios web.



Los detalles del host se muestran en una tabla personalizable. El botón *Customize table* en el encabezado de la tabla permite mostrar, ocultar o reordenar columnas, y restablecer el diseño predeterminado. También puede cambiar el tamaño de cada columna arrastrando su borde derecho; haga doble clic en el borde para restablecerlo. Todas las personalizaciones se aplican de inmediato y siempre se recuerdan para el usuario actual dentro de la pestaña activa de filtro (predeterminada o guardada).

Las siguientes columnas están disponibles:

Column	Description
Name	El nombre visible del host. Al hacer clic en el nombre se abre el menú del host. Un icono de llave inglesa naranja después del nombre indica que este host está en mantenimiento. Haga clic en el encabezado de la columna para ordenar los hosts por nombre en orden ascendente o descendente.
Interface	Se muestra la interfaz principal del host.

Column	Description
Availability	Se muestra la disponibilidad del host por cada interfaz configurada. Los iconos de disponibilidad representan el estado actual de la interfaz del host en el server de Zabbix. Por lo tanto, si deshabilita un host en el frontend, su disponibilidad se actualizará después de que el server de Zabbix haya sincronizado los cambios de configuración. Del mismo modo, si habilita un host, su disponibilidad se actualizará después de que el server de Zabbix haya sincronizado los cambios de configuración y haya sondeado el host. Los iconos de disponibilidad representan solo los tipos de interfaz configurados (Agent, SNMP, IPMI, JMX). Al pasar el cursor sobre el icono se muestra una ventana emergente con una lista de todas las interfaces del mismo tipo, con detalles, estado y errores. Para la interfaz Agent, la ventana emergente muestra las interfaces (pasivas) y las comprobaciones activas. Si un host tiene solo comprobaciones activas, se muestra el icono de la interfaz Agent incluso si el host no tiene configurada una interfaz Agent. La columna está vacía para los hosts sin interfaces. El estado de una sola interfaz del host se determina por la conexión entre un item habilitado que usa la interfaz y el host. El estado puede ser: **Disponible** - la conexión con el host se realizó correctamente; **No disponible** - la conexión con el host no se realizó correctamente (tiempo de espera agotado, problemas de firewall, etc.); **Desconocido** - no se ha intentado la conexión con el host o el resultado es desconocido. Para obtener más detalles sobre cómo el server de Zabbix determina el estado de la interfaz, consulte Estado desconocido de la interfaz y Ajustes de interfaz de host inaccesible/no disponible. El estado de todas las interfaces del host de un mismo tipo (Agent, SNMP, IPMI, JMX) lo determinan aquellas interfaces que son utilizadas por al menos un item habilitado. El estado se indica mediante el color del icono: **Verde** - todas las interfaces están disponibles; **Amarillo** - al menos una interfaz no está disponible, y al menos una está disponible o es desconocida; **Rojo** - todas las interfaces no están disponibles; **Gris** - al menos una interfaz es desconocida, pero ninguna no está disponible. **Disponibilidad de comprobaciones activas.** Si al menos una comprobación activa está habilitada en el host, las comprobaciones activas también afectan a la disponibilidad total de la interfaz Agent, como se describe arriba. Para determinar la disponibilidad de las comprobaciones activas, se envían mensajes de latido en el hilo de comprobación activa del agent. La frecuencia de los mensajes de latido se controla mediante el parámetro `HeartbeatFrequency` en la configuración de Zabbix agent o agent 2 (valor predeterminado 60 segundos, rango 0-3600). Las comprobaciones activas se consideran no disponibles cuando el latido de la comprobación activa tiene más de 2 x HeartbeatFrequency segundos. **Nota:** Los agentes de Zabbix anteriores a la versión 6.2.x no envían latidos de comprobaciones activas, por lo que la disponibilidad de sus hosts permanece desconocida.
Tags	Se muestran las etiquetas del host. Al pasar el cursor o hacer clic en el icono de tres puntos ••• después de las etiquetas mostradas, se abre una ventana emergente con todas las etiquetas. Al hacer clic en el botón ⋮ *Options* en el encabezado de la columna, puede establecer el número de etiquetas mostradas (1–3), elegir cómo se muestran los nombres de las etiquetas (completos, abreviados o ninguno), definir la prioridad de visualización de las etiquetas o duplicar la columna (las columnas duplicadas se pueden renombrar o eliminar).
Tag value	Se muestran los valores de las etiquetas del host. Al hacer clic en el botón ⋮ *Options* en el encabezado de la columna, puede cambiar el nombre de la columna, elegir qué valores de etiqueta mostrar o duplicar la columna (las columnas duplicadas se pueden renombrar o eliminar).
Status	Estado del host: *Enabled* o *Disabled*. Haga clic en el encabezado de la columna para ordenar los hosts por estado en orden ascendente o descendente.

Column	Description
Latest data	Al hacer clic en el enlace se abrirá la sección *Monitoring* > *Latest data* con todos los datos más recientes recopilados del host. Se muestra en gris el número de items con datos recientes.
Problems	Se muestra el número de problemas abiertos de un host, ordenados por severidad. El color de cada icono de problema muestra la severidad, mientras que el número muestra la cantidad de problemas para esa severidad. Al hacer clic en un icono se abre *Monitoring* > *Problems* para ese host; si un host no tiene problemas, se muestra en su lugar un enlace *Problems*. Al hacer clic en el botón ⋮ *Options* en el encabezado de la columna, obtiene acceso a la opción *Show suppressed problems*, que incluye los problemas que de otro modo estarían suprimidos (ocultos) debido al mantenimiento del host o a la supresión de problemas individual.
Graphs	Al hacer clic en el enlace se mostrarán los gráficos configurados para el host. El número de gráficos se muestra en gris. Si un host no tiene gráficos, no se muestra ningún enlace.
Dashboards	Al hacer clic en el enlace se mostrarán los dashboards configurados para el host. El número de dashboards se muestra en gris. Si un host no tiene dashboards, no se muestra ningún enlace.
Web	Al hacer clic en el enlace se mostrarán los escenarios web configurados para el host. El número de escenarios web se muestra en gris. Si un host no tiene escenarios web, no se muestra ningún enlace.
Custom text	Se muestra texto personalizado, valores de macros o una combinación de ambos. Al hacer clic en el botón ⋮ *Options* en el encabezado de la columna, puede cambiar el nombre de la columna, introducir el texto personalizado o las macros, o duplicar la columna (las columnas duplicadas se pueden renombrar o eliminar). Macros compatibles: `{HOST.*}` (excepto `{HOST.METADATA}`), `{INVENTORY.*}` y macros de usuario.

Botones

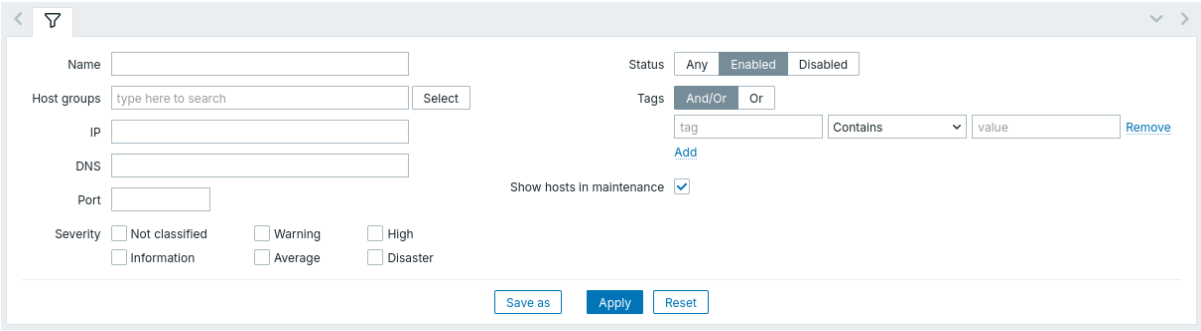
Crear host permite crear un nuevo host. Este botón solo está disponible para usuarios administradores y superadministradores.

Los botones del modo de vista, comunes a todas las secciones, se describen en la página de Monitorización.

Uso del filtro

Puede usar el filtro para mostrar solo los hosts que le interesan. Para mejorar el rendimiento de la búsqueda, los datos se buscan con las macros sin resolver.

El filtro se encuentra encima de la tabla. Es posible filtrar hosts por nombre, grupo de hosts, IP o DNS, puerto de la interfaz, etiquetas, gravedad del problema, estado (habilitado/deshabilitado/cualquiera); también puede seleccionar si desea mostrar hosts que actualmente están en mantenimiento.



Parameter	Description
Name	Filtrar por el nombre visible del host.
Host groups	Filtrar por uno o más grupos de hosts. Especificar un grupo de hosts padre selecciona implícitamente todos los grupos de hosts anidados.
IP	Filtrar por dirección IP.
DNS	Filtrar por nombre DNS.
Port	Filtrar por número de puerto.
Severity	Filtrar por gravedad del problema. De forma predeterminada, se muestran problemas de todas las gravedades. Los problemas se muestran si no están suprimidos.

Parameter	Description
Status	Filtrar por el estado del host.
Tags	Filtrar por nombre y valor de la etiqueta del host. Los hosts se pueden filtrar tanto por etiquetas a nivel de host como por etiquetas de todas las templates vinculadas, incluidas las templates anidadas. Es posible incluir y excluir etiquetas y valores de etiqueta específicos. Se pueden establecer varias condiciones. La coincidencia del nombre de la etiqueta siempre distingue entre mayúsculas y minúsculas. Hay varios operadores disponibles para cada condición: **Exists** - incluir los nombres de etiqueta especificados **Equals** - incluir los nombres y valores de etiqueta especificados (distingue entre mayúsculas y minúsculas) **Contains** - incluir los nombres de etiqueta especificados cuyos valores de etiqueta contengan la cadena introducida (coincidencia de subcadena, no distingue entre mayúsculas y minúsculas) **Does not exist** - excluir los nombres de etiqueta especificados **Does not equal** - excluir los nombres y valores de etiqueta especificados (distingue entre mayúsculas y minúsculas) **Does not contain** - excluir los nombres de etiqueta especificados cuyos valores de etiqueta contengan la cadena introducida (coincidencia de subcadena, no distingue entre mayúsculas y minúsculas) Hay dos tipos de cálculo para las condiciones: **And/Or** - deben cumplirse todas las condiciones; las condiciones que tengan el mismo nombre de etiqueta se agruparán mediante la condición Or **Or** - basta con que se cumpla una condición
Show hosts in maintenance	Marque la casilla para mostrar los hosts que están en mantenimiento (se muestra de forma predeterminada).

Guardar el filtro

La configuración del filtro favorito se puede guardar como pestañas y luego acceder a ellas rápidamente haciendo clic en la pestaña correspondiente encima del filtro.

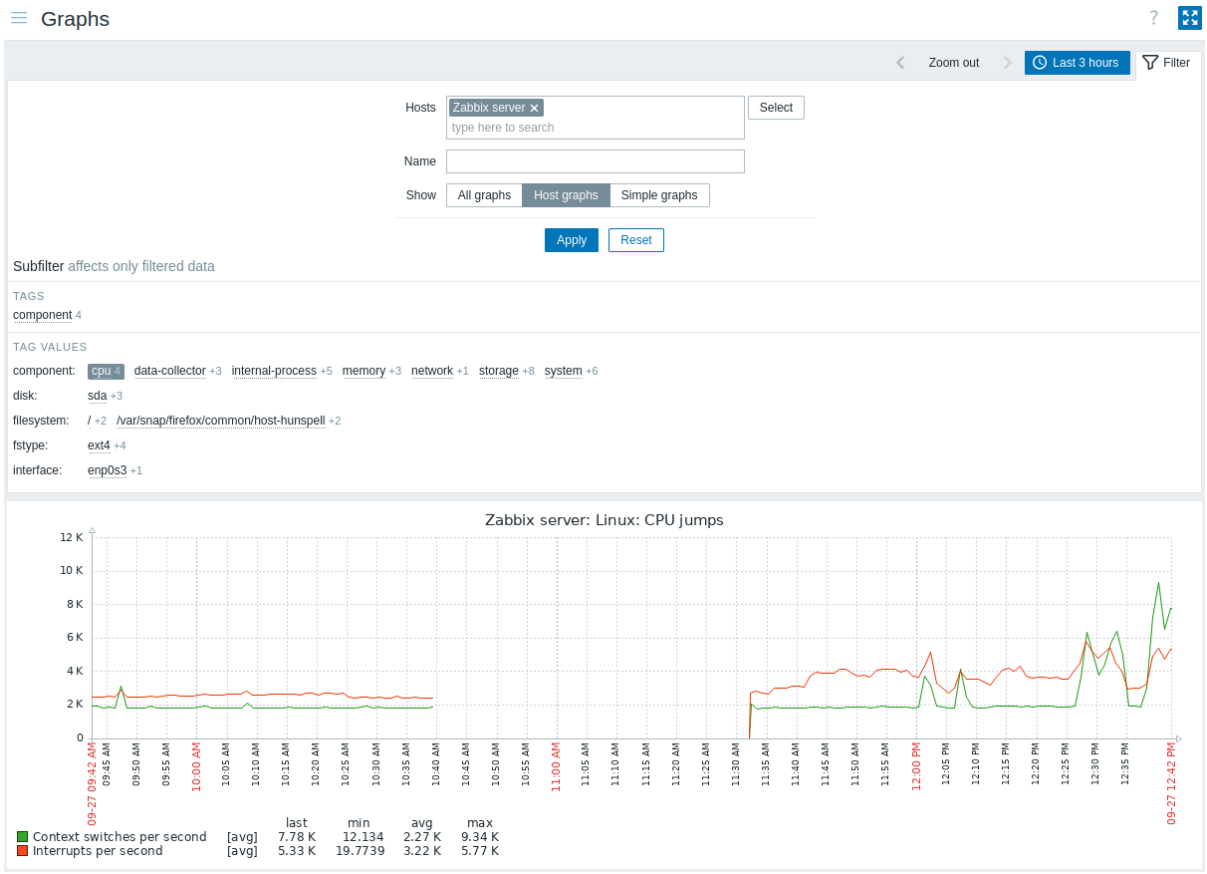
Consulte más detalles sobre guardar filtros.

Gráficos

Resumen

Se puede acceder a los gráficos del host desde *Monitoring* > *Hosts* haciendo clic en Graphs para el host correspondiente.

Se puede mostrar cualquier custom graph que se haya configurado para el host, así como cualquier simple graph.



Los gráficos se ordenan por:

- nombre del gráfico (custom graphs)
- nombre del item (simple graphs)

También se puede acceder a los gráficos de los hosts deshabilitados.

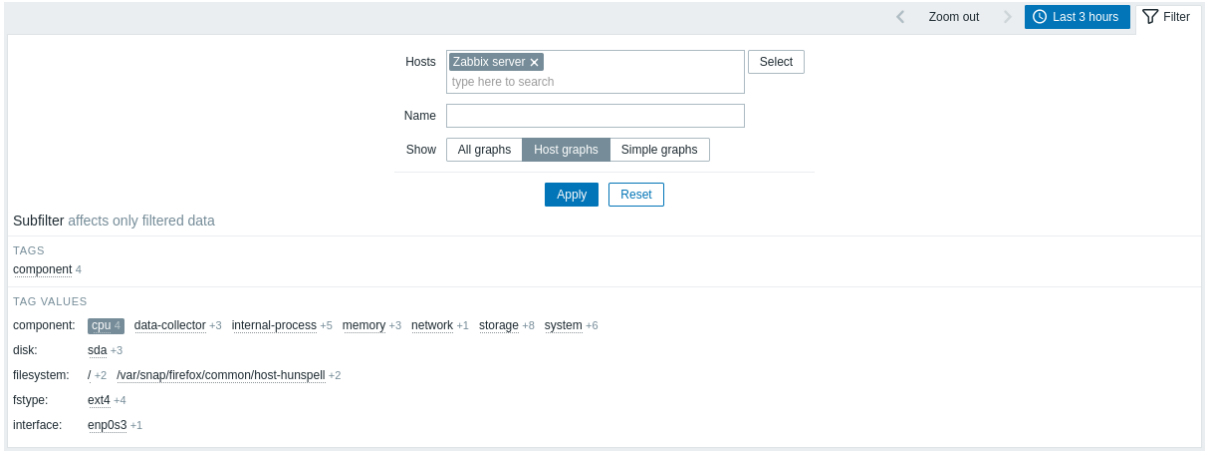
Selector de período de tiempo

Tenga en cuenta el selector de período de tiempo sobre el gráfico. Permite seleccionar períodos frecuentemente requeridos con un solo clic del ratón.

Consulte también: *Selector de período de tiempo*

Uso del filtro

Para ver un gráfico específico, selecciónelo en el filtro. El filtro permite especificar el host, el nombre del gráfico y la opción *Mostrar* (todos/gráficos del host/gráficos simples).



Si no se selecciona ningún host en el filtro, no se muestran gráficos.

Uso del subfiltro

El subfiltro permite modificar aún más el filtrado aplicado por el filtro principal.

Contiene enlaces en los que se puede hacer clic para acceder rápidamente a los gráficos relacionados. Los gráficos están relacionados por una entidad común: el nombre o el valor de una etiqueta. Al hacer clic en un nombre o valor de etiqueta, este se resalta con un fondo gris y los gráficos se filtran inmediatamente (no es necesario hacer clic en *Aplicar* en el filtro principal). Al hacer clic en otro nombre o valor de etiqueta, este se añade a los resultados filtrados. Al volver a hacer clic en el nombre o valor de etiqueta, se elimina el filtrado.

Los subfiltros se generan a partir de los datos filtrados, que están limitados a 1000 registros. Si desea ver más registros en el subfiltro, debe aumentar el valor del parámetro *Límite para los resultados de búsqueda y filtrado* (en *Administración > General > Interfaz gráfica de usuario*).

A diferencia del filtro principal, el subfiltro se actualiza con cada solicitud de actualización de la tabla para obtener siempre información actualizada sobre las opciones de filtrado disponibles y sus números de contador.

El número de entidades mostradas está limitado a 100 horizontalmente. Si hay más, se muestra un icono de tres puntos al final; no se puede hacer clic en él. Las listas verticales (como las etiquetas con sus valores) están limitadas a 20 entradas. Si hay más, se muestra un icono de tres puntos; no se puede hacer clic en él.

El número situado junto a cada entidad en la que se puede hacer clic indica el número de gráficos que tiene en los resultados del filtro principal.

Una vez seleccionada una entidad, los números de las demás entidades disponibles se muestran con un signo más, que indica cuántos gráficos pueden añadirse a la selección actual.

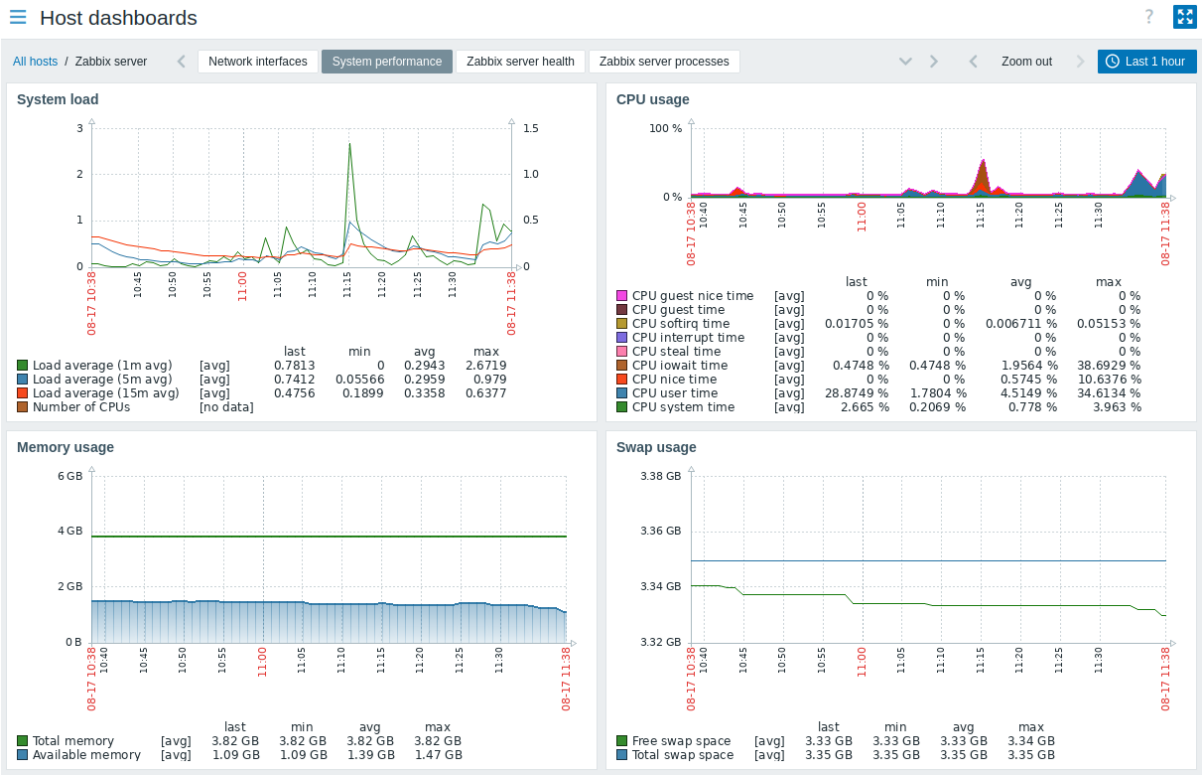
Botones

Los botones del modo de vista, comunes a todas las secciones, se describen en la página Monitoring.

Paneles de host

Descripción general

Los dashboards de host tienen un aspecto similar al de los dashboards globales; sin embargo, los dashboards de host no tienen un propietario y muestran datos únicamente del host seleccionado.



Al visualizar los dashboards de host, puede cambiar entre los dashboards configurados haciendo clic en:

- las pestañas de los dashboards;
- los botones de flecha  situados debajo del encabezado;
- el botón de flecha  situado debajo del encabezado, que mostrará la lista completa de dashboards de host disponibles.

Para cambiar a la sección *Monitoring* > *Hosts*, haga clic en el enlace de navegación *All hosts*, situado debajo del encabezado, en la esquina superior izquierda.

Configuración

Los paneles de equipos se configuran a nivel de plantilla. Una vez que una plantilla se vincula a un equipo, se generan paneles de equipo para ese equipo. Tenga en cuenta que los paneles de equipos *no pueden* configurarse en la sección *Paneles*, que está reservada para paneles globales.

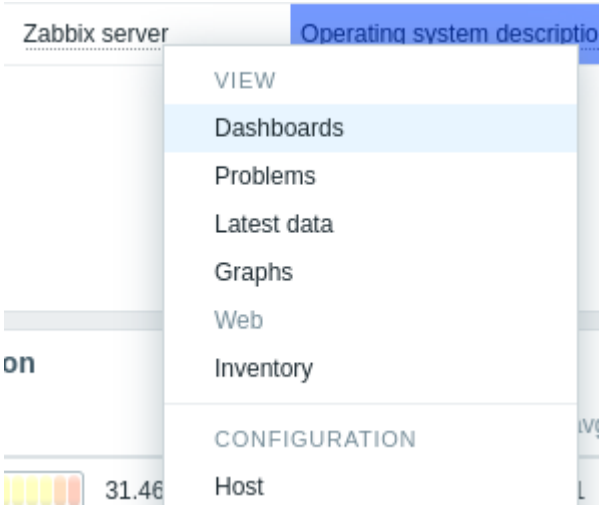
Los widgets de los paneles de equipos también solo pueden configurarse a nivel de plantilla, excepto para cambiar el intervalo de actualización. Además, los widgets de los paneles de equipos solo pueden copiarse a otros paneles de equipos dentro de la misma plantilla. Tenga en cuenta que los widgets de los paneles globales *no pueden* copiarse a los paneles de equipos.

Note:
Los paneles de equipos solían ser pantallas de equipos antes de Zabbix 5.2. Al importar una plantilla antigua que contiene pantallas, la importación de la pantalla será ignorada.

Acceso

Se puede acceder a los dashboards del host:

- después de buscar un nombre de host en búsqueda global (haga clic en el enlace *Dashboards* proporcionado en los resultados de la búsqueda);
- después de hacer clic en un nombre de host en *Inventory* > *Hosts* (haga clic en el enlace *Dashboards* proporcionado en la vista general del host);
- desde el menú del host haciendo clic en *Dashboards*.

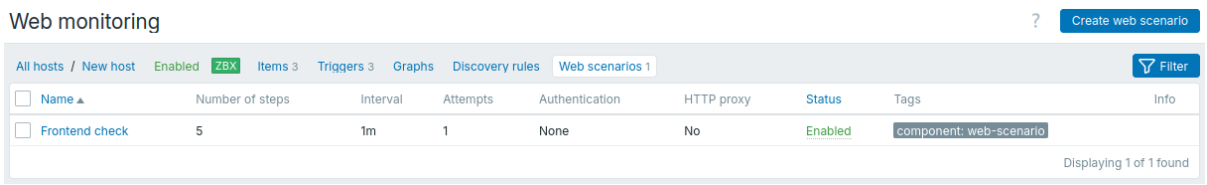


Tenga en cuenta que los dashboards del host *no se pueden* acceder directamente en la sección *Dashboards*, que está reservada para dashboards globales.

Escenarios web

Descripción general

Se puede acceder a la información del escenario web del host desde *Monitoring* > *Hosts* haciendo clic en Web para el host correspondiente.

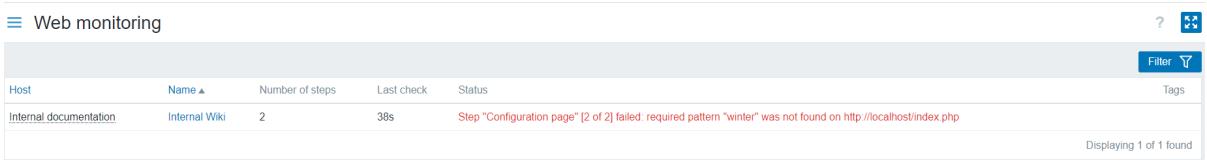


Al hacer clic en el nombre del host se abre el menú del host. También se puede acceder a los datos de los hosts deshabilitados. El nombre de un host deshabilitado se muestra en rojo.

El estado de un escenario web puede ser:

- OK (verde) - todos los pasos se han completado correctamente y Zabbix ha recopilado datos del escenario.

- Error (rojo) - uno o más pasos han fallado (consulte el mensaje de error para obtener pistas sobre qué parámetros pueden necesitar corrección).
- Unknown (gris) - aún no hay estado de pasos disponible o no se han recopilado datos.



El número máximo de escenarios mostrados por página depende de la configuración de perfil de usuario *Rows per page* setting.

De forma predeterminada, solo se muestran los valores que se encuentran dentro de las últimas 24 horas. Este límite se ha introducido con el objetivo de mejorar los tiempos de carga iniciales en páginas grandes de datos más recientes. Puede ampliar este período de tiempo cambiando el valor del parámetro *Max history display period* en la sección de menú *Administration > General > GUI*.

El nombre del escenario es un enlace a estadísticas más detalladas sobre él:



Uso del filtro

La página muestra una lista de todos los escenarios web del host seleccionado. Para ver escenarios web de otro host o grupo de hosts sin volver a la página *Monitoring > Hosts*, seleccione ese host o grupo en el filtro. También puede filtrar escenarios en función de etiquetas.

Botones

Los botones del modo de vista, comunes a todas las secciones, se describen en la página Monitoring.

Últimos datos

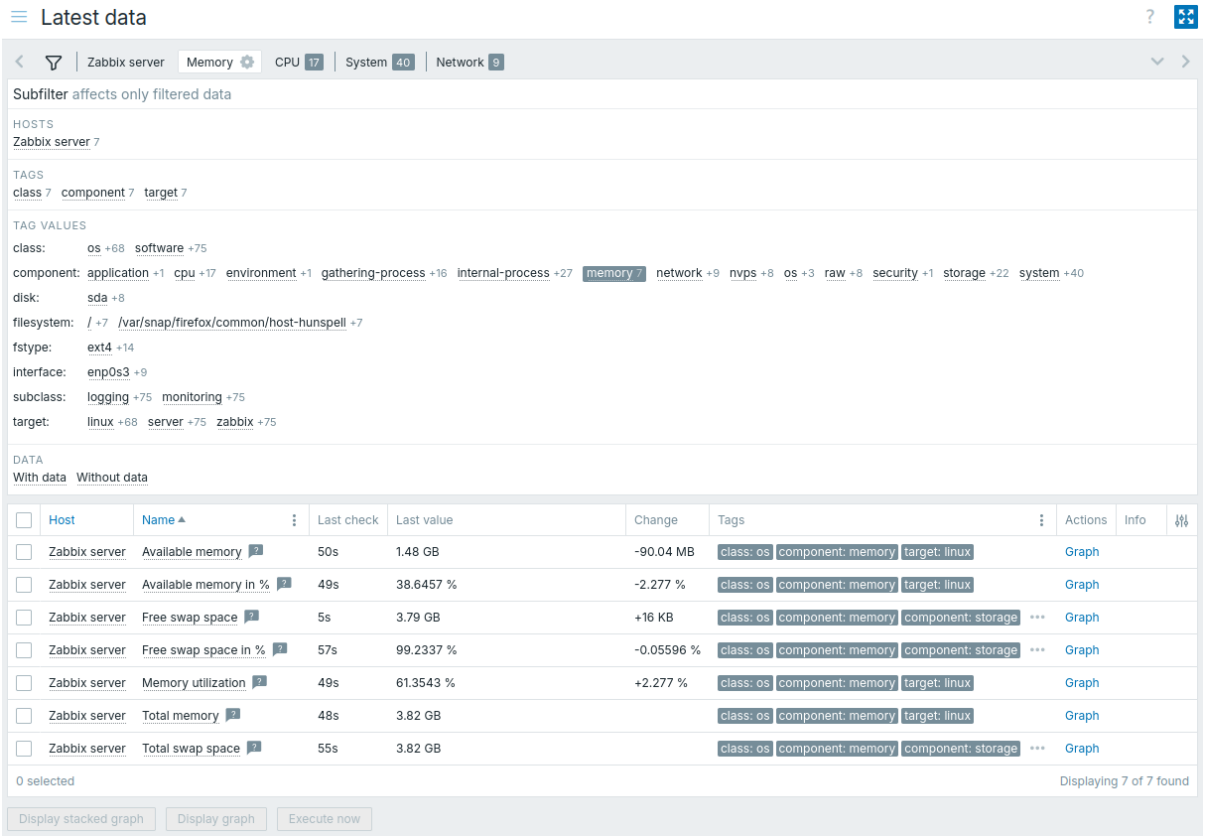
Overview

La sección *Monitoring* > *Latest data* muestra los últimos valores recopilados por los items.

Esta sección contiene los siguientes elementos:

- Filter
- Subfilter
- Lista de items

Note:
El subfilter y la lista de items se muestran solo si el filter está configurado y hay resultados para mostrar.



Los detalles del último valor se muestran en una tabla personalizable. El botón *Customize table* en el encabezado de la tabla permite mostrar, ocultar o reordenar columnas, y restablecer el diseño predeterminado. También puede cambiar el tamaño de cada columna arrastrando su borde derecho; haga doble clic en el borde para restablecerlo. Todas las personalizaciones se aplican de inmediato y siempre se recuerdan para el usuario actual dentro de la pestaña activa de filter (predeterminada o guardada).

Las siguientes columnas están disponibles:

Column	Description
Host	Nombre del host al que pertenece el item. Al hacer clic en el nombre se abre el host context menu. Si un host está en mantenimiento, se muestra después del nombre del host un icono naranja de llave inglesa . Si un host está deshabilitado, el nombre del host se muestra en rojo. Tenga en cuenta que los datos de los hosts deshabilitados (incluidos los gráficos y las listas de valores de items) son accesibles en la sección *Latest data*.
Name	Nombre del item. Al hacer clic en el nombre se abre el item menu. Se muestra un icono de signo de interrogación junto al nombre del item para todos los items que tienen una descripción. Pase el cursor sobre el icono para mostrar una información emergente con la descripción del item. Al hacer clic en el botón *Options* en el encabezado de la columna, puede acceder a la opción *Show item key*, que muestra la clave del item debajo del nombre de cada item.
Interval	Intervalo de actualización del item.
History	Período de almacenamiento del historial del item.
Trends	Período de almacenamiento de tendencias del item.

Column	Description
Type	Tipo del item.
Last check	Tiempo transcurrido desde la última comprobación del item.
Last value	Valor más reciente del item. Los valores se muestran con conversión de unidades y asignación de valores aplicadas. Pase el cursor sobre el valor para mostrar los datos sin procesar (truncados a 8192 caracteres). Los valores binarios se muestran como un marcador de posición (*binary value*) en lugar del valor real. De forma predeterminada, solo se muestran los valores recibidos en las últimas 24 horas. Este límite mejora los tiempos de carga iniciales de páginas grandes de latest data; para ampliarlo, actualice el valor del parámetro *Max history display period* en *Administration > General > GUI*.
Change	Diferencia entre el valor anterior y el valor más reciente. Para items con una frecuencia de actualización de 1 día o más, la cantidad de cambio nunca se mostrará (con la configuración predeterminada). En este caso, el último valor no se mostrará en absoluto si se recibió hace más de 24 horas.
Tags	Se muestran las etiquetas asociadas con el item. Al hacer clic en una etiqueta, esta se habilita en el subfilter, filtrando así la lista de items por esa etiqueta (y por cualquier otra etiqueta seleccionada en el subfilter). Una vez pulsadas, esas etiquetas dejan de ser clicables; para ajustar el filtrado por etiquetas, debe usarse el subfilter. Al hacer clic en el botón *Options* en el encabezado de la columna, puede establecer el número de etiquetas mostradas (1-3), elegir cómo se muestran los nombres de las etiquetas (completos, abreviados o ninguno), definir la prioridad de visualización de las etiquetas o duplicar la columna (las columnas duplicadas pueden renombrarse o eliminarse).
Tag value	Se muestran los valores de las tag del item. Al hacer clic en el botón *Options* en el encabezado de la columna, puede cambiar el nombre de la columna, elegir qué valores de etiqueta mostrar o duplicar la columna (las columnas duplicadas pueden renombrarse o eliminarse).
Actions	Enlace al simple graph/history de los valores del item.
Info	Información adicional sobre el item. Si un item tiene errores (por ejemplo, ha pasado a no ser compatible), se muestra un icono de información. Pase el cursor sobre el icono para ver los detalles.
Custom text	Se muestra texto personalizado, valores de macros o una combinación de ambos. Al hacer clic en el botón *Options* en el encabezado de la columna, puede cambiar el nombre de la columna, introducir el texto personalizado o las macros, o duplicar la columna (las columnas duplicadas pueden renombrarse o eliminarse). Supported macros: {HOST.*} (excepto {HOST.METADATA}), {INVENTORY.*}, {ITEM.*} (excepto {ITEM.STATE.ERROR}), y user macros.

Botones

Los botones del modo de vista, comunes para todas las secciones, se describen en la página Monitoring.

Acciones masivas

Los botones debajo de la lista ofrecen acciones masivas con uno o varios items seleccionados:

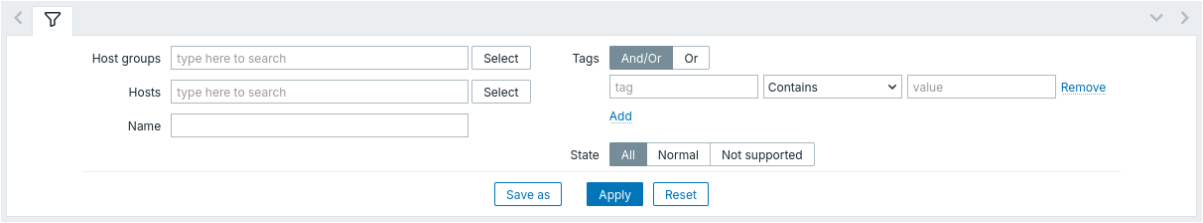
- *Mostrar gráfico apilado* - muestra un gráfico ad hoc apilado.
- *Mostrar gráfico* - muestra un gráfico ad hoc simple.
- *Ejecutar ahora* - ejecuta inmediatamente una comprobación de nuevos valores de item. Compatible solo con comprobaciones **pasivas** (consulte más detalles). Esta opción solo está disponible para hosts con acceso de lectura y escritura. El acceso a esta opción para hosts con permisos de solo lectura depende de la opción de rol de usuario llamada *Invocar "Ejecutar ahora" en hosts de solo lectura*.

Para usar estas opciones, marque las casillas antes de los items correspondientes y luego haga clic en el botón requerido.

Uso del filtro

Puede utilizar el filtro para mostrar únicamente los items que le interesan.
Para mejorar el rendimiento de la búsqueda, los datos se buscan con las macros sin resolver.

El icono  del filtro se encuentra encima de la lista de items y del subfiltro.
Haga clic en él para expandir el filtro.



El filtro permite limitar la lista de items por grupo de hosts, host, nombre del item, etiquetas y estado.
Especificar un grupo de hosts principal en el filtro selecciona implícitamente todos los grupos de hosts anidados.
Consulte *Monitoring* > *Problems* para obtener más información sobre el filtrado por etiquetas.

La opción de filtro *Show details* permite ampliar la información mostrada para los items.
Selecciónela para mostrar detalles como el intervalo de actualización del item, la configuración del historial y las tendencias, el tipo de item y los errores del item (correcto/no compatible).

Guardar filtro

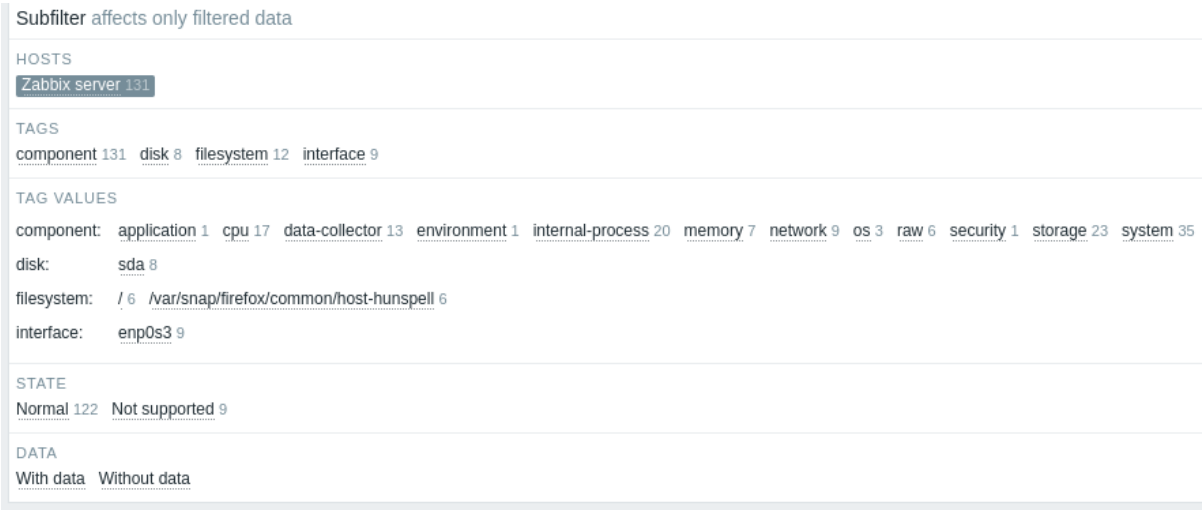
La configuración del filtro favorito se puede guardar como pestañas y luego acceder rápidamente haciendo clic en la pestaña correspondiente encima del filtro.

Consulte más detalles sobre guardar filtros.

Uso del subfiltro

El subfiltro permite modificar aún más el filtrado del filtro principal.

Contiene enlaces en los que se puede hacer clic para acceder rápidamente a los elementos relacionados. Los elementos están relacionados por una entidad común: host, nombre o valor de etiqueta, estado del item o estado de los datos. Al hacer clic en una entidad, esta se resalta con un fondo gris y los elementos se filtran inmediatamente (no es necesario hacer clic en *Aplicar* en el filtro principal). Al hacer clic en otra entidad, esta se añade a los resultados filtrados. Al volver a hacer clic en la entidad, se elimina el filtrado.



Los subfiltros se generan a partir de los datos filtrados, que están limitados a 1000 registros. Si hay 20 hosts, cada uno con 100 items (es decir, 2000 registros en total), solo la mitad de los hosts estará visible en el subfiltro. Si desea ver más registros en el subfiltro, debe aumentar el valor del parámetro *Límite para los resultados de búsqueda y filtrado* (en *Administración* > *General* > *GUI*).

A diferencia del filtro principal, el subfiltro se actualiza con cada solicitud de actualización de la tabla para mostrar siempre información actualizada sobre las opciones de filtrado disponibles y sus contadores.

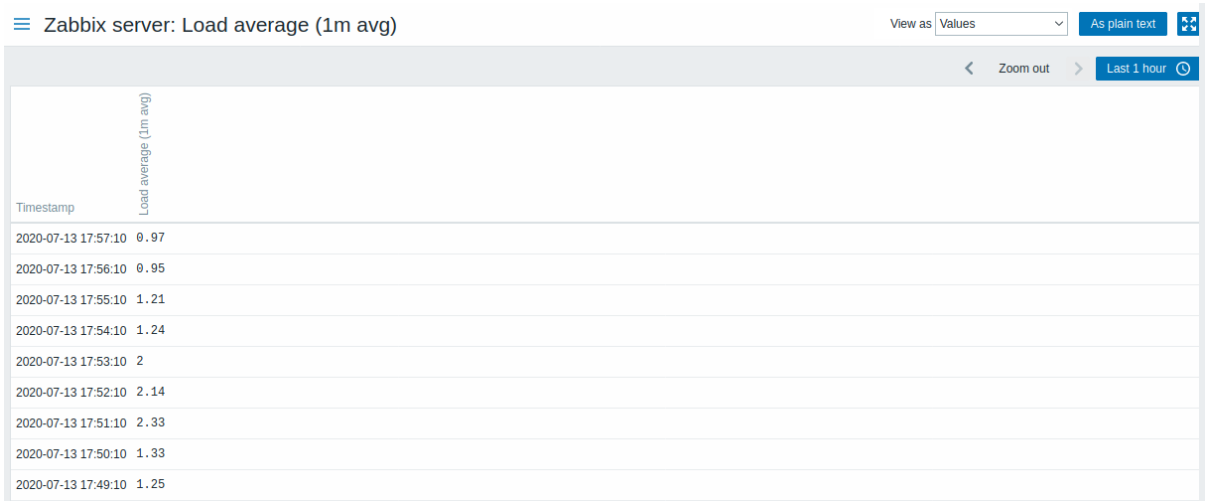
Para cada grupo de entidades (hosts, etiquetas, valores de etiquetas, etc.), se muestran hasta 10 filas de entidades. Si hay más entidades, esta lista se puede ampliar para mostrar un máximo de 1000 entradas (el valor de SUBFILTER_VALUES_PER_GROUP en las definiciones del frontend) haciendo clic en el icono de tres puntos ••• al final de la lista. En el caso de los *valores de etiquetas*, la lista se puede ampliar para mostrar un máximo de 200 nombres de etiquetas con sus valores correspondientes. Tenga en cuenta que, una vez ampliada por completo, la lista no se puede contraer.

Un número junto a cada entidad en la que se puede hacer clic indica el número de items agrupados en ella (según los resultados del filtro principal). Al hacer clic en una entidad, los números de las demás entidades disponibles se muestran con un signo más, que indica cuántos items se pueden añadir a la selección actual. Las entidades sin items no se muestran, a menos que se hayan seleccionado previamente en el subfiltro.

Gráficos e historial

La columna *Graph/History* en la lista de items ofrece los siguientes enlaces:

- **History** - para todos los items de texto, que llevan a listados (*Values/500 latest values*) que muestran el historial de valores anteriores del item.
- **Graph** - para todos los items numéricos, que llevan a un simple graph. Tenga en cuenta que, cuando se muestra el gráfico, un menú desplegable en la parte superior derecha también ofrece la posibilidad de cambiar a *Values/500 latest values*.



Los valores mostrados en esta lista son sin procesar, es decir, no se aplica ningún posprocesamiento.

Note:
La cantidad total de valores mostrados está definida por el valor del parámetro *Limit for search and filter results*, configurado en *Administration* > *General* > *GUI*.

Mapas

Resumen

En la sección *Monitoring* > *Maps* puede configurar, administrar y ver mapas de red.

Cuando abra esta sección, verá el último mapa al que accedió o una lista de todos los mapas a los que tiene acceso.

Todos los mapas pueden ser públicos o privados. Los mapas públicos están disponibles para todos los usuarios, mientras que los mapas privados solo son accesibles para su propietario y los usuarios con los que se comparte el mapa.

Lista de mapas



Datos mostrados:

Column	Description
Name	Nombre del mapa. Haga clic en el nombre para ver el mapa.
Width	Se muestra el ancho del mapa.
Height	Se muestra la altura del mapa.
Actions	Hay dos acciones disponibles: **Properties** - establecer las properties generales del mapa **Edit** - acceder a la cuadrícula para añadir map elements

Para configurar un nuevo mapa, haga clic en el botón *Create map* en la esquina superior derecha. Para importar un mapa desde un archivo YAML, XML o JSON, haga clic en el botón *Import* en la esquina superior derecha. El usuario que importe el mapa quedará establecido como su propietario.

Dos botones debajo de la lista ofrecen algunas opciones de edición masiva:

- *Export* - exportar los mapas a un archivo YAML, XML o JSON
- *Delete* - eliminar los mapas

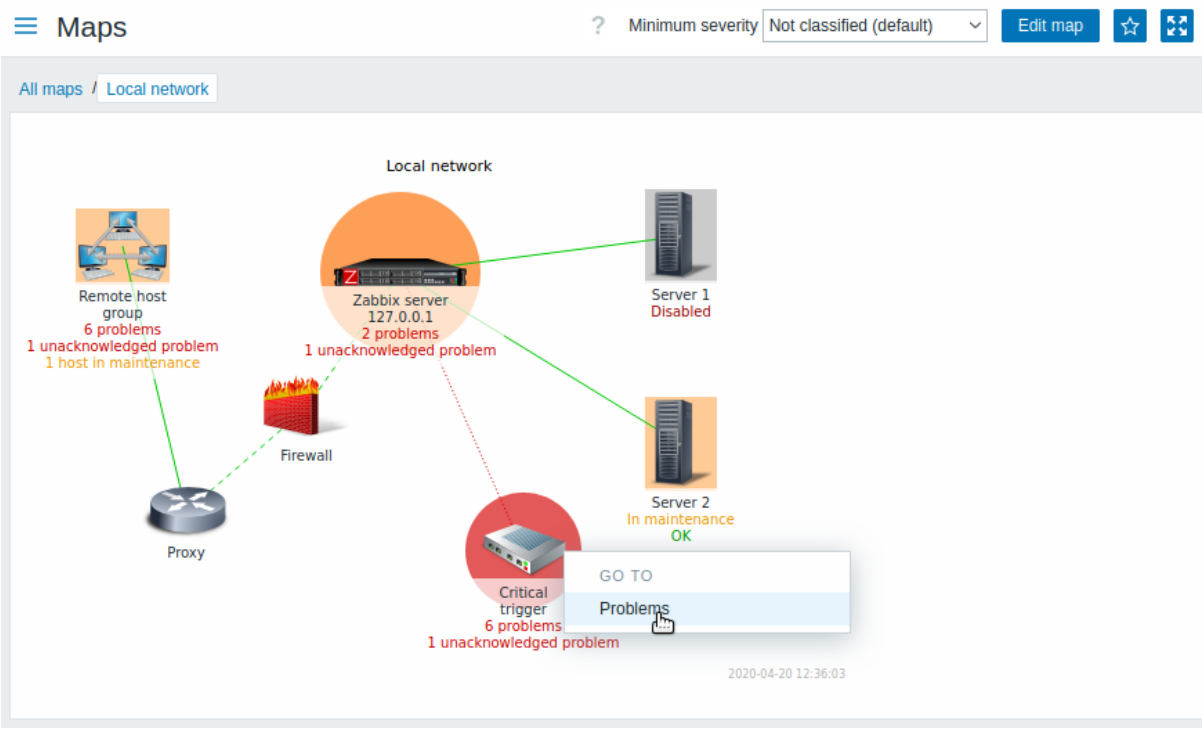
Para usar estas opciones, marque las casillas antes de los mapas correspondientes y luego haga clic en el botón requerido.

Usando el filtro

Puede usar el filtro para mostrar solo los mapas que le interesan. Para un mejor rendimiento de búsqueda, los datos se buscan con macros sin resolver.

Visualización de mapas

Para ver un mapa, haga clic en su nombre en la lista de todos los mapas.



Puede usar el menú desplegable en la barra de título del mapa para seleccionar el nivel mínimo de severidad de los triggers problemáticos que se mostrarán. La severidad marcada como *default* es el nivel establecido en la configuración del mapa. Si el mapa contiene un submapa, al navegar al submapa se conservará la severidad del mapa de nivel superior (excepto si es *Not classified*; en este caso, no se pasará al submapa).

Resaltado de iconos

Si un elemento del mapa se encuentra en estado de problema, se resalta con un círculo. El color de relleno del círculo corresponde al color de severidad del problema. Solo los problemas con el nivel de severidad seleccionado o superior se mostrarán junto con el elemento. Si todos los problemas están reconocidos, se muestra un borde verde grueso alrededor del círculo.

Además:

- un host en mantenimiento se resalta con un cuadrado naranja relleno. Tenga en cuenta que el resaltado de mantenimiento tiene prioridad sobre el resaltado de severidad del problema si el elemento del mapa es un host.
- un host deshabilitado (no monitorizado) se resalta con un cuadrado gris relleno.

El resaltado se muestra si la casilla *Resaltado de iconos* está marcada en la configuración del mapa.

Marcadores de cambios recientes

Los triángulos rojos orientados hacia dentro alrededor de un elemento indican un cambio reciente en el estado del trigger, ocurrido en los últimos 30 minutos. Estos triángulos se muestran si la casilla *Marcar elementos al cambiar el estado del trigger* está seleccionada en la configuración del mapa.

Enlaces

Al hacer clic en un elemento del mapa se abre un menú con algunos enlaces disponibles. Al hacer clic en el nombre del host se muestra el menú del host.

Botones

Los botones de la derecha ofrecen las siguientes opciones:

Edit map	Ir a la edición del contenido del mapa.
	Añadir el mapa al widget de favoritos en Dashboards.
	El mapa está en el widget de favoritos en Dashboards. Haga clic para eliminar el mapa del widget de favoritos.

Los botones del modo de vista, comunes a todas las secciones, se describen en la página Monitoring.

Resumen legible en los mapas

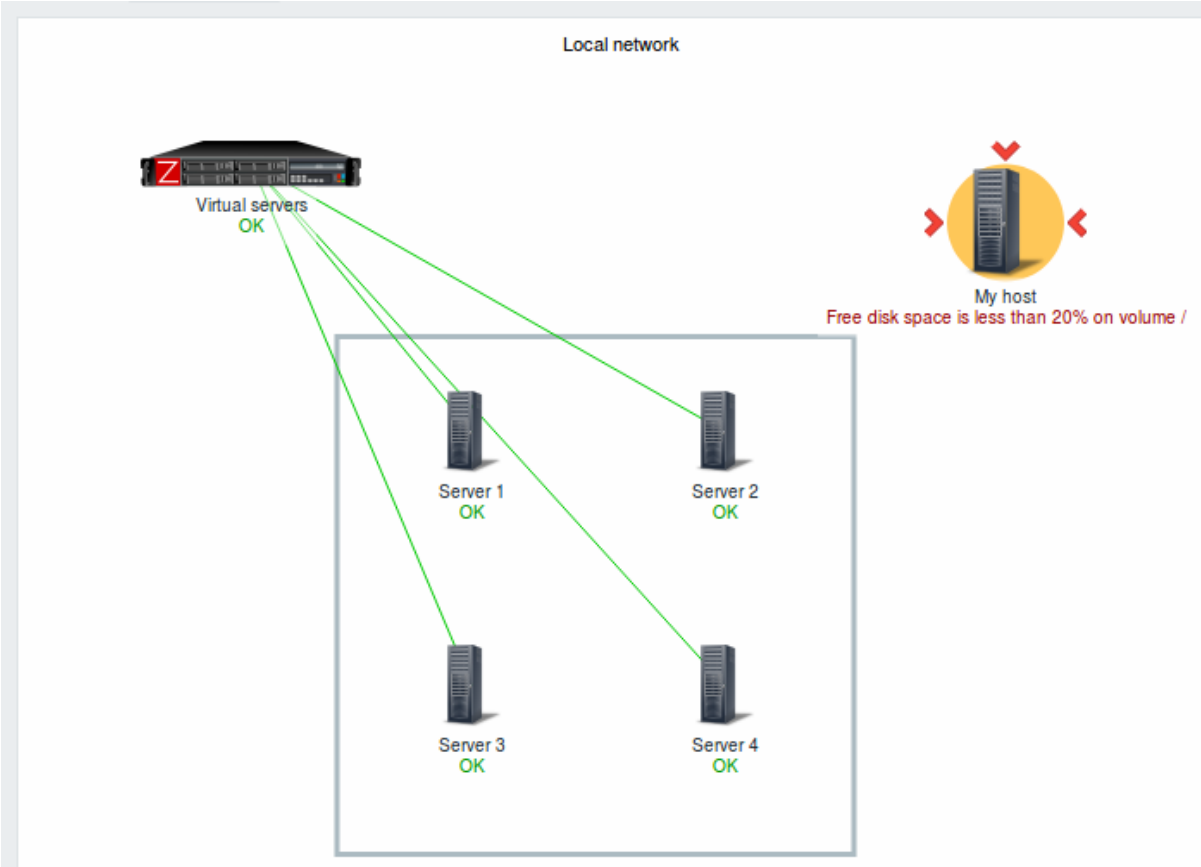
Hay disponible una propiedad oculta "aria-label" que permite leer la información del mapa con un lector de pantalla. Están disponibles tanto la descripción general del mapa como la descripción de cada elemento, en el siguiente formato:

- para la descripción del mapa: `<Nombre del mapa>, <* de * elementos en estado de problema>, <* problemas en total>.`
- para describir un elemento con un problema: `<Tipo de elemento>, Estado <estado del elemento>, <nombre del elemento>, <descripción del problema>.`
- para describir un elemento con varios problemas: `<Tipo de elemento>, Estado <estado del elemento>, <nombre del elemento>, <* problemas>.`
- para describir un elemento sin problemas: `<Tipo de elemento>, Estado <estado del elemento>, <nombre del elemento>.`

Por ejemplo, está disponible la siguiente descripción:

```
'Red local, 1 de 6 elementos en estado de problema, 1 problema en total. Host, Estado problema, Mi host, E
```

para el siguiente mapa:



Referenciando un mapa de red

Los mapas de red pueden ser referenciados tanto por los parámetros GET `sysmapid` como `mapname`. Por ejemplo,

```
http://zabbix/zabbix/zabbix.php?action=map.view&mapname=Local%20network
```

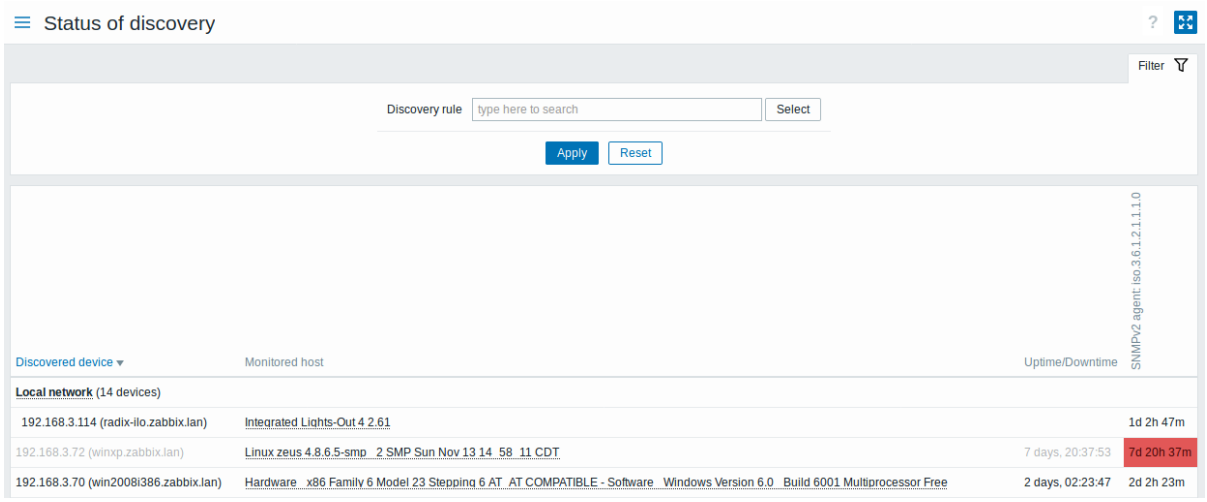
abrirá el mapa con ese nombre (Local network).

Si se especifican ambos, `sysmapid` (ID del mapa) y `mapname` (nombre del mapa), `mapname` tiene mayor prioridad.

Descubrimiento

Descripción general

En la sección *Monitorización* > *Descubrimiento* se muestran los resultados del descubrimiento de red. Los dispositivos descubiertos se ordenan según la regla de descubrimiento.



Datos mostrados:

Columna	Descripción
Dispositivo descubierto	Los dispositivos descubiertos se muestran agrupados por regla de descubrimiento. Al hacer clic en la regla de descubrimiento, se abre el menú de la regla, que contiene el enlace al formulario de configuración de la regla de descubrimiento. Si un dispositivo tiene varias direcciones IP (según lo determinado por los *Criterios de unicidad del dispositivo* de la regla de descubrimiento), se muestran juntas: primero la dirección IP principal del dispositivo, seguida de sus otras direcciones IP (con sangría). La dirección IP principal es la dirección IP más baja que tiene un host listado en la columna *Host monitorizado* o, si no existe tal dirección, la dirección IP más baja del dispositivo. Los dispositivos se pueden ordenar por su dirección IP principal en orden ascendente o descendente; las demás direcciones IP de un dispositivo siempre se muestran debajo de su dirección IP principal, en orden ascendente.
Host monitorizado	Si un dispositivo ya está siendo monitorizado, el nombre del host se muestra en esta columna. Al hacer clic en el nombre del host, se abre el menú del host.
Tiempo de actividad/inactividad	En esta columna se muestra el tiempo durante el que el dispositivo ha sido descubierto o se ha perdido después del descubrimiento anterior. Para un dispositivo con varias direcciones IP, la duración solo se muestra para su dirección IP principal.
Comprobación de descubrimiento	Se muestra el estado del servicio individual (comprobación de descubrimiento) para cada dispositivo descubierto. Una celda roja indica que el servicio está inactivo. El tiempo de actividad o inactividad del servicio se incluye en la celda. Esta columna solo se muestra si el servicio se ha encontrado en al menos un dispositivo descubierto.

Note:
El número de reglas de descubrimiento y el número total de dispositivos descubiertos que se muestran en esta sección están limitados por el parámetro *Límite para los resultados de búsqueda y filtrado* en *Administración > General > GUI*. Las reglas o los dispositivos que superen el límite no se muestran, y no se indica que esto haya ocurrido.

Botones

Los botones del modo de vista, comunes para todas las secciones, se describen en la página Monitoring.

Uso del filtro

Puede usar el filtro para mostrar solo las reglas de descubrimiento que le interesan. Para mejorar el rendimiento de la búsqueda, los datos se buscan con macros sin resolver.

Si no hay nada seleccionado en el filtro, se muestran todas las reglas de descubrimiento habilitadas. Para seleccionar una regla de descubrimiento específica para mostrarla, comience a escribir su nombre en el filtro. Se mostrarán para su selección todas las reglas de descubrimiento habilitadas que coincidan. Se puede seleccionar más de una regla de descubrimiento.

APM

Descripción general

El menú *APM* está destinado a las funciones de supervisión del rendimiento de las aplicaciones (APM) de Zabbix. Muestra los datos de telemetría que sus aplicaciones y servicios envían mediante OpenTelemetry, un estándar abierto para recopilar datos de telemetría. Estos datos le permiten ver cómo fluyen las solicitudes a través de sus aplicaciones y servicios, medir su rendimiento y encontrar la causa de las operaciones lentas o fallidas.

El menú contiene las siguientes secciones, que corresponden a los tipos de señales de OpenTelemetry:

- Trazas
- Métricas
- Registros

A diferencia de *Supervisión* > *Últimos datos*, que muestra los valores recopilados por los items de Zabbix, las secciones de *APM* muestran los datos de telemetría directamente desde el origen de datos, por lo que no es necesario crear ningún item. Para mostrar datos en las secciones de *APM*:

1. Especifique un origen de datos de telemetría en *Administración* > *Origen de datos*.
2. Asegúrese de que los datos de telemetría se escriban en el origen de datos mediante un proxy de Zabbix con APM habilitado, o mediante un recopilador de OpenTelemetry. Tenga en cuenta que el server de Zabbix no puede recibir datos de telemetría; solo puede hacerlo el proxy de Zabbix.

Para utilizar datos de telemetría en triggers o gráficos, utilice el item de consulta de telemetría.

Trazas

Descripción general

La sección *APM* > *Traces* muestra los traces recopilados de sus aplicaciones y servicios.

Un **trace** representa la ruta que sigue una solicitud a través de sus aplicaciones y servicios. Por ejemplo, cuando un usuario hace clic en el botón *Realizar pedido* de una tienda en línea, la solicitud puede pasar del servicio frontend a los servicios de pago y de procesamiento del pedido, y finalmente a la base de datos.

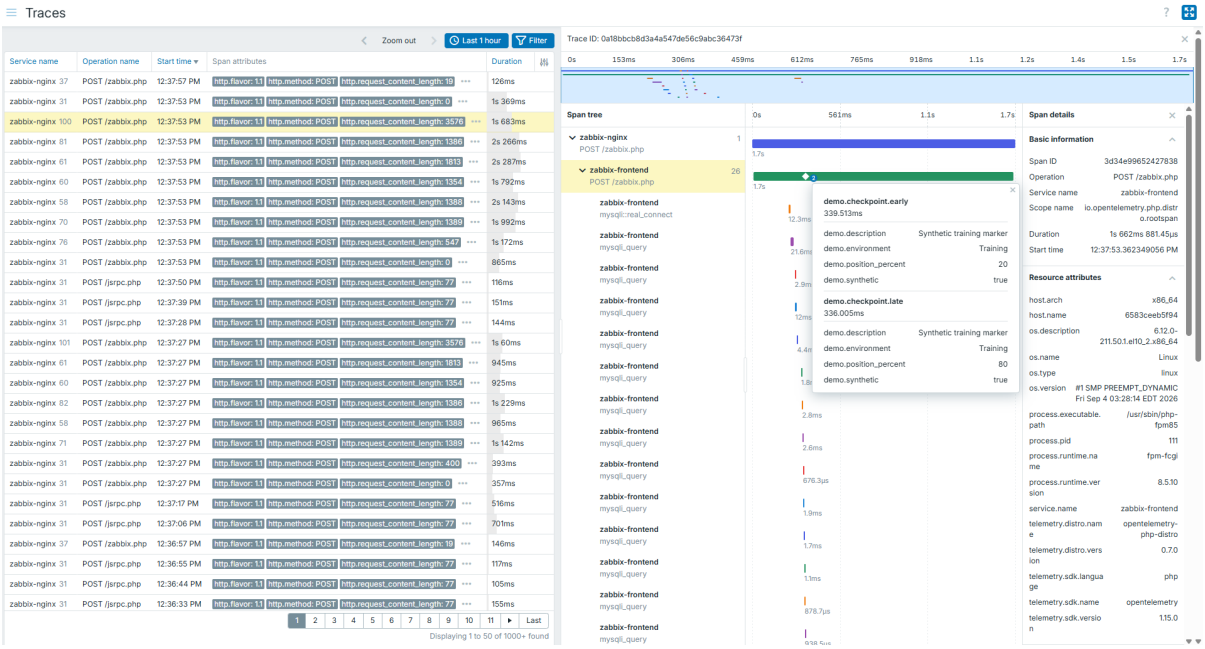
Un trace consta de **spans**. Cada span representa una única operación, como una solicitud HTTP que el servicio frontend envía al servicio de procesamiento del pedido, una consulta a la base de datos que guarda el pedido o una llamada a una función que calcula el costo de envío.

Los spans forman una jerarquía. El **span raíz** representa la primera operación de la solicitud. Cada uno de los demás spans es un **span secundario** del span durante el cual se inició su operación. Por ejemplo, el servicio frontend que recibe la solicitud *Realizar pedido* es el span raíz, y el servicio de procesamiento del pedido es su span secundario.

Cada span es producido por un **servicio** (por ejemplo, `zabbix-frontend`; identificado mediante el atributo de recurso `service.name` de OpenTelemetry) y por un ámbito de instrumentación, es decir, la parte del código que lo creó (por ejemplo, una biblioteca que realiza el seguimiento de solicitudes HTTP).

Para proporcionar contexto sobre cada operación, los spans también tienen **atributos** clave-valor: los atributos de span describen la operación en sí (por ejemplo, `http.method: POST`), mientras que los atributos de recurso describen el servicio y el entorno en el que se ejecuta (por ejemplo, `host.name: eabf8c2a5ddc`).

Para obtener más información sobre los traces, consulte la documentación de OpenTelemetry.



La sección *APM* > *Traces* contiene los siguientes elementos:

- Botón Modo quiosco: muestra únicamente el contenido de la página, sin el menú ni el encabezado.
- Selector de período de tiempo: selecciona el período de tiempo durante el cual se mostrarán los traces.
- Filtro: muestra únicamente los traces que le interesan.
- Lista de traces: permite ver los traces, descritos mediante sus spans raíz.
- Panel de trace: permite ver todos los spans del trace seleccionado en una línea de tiempo.

Lista de trazas

Cada fila de la lista de trazas representa una traza; las columnas muestran los datos de su span raíz.

Service name	Operation name	Start time ▾	Span attributes	Duration
zabbix-nginx 42	POST /zabbix.php	03:40:28 PM	http.flavor: 1.1 http.method: POST http.request_content_length: 796	2s 983ms
zabbix-nginx 45	POST /zabbix.php	03:40:28 PM	http.flavor: 1.1 http.method: POST http.request_content_length: 827	2s 694ms
zabbix-nginx 41	POST /zabbix.php	03:40:28 PM	http.flavor: 1.1 http.method: POST http.request_content_length: 789	2s 595ms
zabbix-nginx 49	POST /zabbix.php	03:40:28 PM	http.flavor: 1.1 http.method: POST http.request_content_length: 1375	2s 964ms
zabbix-nginx 46	POST /zabbix.php	03:40:28 PM	http.flavor: 1.1 http.method: POST http.request_content_length: 1359	3s 75ms
zabbix-nginx 46	POST /zabbix.php	03:40:28 PM	http.flavor: 1.1 http.method: POST http.request_content_length: 1368	2s 983ms
zabbix-nginx 56	POST /zabbix.php	03:40:28 PM	http.flavor: 1.1 http.method: POST http.request_content_length: 1821	3s 160ms
zabbix-nginx 41	POST /zabbix.php	03:40:28 PM	http.flavor: 1.1 http.method: POST http.request_content_length: 790	2s 777ms
zabbix-nginx 46	POST /zabbix.php	03:40:28 PM	http.flavor: 1.1 http.method: POST http.request_content_length: 803	2s 992ms
zabbix-nginx 42	POST /zabbix.php	03:40:28 PM	http.flavor: 1.1 http.method: POST http.request_content_length: 793	2s 900ms

Las trazas se muestran en una tabla personalizable. El botón *Personalizar tabla*, situado en el encabezado de la tabla, permite mostrar, ocultar o reordenar columnas, así como restablecer el diseño predeterminado. También puede cambiar el ancho de cada columna arrastrando su borde derecho; haga doble clic en el borde para restablecerlo. Todas las personalizaciones se aplican inmediatamente y se recuerdan para el usuario actual.

La lista de trazas puede tener las siguientes columnas:

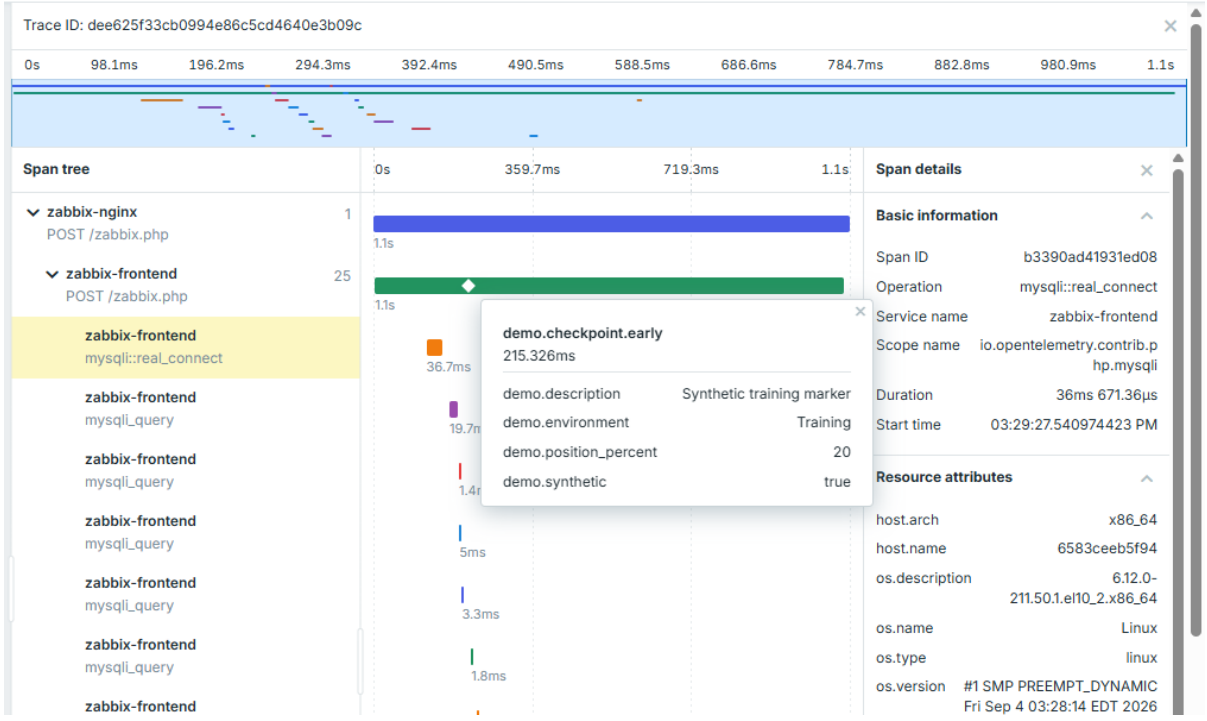
Note:
Cada columna corresponde a una columna de la tabla `otel_traces` del origen de datos de telemetría; para consultar la lista de columnas, vea Columnas de datos de telemetría compatibles. La mayoría de las columnas tienen el mismo nombre que en el origen de datos (por ejemplo, *ID de traza* corresponde a la columna `TraceId`). Cuando los nombres difieren, la columna del origen de datos se especifica en la tabla siguiente.

Columna	Descripción
ID de traza	ID de la traza, compartido por todos los spans que pertenecen a la traza.
ID de span	ID del span raíz.

Columna	Descripción
Estado de la traza	Datos adicionales, si los hay, que las herramientas de trazado adjuntan a la traza (consulte W3C Trace Context).
Nombre del servicio	Servicio que produjo el span raíz, seguido del número de spans de la traza. Si algún span tiene el código de estado *Error* (consulte más abajo), un icono rojo muestra su cantidad.
Nombre de la operación	Operación que representa el span raíz (por ejemplo, `POST /zabbix.php`). Columna del origen de datos: `SpanName`.
Tipo de span	Rol del span raíz: **No especificado**: no especificado. **Interno**: operación realizada dentro de una aplicación. **Server**: gestión de una solicitud entrante. **Cliente**: solicitud saliente. **Productor**: creación de una operación asíncrona (por ejemplo, el envío de un mensaje). **Consumidor**: procesamiento de una operación asíncrona.
Nombre del ámbito	Ámbito de instrumentación que produjo el span raíz.
Versión del ámbito	Versión del ámbito de instrumentación.
Código de estado	Estado de la operación que representa el span raíz: **No establecido**: no se ha asignado ningún estado explícito al span raíz. **Correcto**: la operación se completó correctamente. **Error**: la operación se completó con un error.
Mensaje de estado	Información adicional, si la hay, sobre el estado.
Hora de inicio	Fecha y hora en que se inició el span raíz. Columna del origen de datos: `Timestamp`.
Atributos del span	Atributos del span raíz. Haga clic en el botón de tres puntos ••• para mostrar todos los atributos.
Atributos del recurso	Atributos del recurso del span raíz. Haga clic en el botón de tres puntos ••• para mostrar todos los atributos.
Duración	Duración de la traza; la barra gris la muestra en relación con las demás trazas.

Panel de trazas

Haga clic en una traza de la lista de trazas para abrir el panel de trazas, que muestra todos los spans de la traza en una línea de tiempo: qué operaciones se realizaron, en qué orden y cuánto tiempo tomó cada una.

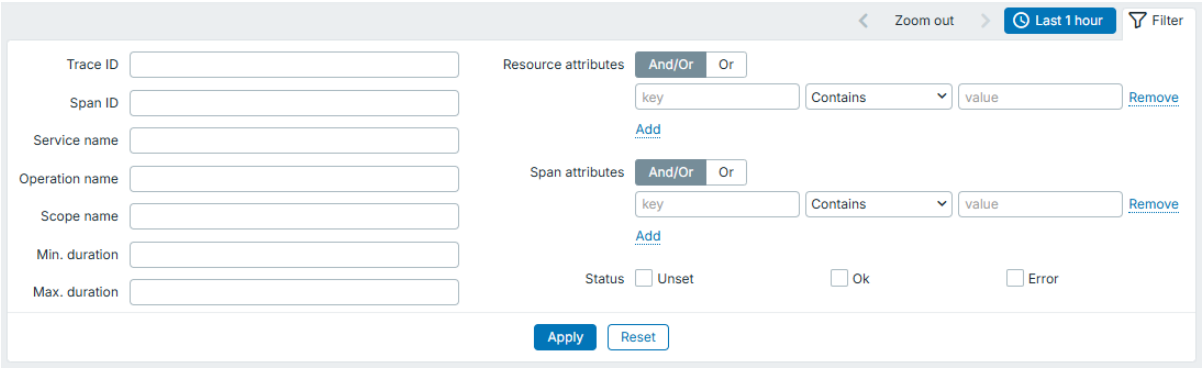


El panel de trazas puede contener los siguientes elementos:

Elemento	Descripción
ID de la traza	ID de la traza seleccionada.
Descripción general de la línea de tiempo de la traza	La traza completa, con los spans distribuidos en hasta 8 filas (varios spans pueden compartir una fila). Arrastre los bordes del rango seleccionado, o el propio rango, para ampliar la línea de tiempo de los spans en esa parte de la traza.
Árbol de spans	Spans organizados jerárquicamente; cada uno muestra su servicio (en negrita), el nombre de la operación y el número de spans secundarios directos.
Línea de tiempo de spans	Spans representados como barras: la posición y la longitud muestran cuándo comenzó cada span y cuánto tiempo tomó. Los marcadores muestran eventos (por ejemplo, excepciones); pase el cursor sobre un marcador o haga clic en él para ver los detalles.
Detalles del span	Haga clic en un span del árbol de spans para ver sus detalles y atributos.

Uso del filtro

Puede utilizar el filtro para mostrar únicamente los traces que le interesan.



Para filtrar por atributos, introduzca la clave del atributo, seleccione el operador e introduzca el valor del atributo.

Se admiten los siguientes operadores:

- **Contiene** - *(predeterminado)* el valor del atributo contiene la cadena introducida.
- **Es igual a** - el valor del atributo es igual a la cadena introducida.
- **No contiene** - el valor del atributo no contiene la cadena introducida.
- **No es igual a** - el valor del atributo no es igual a la cadena introducida.
- **Existe** - existe el atributo con la clave introducida.
- **No existe** - no existe el atributo con la clave introducida.

Si añade varias condiciones, seleccione cómo evaluarlas:

- **Y/O** - *(predeterminado)* deben cumplirse todas las condiciones; las condiciones con la misma clave de atributo se combinan con *O*.
- **O** - debe cumplirse al menos una condición.

Métricas

Descripción general

La sección *APM* > *Métricas* muestra los puntos de datos de métricas recopilados de sus aplicaciones y servicios.

Una **métrica** es una medición (por ejemplo, la duración de una solicitud, el número de pedidos procesados o el uso de memoria). Las aplicaciones envían sus valores como **puntos de datos** que contienen el valor (o los valores agregados) medido en un momento específico. Una métrica es similar a un item de Zabbix, y un punto de datos es similar a un valor de ese item.

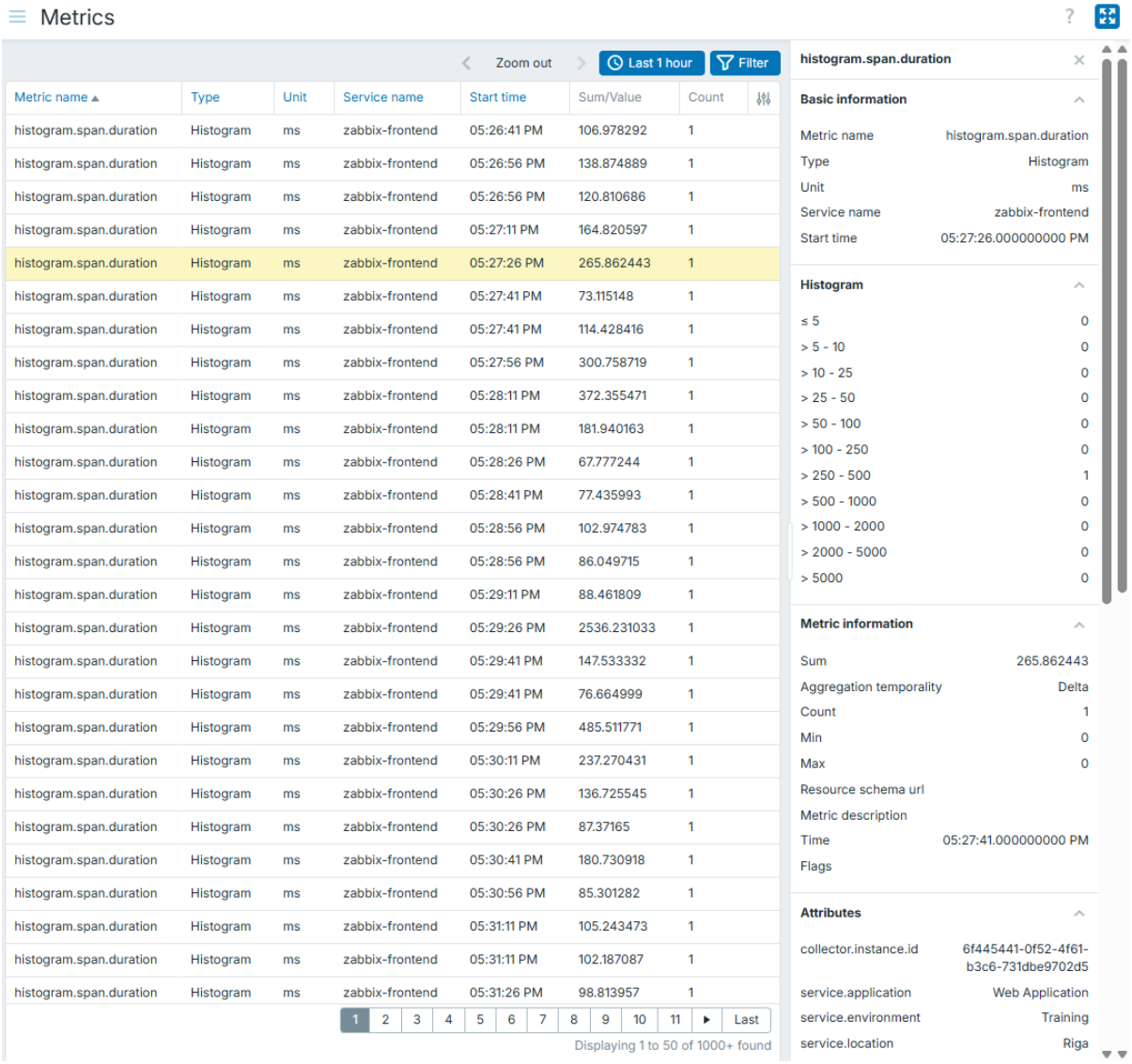
Un punto de datos puede ser de uno de los siguientes tipos:

- **Gauge**: un valor medido en un momento específico (por ejemplo, el uso actual de memoria).
- **Sum**: una suma de mediciones durante un período de tiempo (por ejemplo, el número total de solicitudes).
- **Histogram**: el número de mediciones durante un período de tiempo que se encuentran dentro de cada uno de varios rangos de valores fijos, llamados buckets (por ejemplo, cuántas solicitudes tardaron entre 0 y 100 ms, entre 100 y 250 ms o entre 250 y 500 ms).
- **Exponential histogram**: similar a un histograma, pero cada bucket es más amplio que el anterior por el mismo factor (por ejemplo, 1-2 ms, 2-4 ms, 4-8 ms).

Cada punto de datos es producido por un **servicio** (por ejemplo, `zabbix-frontend`; cuyo nombre se define mediante el atributo de recurso `service.name` de OpenTelemetry) y por un ámbito de instrumentación, es decir, la parte del código que lo creó (por ejemplo, una biblioteca que mide la duración de las solicitudes HTTP).

Para proporcionar contexto sobre cada medición, los puntos de datos también tienen **atributos** clave-valor: los atributos de los puntos de datos describen qué se midió y distinguen los puntos de datos de una misma métrica (por ejemplo, `http.request.method: POST`), mientras que los atributos de recursos describen el servicio y el entorno en el que se ejecuta (por ejemplo, `host.name: eabf8c2a5ddc`).

Para obtener más información sobre las métricas, consulte la documentación de OpenTelemetry.

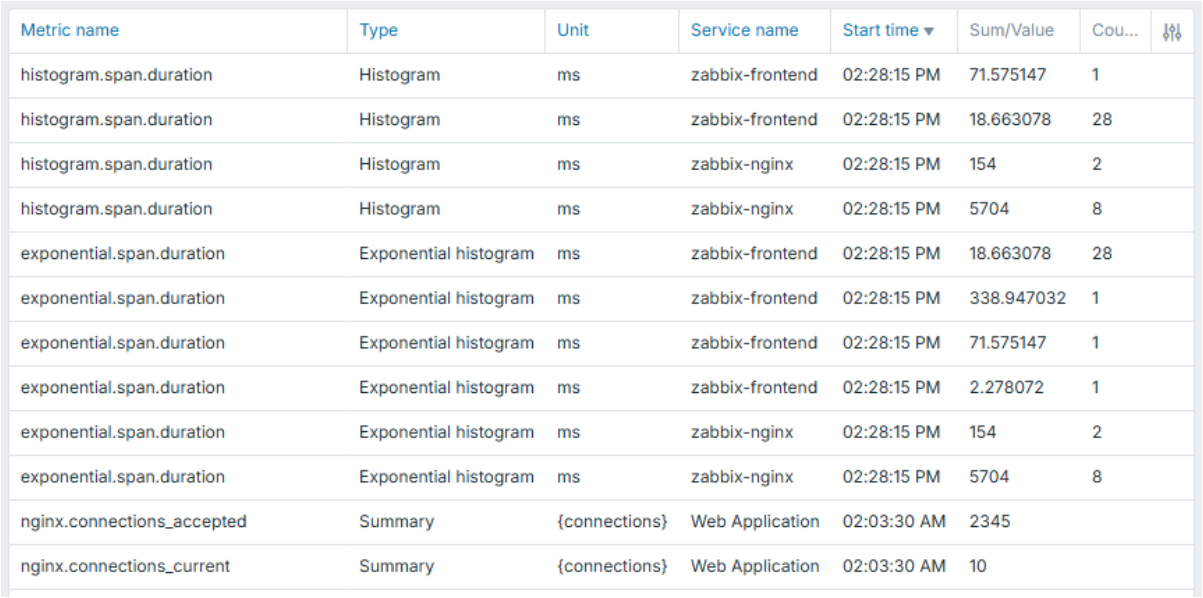


La sección *APM* > *Métricas* contiene los siguientes elementos:

- Botón Modo quiosco: muestra únicamente el contenido de la página, sin el menú ni el encabezado.
- Selector de período de tiempo: selecciona el período de tiempo para el que se mostrarán los puntos de datos.
- Filtro: muestra únicamente los puntos de datos que le interesan.
- Lista de métricas: permite consultar los puntos de datos.
- Detalles de la métrica: permite consultar los detalles del punto de datos seleccionado.

Lista de métricas

Cada fila de la lista de métricas representa un punto de datos.



Metric name	Type	Unit	Service name	Start time ▾	Sum/Value	Cou...
histogram.span.duration	Histogram	ms	zabbix-frontend	02:28:15 PM	71.575147	1
histogram.span.duration	Histogram	ms	zabbix-frontend	02:28:15 PM	18.663078	28
histogram.span.duration	Histogram	ms	zabbix-nginx	02:28:15 PM	154	2
histogram.span.duration	Histogram	ms	zabbix-nginx	02:28:15 PM	5704	8
exponential.span.duration	Exponential histogram	ms	zabbix-frontend	02:28:15 PM	18.663078	28
exponential.span.duration	Exponential histogram	ms	zabbix-frontend	02:28:15 PM	338.947032	1
exponential.span.duration	Exponential histogram	ms	zabbix-frontend	02:28:15 PM	71.575147	1
exponential.span.duration	Exponential histogram	ms	zabbix-frontend	02:28:15 PM	2.278072	1
exponential.span.duration	Exponential histogram	ms	zabbix-nginx	02:28:15 PM	154	2
exponential.span.duration	Exponential histogram	ms	zabbix-nginx	02:28:15 PM	5704	8
nginx.connections_accepted	Summary	{connections}	Web Application	02:03:30 AM	2345	
nginx.connections_current	Summary	{connections}	Web Application	02:03:30 AM	10	

Los puntos de datos se muestran en una tabla personalizable. El botón *Personalizar tabla*, situado en el encabezado de la tabla, permite mostrar, ocultar o reordenar columnas, así como restablecer el diseño predeterminado. También puede cambiar el tamaño de cada columna arrastrando su borde derecho; haga doble clic en el borde para restablecerlo. Todas las personalizaciones se aplican inmediatamente y se recuerdan para el usuario actual.

La lista de métricas puede tener las siguientes columnas:

Note:
Cada columna, excepto *Tipo*, corresponde a una columna de las tablas de métricas (`otel_metrics_gauge`, `otel_metrics_sum`, `otel_metrics_histogram` y `otel_metrics_exponential_histogram`) del origen de datos de telemetría; para consultar la lista de columnas, consulte Columnas de datos de telemetría compatibles. La mayoría de las columnas tienen el mismo nombre que en el origen de datos (por ejemplo, *Nombre de la métrica* corresponde a la columna `MetricName`). Cuando los nombres difieren, la columna del origen de datos se especifica en la tabla siguiente.

Columna	Descripción
Nombre de la métrica	Métrica a la que pertenece el punto de datos (por ejemplo, `http.server.duration`).
Tipo	Tipo de punto de datos: *Gauge*, *Sum*, *Histogram* o *Exponential histogram*. Se determina según la tabla de métricas que contiene el punto de datos.
Unidad	Unidad, si existe, de los valores de la métrica (por ejemplo, `s` o `By`; consulte UCUM). Columna del origen de datos: `MetricUnit`.
Nombre del servicio	Servicio que produjo el punto de datos.
Nombre del ámbito	Ámbito de instrumentación que produjo el punto de datos.
Hora de inicio	Fecha y hora en que comenzó el período de medición del punto de datos. Columna del origen de datos: `StartTimeUnix`.
Suma/Valor	Suma de todas las mediciones (*Histogram*, *Exponential histogram*) o el valor (*Gauge*, *Sum*) del punto de datos.
Recuento	Número de mediciones en un punto de datos *Histogram* o *Exponential histogram*.
Hora	Fecha y hora del punto de datos; para los puntos de datos *Sum*, *Histogram* y *Exponential histogram*, el final del período de medición. Columna del origen de datos: `TimeUnix`.
Indicadores	Indicadores del punto de datos: **0** - no se ha establecido ningún indicador. **1** - no se registró ningún valor para el punto de datos.
Temporalidad de agregación	Cómo se agregan a lo largo del tiempo los valores de un punto de datos *Sum*, *Histogram* o *Exponential histogram*: **No especificada** - no especificada. **Delta** - agregada durante el período transcurrido desde el punto de datos anterior. **Acumulativa** - agregada durante el período transcurrido desde la hora de inicio.
Descripción de la métrica	Descripción, si existe, de la métrica.
Atributos	Atributos del punto de datos. Haga clic en el botón de tres puntos ••• para mostrar todos los atributos.

Columna	Descripción
Atributos del recurso	Atributos del recurso del punto de datos. Haga clic en el botón de tres puntos ••• para mostrar todos los atributos.
Atributos del ámbito	Atributos del ámbito de instrumentación. Haga clic en el botón de tres puntos ••• para mostrar todos los atributos.

Detalles de la métrica

Haga clic en un punto de datos de la lista de métricas para abrir el panel de detalles de la métrica, que muestra toda la información sobre el punto de datos.

exponential.span.calls

Basic information

Metric name	exponential.span.calls
Type	Summary
Unit	
Service name	zabbix-frontend
Start time	02:22:15.000000000 PM

Metric information

Value	1
Aggregation temporality	Delta
Is monotonic	true
Resource schema url	
Metric description	
Time	02:22:30.000000000 PM
Flags	

Attributes

Resource attributes

Scope attributes

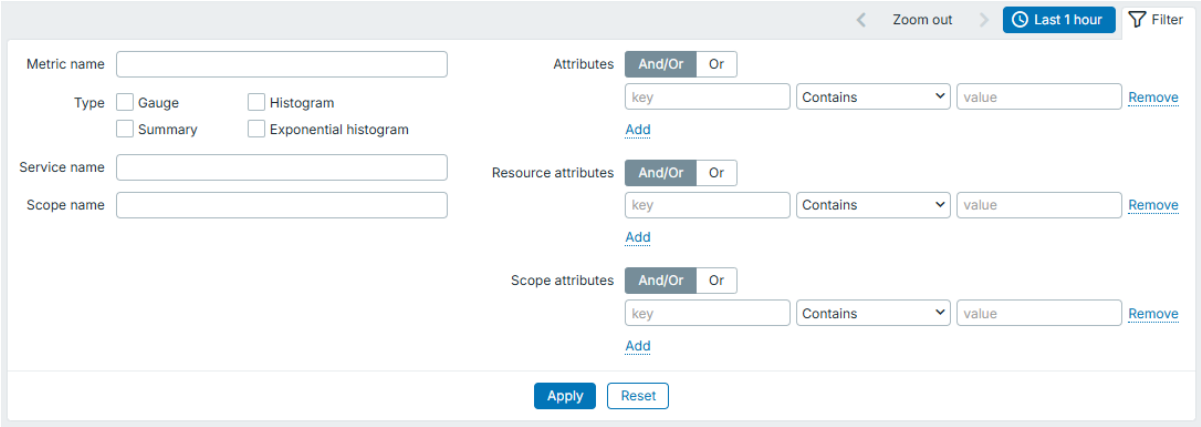
Scope

El panel de detalles de la métrica puede contener las siguientes secciones:

Sección	Descripción
Información básica	Nombre de la métrica, tipo, unidad, nombre del servicio y hora de inicio.
Histograma	Buckets (por ejemplo, `> 5 – 10`), cada uno con el número de mediciones que contiene. Se muestra únicamente para puntos de datos de tipo *Histograma*.
Histograma exponencial	Buckets (por ejemplo, `>= 249.7885 – 249.7887`), cada uno con el número de mediciones que contiene. Se muestra únicamente para puntos de datos de tipo *Histograma exponencial*.
Información de la métrica	Valores del punto de datos, que dependen de su tipo, así como la descripción de la métrica, la hora, los indicadores y la URL del esquema del recurso.
Atributos	Atributos del punto de datos.
Atributos del recurso	Atributos del recurso.
Atributos del ámbito	Atributos, si los hay, del ámbito de instrumentación.
Ámbito	Nombre, versión y URL del esquema del ámbito de instrumentación.

Uso del filtro

Puede utilizar el filtro para mostrar únicamente los puntos de datos que le interesan.



Para filtrar por atributos, introduzca la clave del atributo, seleccione el operador e introduzca el valor del atributo.

Se admiten los siguientes operadores:

- **Contiene** - *(predeterminado)* el valor del atributo contiene la cadena introducida.
- **Es igual a** - el valor del atributo es igual a la cadena introducida.
- **No contiene** - el valor del atributo no contiene la cadena introducida.
- **No es igual a** - el valor del atributo no es igual a la cadena introducida.
- **Existe** - existe el atributo con la clave introducida.
- **No existe** - no existe el atributo con la clave introducida.

Si añade varias condiciones, seleccione cómo evaluarlas:

- **Y/O** - *(predeterminado)* deben cumplirse todas las condiciones; las condiciones con la misma clave de atributo se combinan con *O*.
- **O** - debe cumplirse al menos una condición.

Registros

Descripción general

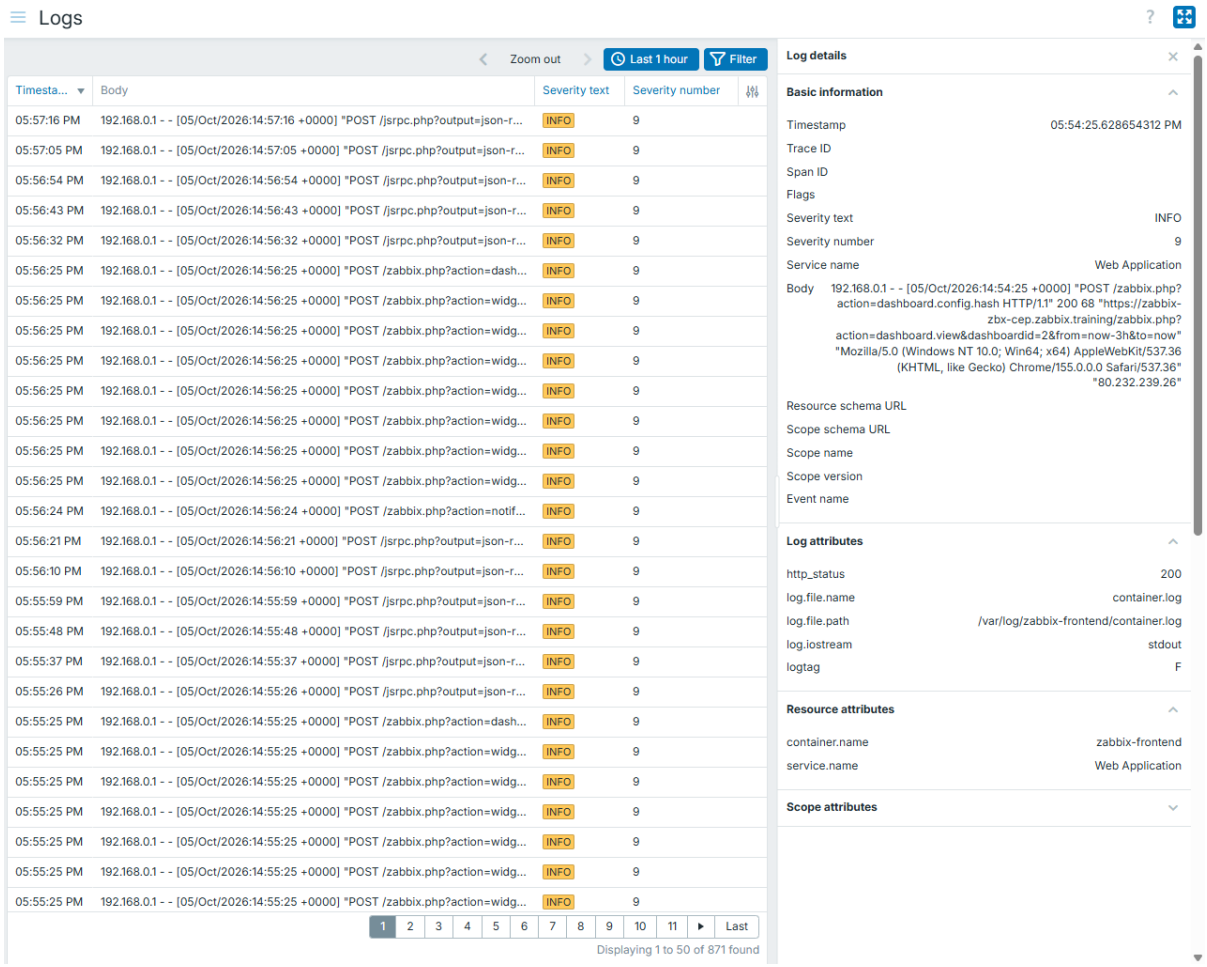
La sección *APM* > *Logs* muestra los registros de logs recopilados de sus aplicaciones y servicios.

Un **registro de log** es un registro de texto de un evento (por ejemplo, un pago fallido o un tiempo de espera agotado de la base de datos). Cada registro de log tiene una gravedad y un mensaje (cuerpo). Si un registro de log se produjo durante una operación que forma parte de una traza, también contiene los ID de esa traza y de ese span.

Cada registro de log es producido por un **servicio** (por ejemplo, `zabbix-frontend`; denominado mediante el atributo de recurso `service.name` de OpenTelemetry) y por un ámbito de instrumentación, es decir, la parte del código que lo creó (por ejemplo, una biblioteca de logging).

Para proporcionar contexto sobre cada evento, los registros de log también tienen **atributos** clave-valor: los atributos de log describen el evento en sí (por ejemplo, `log.file.path: /var/log/zabbix/zabbix_frontend.log`), mientras que los atributos de recurso describen el servicio y el entorno en el que se ejecuta (por ejemplo, `host.name: eabf8c2a5ddc`).

Para obtener más información sobre los logs, consulte la documentación de OpenTelemetry.

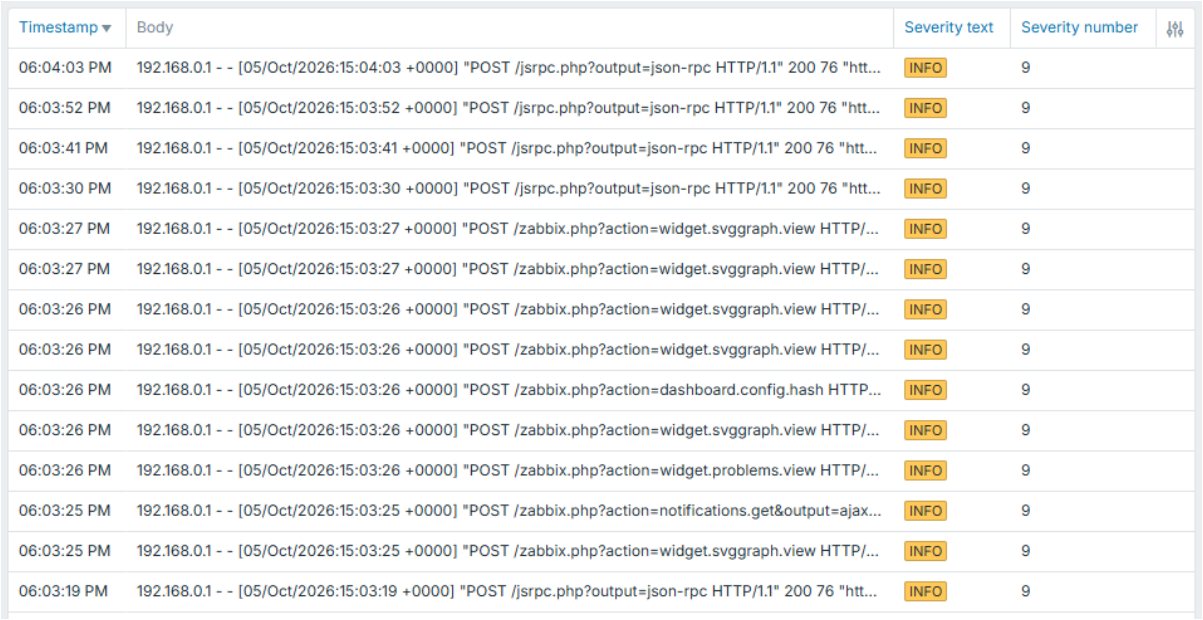


La sección *APM > Logs* contiene los siguientes elementos:

- Botón Modo kiosco: muestra únicamente el contenido de la página, sin el menú ni el encabezado.
- Selector de período de tiempo: selecciona el período de tiempo durante el cual se mostrarán los registros de log.
- Filtro: muestra únicamente los registros de log que le interesan.
- Lista de logs: permite ver los registros de log.
- Detalles del log: permite ver los detalles del registro de log seleccionado.

Lista de registros

Cada fila de la lista de registros representa un registro de log.



Timestamp ▾	Body	Severity text	Severity number
06:04:03 PM	192.168.0.1 - - [05/Oct/2026:15:04:03 +0000] "POST /jsrpc.php?output=json-rpc HTTP/1.1" 200 76 "htt...	INFO	9
06:03:52 PM	192.168.0.1 - - [05/Oct/2026:15:03:52 +0000] "POST /jsrpc.php?output=json-rpc HTTP/1.1" 200 76 "htt...	INFO	9
06:03:41 PM	192.168.0.1 - - [05/Oct/2026:15:03:41 +0000] "POST /jsrpc.php?output=json-rpc HTTP/1.1" 200 76 "htt...	INFO	9
06:03:30 PM	192.168.0.1 - - [05/Oct/2026:15:03:30 +0000] "POST /jsrpc.php?output=json-rpc HTTP/1.1" 200 76 "htt...	INFO	9
06:03:27 PM	192.168.0.1 - - [05/Oct/2026:15:03:27 +0000] "POST /zabbix.php?action=widget.svggraph.view HTTP/...	INFO	9
06:03:27 PM	192.168.0.1 - - [05/Oct/2026:15:03:27 +0000] "POST /zabbix.php?action=widget.svggraph.view HTTP/...	INFO	9
06:03:26 PM	192.168.0.1 - - [05/Oct/2026:15:03:26 +0000] "POST /zabbix.php?action=widget.svggraph.view HTTP/...	INFO	9
06:03:26 PM	192.168.0.1 - - [05/Oct/2026:15:03:26 +0000] "POST /zabbix.php?action=widget.svggraph.view HTTP/...	INFO	9
06:03:26 PM	192.168.0.1 - - [05/Oct/2026:15:03:26 +0000] "POST /zabbix.php?action=dashboard.config.hash HTTP...	INFO	9
06:03:26 PM	192.168.0.1 - - [05/Oct/2026:15:03:26 +0000] "POST /zabbix.php?action=widget.svggraph.view HTTP/...	INFO	9
06:03:26 PM	192.168.0.1 - - [05/Oct/2026:15:03:26 +0000] "POST /zabbix.php?action=widget.problems.view HTTP/...	INFO	9
06:03:25 PM	192.168.0.1 - - [05/Oct/2026:15:03:25 +0000] "POST /zabbix.php?action=notifications.get&output=ajax...	INFO	9
06:03:25 PM	192.168.0.1 - - [05/Oct/2026:15:03:25 +0000] "POST /zabbix.php?action=widget.svggraph.view HTTP/...	INFO	9
06:03:19 PM	192.168.0.1 - - [05/Oct/2026:15:03:19 +0000] "POST /jsrpc.php?output=json-rpc HTTP/1.1" 200 76 "htt...	INFO	9

Los registros de log se muestran en una tabla personalizable. El botón *Personalizar tabla*, situado en el encabezado de la tabla, permite mostrar, ocultar o reordenar columnas, así como restablecer el diseño predeterminado. También puede cambiar el tamaño de cada columna arrastrando su borde derecho; haga doble clic en el borde para restablecerlo. Todas las personalizaciones se aplican inmediatamente y se recuerdan para el usuario actual.

La lista de registros puede tener las siguientes columnas:

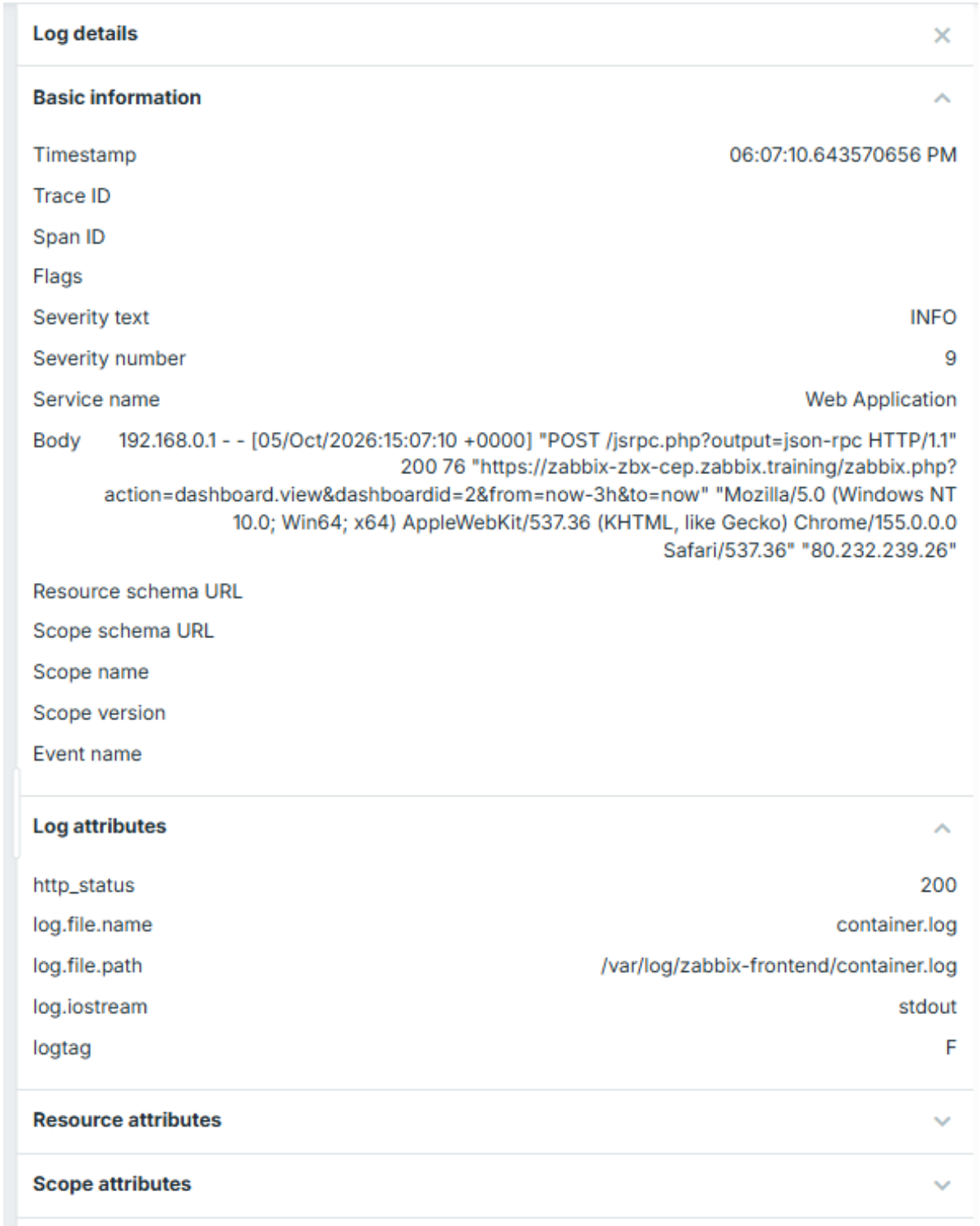
Note:
Cada columna corresponde a una columna de la tabla `otel_logs` del origen de datos de telemetría; para consultar la lista de columnas, consulte Columnas de datos de telemetría compatibles. La mayoría de las columnas tienen el mismo nombre que en el origen de datos (por ejemplo, *Texto de gravedad* corresponde a la columna `SeverityText`). Cuando los nombres difieren, la columna del origen de datos se especifica en la tabla siguiente.

Columna	Descripción
Marca de tiempo	Fecha y hora en que ocurrió el evento.
Cuerpo	Mensaje del registro de log.
Texto de gravedad	Nivel de gravedad proporcionado por la aplicación (por ejemplo, `INFO`, `WARN` o `ERROR`). El color de la etiqueta depende de la gravedad.
Número de gravedad	Nivel de gravedad como número: **0** - UNSPECIFIED. **1-4** - TRACE. **5-8** - DEBUG. **9-12** - INFO. **13-16** - WARN. **17-20** - ERROR. **21-24** - FATAL.
ID de traza	ID, si existe, de la traza durante la cual se produjo el registro de log.
ID de span	ID, si existe, del span durante el cual se produjo el registro de log.
Indicadores	Indicadores de traza del registro de log: **0** - la traza no se muestrea. **1** - la traza se muestrea (se registra). Columna del origen de datos: `TraceFlags`.
Nombre del servicio	Servicio que produjo el registro de log.
Nombre del ámbito	Ámbito de instrumentación que produjo el registro de log.
Versión del ámbito	Versión del ámbito de instrumentación.
Nombre del evento	Nombre, si existe, del tipo de evento que representa el registro de log.
Atributos del log	Atributos de log del registro de log. Haga clic en el botón de tres puntos ••• para mostrar todos los atributos.
Atributos del recurso	Atributos del recurso del registro de log. Haga clic en el botón de tres puntos ••• para mostrar todos los atributos.

Columna	Descripción
Atributos del ámbito	Atributos del ámbito de instrumentación. Haga clic en el botón de tres puntos ••• para mostrar todos los atributos.

Detalles del registro

Haga clic en un registro en la lista de registros para abrir el panel *Detalles del registro*, que muestra toda la información del registro.



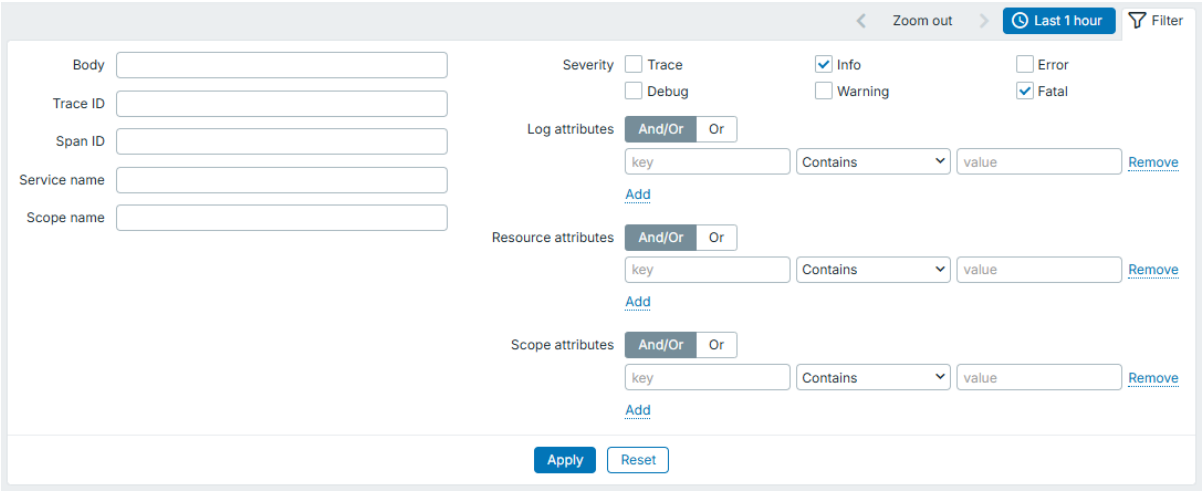
El panel *Detalles del registro* puede tener las siguientes secciones:

Sección	Descripción
Información básica	ID de traza, ID de span y flags de traza del registro.
Atributos del registro	Atributos del registro.
Atributos del recurso	URL del esquema y atributos del recurso.

Sección	Descripción
Atributos del ámbito	URL del esquema, nombre, versión y atributos del ámbito de instrumentación.

Uso del filtro

Puede utilizar el filtro para mostrar únicamente los registros de log que le interesen.



Para filtrar por atributos, introduzca la clave del atributo, seleccione el operador e introduzca el valor del atributo.

Se admiten los siguientes operadores:

- **Contiene** - *(predeterminado)* el valor del atributo contiene la cadena introducida.
- **Es igual a** - el valor del atributo es igual a la cadena introducida.
- **No contiene** - el valor del atributo no contiene la cadena introducida.
- **No es igual a** - el valor del atributo no es igual a la cadena introducida.
- **Existe** - existe el atributo con la clave introducida.
- **No existe** - no existe el atributo con la clave introducida.

Si añade varias condiciones, seleccione cómo evaluarlas:

- **Y/O** - *(predeterminado)* deben cumplirse todas las condiciones; las condiciones con la misma clave de atributo se combinan con *O*.
- **O** - debe cumplirse al menos una condición.

Servicios

Descripción general

El menú Servicios es para las funciones de monitorización de servicios de Zabbix.

Servicios

Descripción general

En esta sección puede consultar el estado general de todos los servicios que se han configurado en Zabbix, en función de su infraestructura.

Un servicio puede ser una jerarquía compuesta por varios niveles de otros servicios, denominados servicios "secundarios", que influyen en el estado general del servicio (consulte también una descripción general de la funcionalidad de monitorización de servicios).

Las principales categorías del estado de un servicio son *Correcto* o *Problema*, donde el estado *Problema* se expresa mediante el nombre y el color de la gravedad del problema correspondiente.

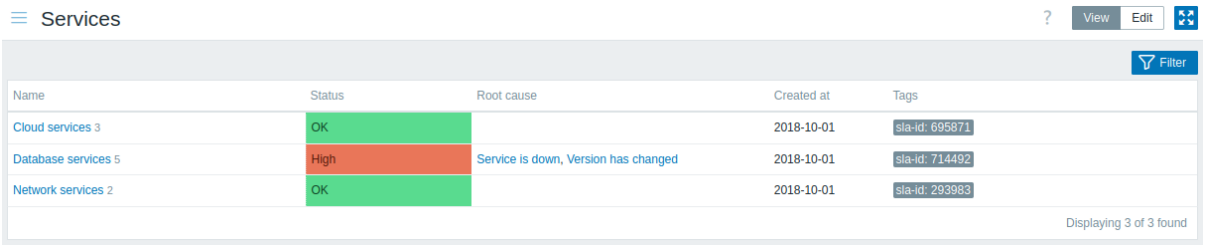
Aunque el modo de visualización permite monitorizar los servicios junto con su estado y otros detalles, también puede configurar la jerarquía de servicios en esta sección (añadir o editar servicios y servicios secundarios) cambiando al modo de edición.

Para cambiar del modo de visualización al modo de edición (y viceversa), haga clic en el botón correspondiente situado en la esquina superior derecha:

- View - ver servicios
- Edit - añadir o editar servicios y servicios secundarios

Tenga en cuenta que el acceso a la edición depende de la configuración del rol de usuario.

Visualización de servicios



Se muestra una lista de los servicios existentes.

Datos mostrados:

Parameter	Description
Name	Nombre del servicio. El nombre del servicio es un enlace a detalles del servicio. El número después del nombre indica cuántos servicios secundarios tiene el servicio.
Status	Estado del servicio: **OK** - no hay problemas **<problem color and severity>** - indica un problema y su gravedad. En caso de varios problemas, se muestran el color y la gravedad del problema más crítico.
Root cause	Se enumeran los problemas subyacentes que afectan directa o indirectamente al estado del servicio. Los mismos problemas se enumeran tal como los devuelve la macro {SERVICE.ROOTCAUSE}. Haga clic en el nombre del problema para ver más detalles sobre él en *Monitoring* > *Problems*. Los problemas que no afectan al estado del servicio no aparecen en la lista.
Created at	Se muestra la hora en que se creó el servicio.
Tags	Se muestran las etiquetas del servicio. Las etiquetas se usan para identificar un servicio en acciones y SLAs del servicio.

Botones

Los botones del modo de vista, comunes a todas las secciones, se describen en la página Monitoring.

Uso del filtro

Puede utilizar el filtro para mostrar solo los servicios que le interesan.

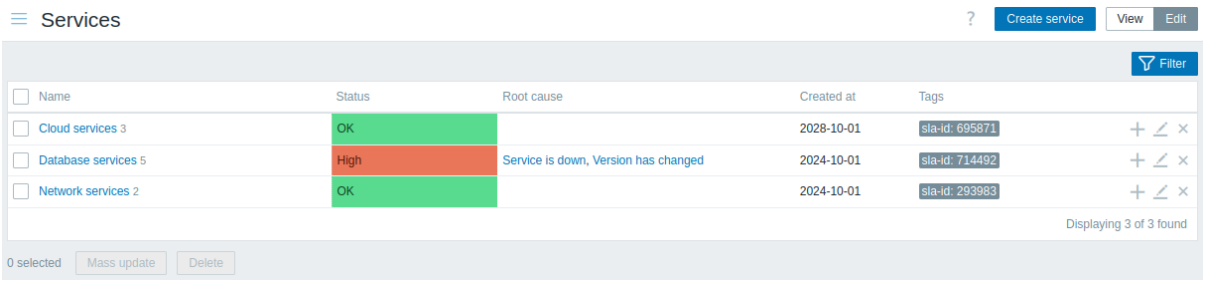
Parámetro	Descripción
Nombre	Filtrar por nombre de servicio.
Estado	Filtrar por estado del servicio.
Solo servicios sin hijos	Marque la casilla para mostrar solo los servicios sin servicios hijos. Este parámetro solo está disponible en el modo de edición de servicios.
Solo servicios sin etiquetas de problema	Marque la casilla para mostrar solo los servicios sin etiquetas de problema. Este parámetro solo está disponible en el modo de edición de servicios.

Parámetro	Descripción
Etiquetas	Filtrar por nombre y valor de la etiqueta del servicio, o por nombre y valor de la etiqueta de problema del servicio (en el modo de edición de servicios). Es posible incluir o excluir etiquetas y valores de etiquetas específicos. Se pueden establecer varias condiciones. La coincidencia de nombres de etiquetas distingue entre mayúsculas y minúsculas. Hay varios operadores disponibles para cada condición: **Existe** - incluir los nombres de etiquetas especificados; **Igual** - incluir los nombres y valores de etiquetas especificados (distingue mayúsculas y minúsculas); **Contiene** - incluir los nombres de etiquetas especificados donde los valores de las etiquetas contengan la cadena introducida (coincidencia de subcadena, no distingue mayúsculas y minúsculas); **No existe** - excluir los nombres de etiquetas especificados; **No es igual** - excluir los nombres y valores de etiquetas especificados (distingue mayúsculas y minúsculas); **No contiene** - excluir los nombres de etiquetas especificados donde los valores de las etiquetas contengan la cadena introducida (coincidencia de subcadena, no distingue mayúsculas y minúsculas). Hay dos tipos de cálculo para las condiciones: **Y/O** - se deben cumplir todas las condiciones, las condiciones con el mismo nombre de etiqueta se agruparán por la condición O; **O** - basta con que se cumpla una condición.

Edición de servicios

Haga clic en el botón *Edit* para acceder al modo de edición. Cuando está en modo de edición, la lista se complementa con casillas de verificación antes de las entradas y también con estas opciones adicionales:

-  - agregar un servicio secundario a este servicio
-  - editar este servicio
-  - eliminar este servicio

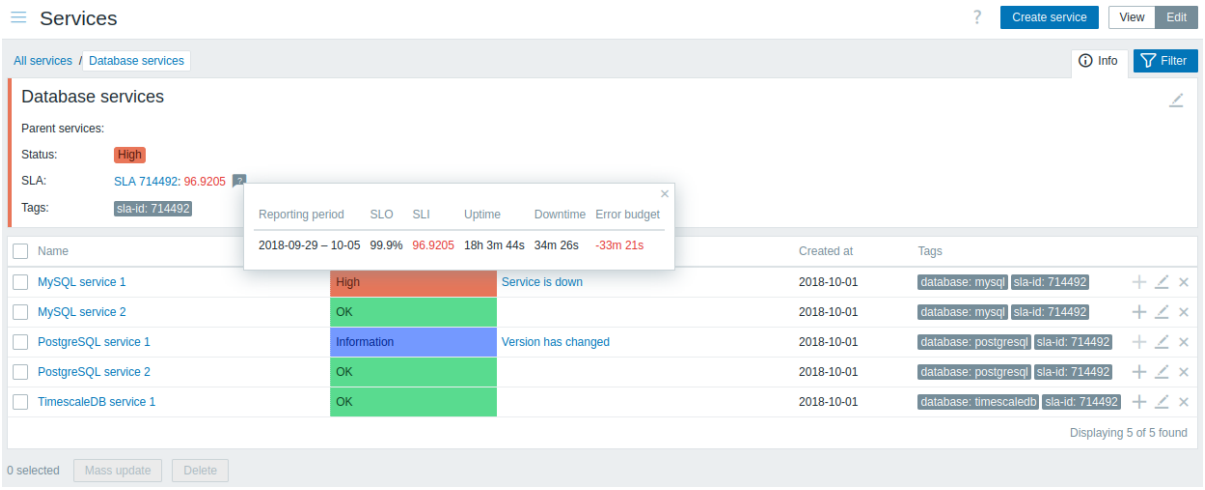


Para configurar un nuevo servicio, haga clic en el botón *Create service* en la esquina superior derecha.

Detalles del servicio

Para acceder a los detalles del servicio, haga clic en el nombre del servicio. Para volver a la lista de todos los servicios, haga clic en *All services*.

Los detalles del servicio incluyen el cuadro de información y la lista de servicios secundarios.



Para acceder al cuadro de información, haga clic en la pestaña *Info*. El cuadro de información contiene las siguientes entradas:

- Nombres de los servicios padre (si los hay)
- Estado actual de este servicio
- SLA actuales de este servicio, en el formato `SLA name:service level indicator`. 'SLA name' también es un enlace al informe de SLA para este servicio. Si coloca el mouse sobre el cuadro de información junto al indicador de nivel de servicio (SLI), se mostrará una lista emergente con los detalles del SLI. El indicador de nivel de servicio muestra el nivel actual del servicio, en porcentaje.
- Etiquetas del servicio

El cuadro de información también contiene un enlace a la configuración del servicio.

Para usar el filtro de servicios secundarios, haga clic en la pestaña *Filter*.

Cuando está en modo de edición, la lista de servicios secundarios se complementa con opciones adicionales de edición:

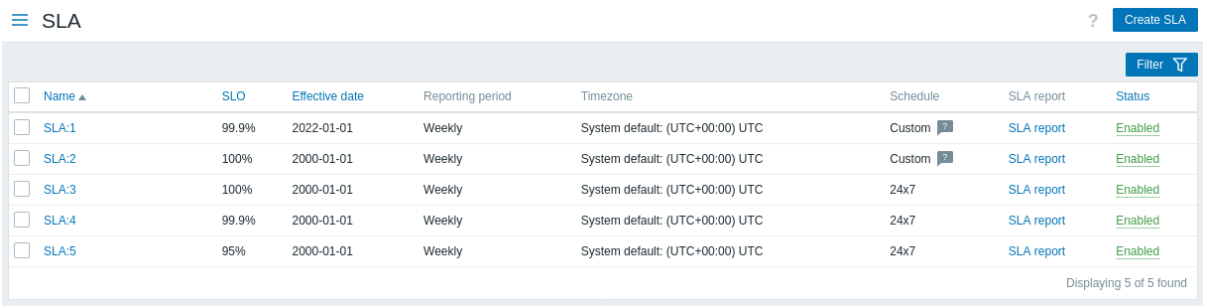
-  - agregar un servicio secundario a este servicio
-  - editar este servicio
-  - eliminar este servicio

SLA

Descripción general

Esta sección permite ver y configurar los SLA.

SLAs



Se muestra una lista de los SLAs configurados. *Nota* que solo se mostrarán los SLAs relacionados con servicios accesibles para el usuario (como solo lectura, a menos que *Manage SLA* esté habilitado para el rol de usuario).

Datos mostrados:

Parameter	Description
Name	Se muestra el nombre del SLA. El nombre es un enlace a SLA configuration.
SLO	Se muestra el objetivo de nivel de servicio (SLO).
Effective date	Se muestra la fecha de inicio del cálculo del SLA.

Parameter	Description
Reporting period	Se muestra el período utilizado en el informe del SLA: *daily*, *weekly*, *monthly*, *quarterly* o *annually*.
Time zone	Se muestra la zona horaria del SLA.
Schedule	Se muestra la programación del SLA: 24x7 o personalizada.
SLA report	Haga clic en el enlace para ver el informe del SLA de este SLA.
Status	Se muestra el estado del SLA: habilitado o deshabilitado.

Informe de SLA

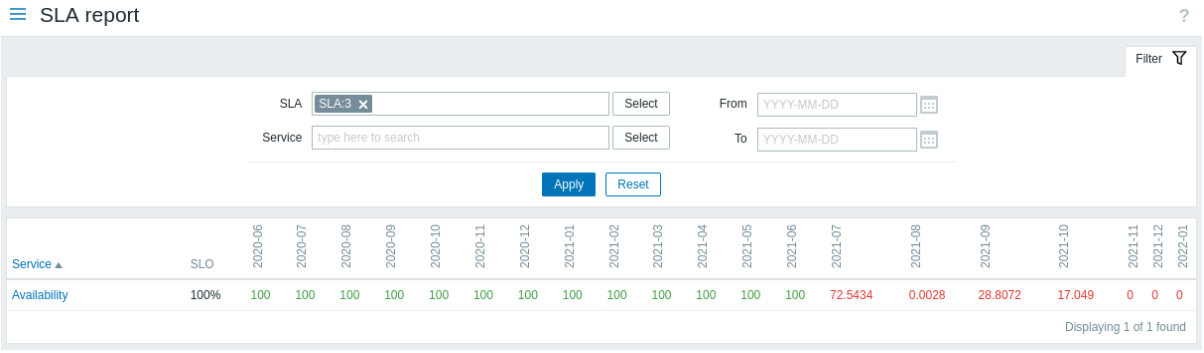
Descripción general

Esta sección permite consultar informes de SLA según los criterios seleccionados en el filtro.

Los informes de SLA también se pueden mostrar como un widget del dashboard.

Informe

El filtro permite seleccionar el informe en función del nombre del SLA y del nombre del servicio. También es posible limitar el período mostrado.

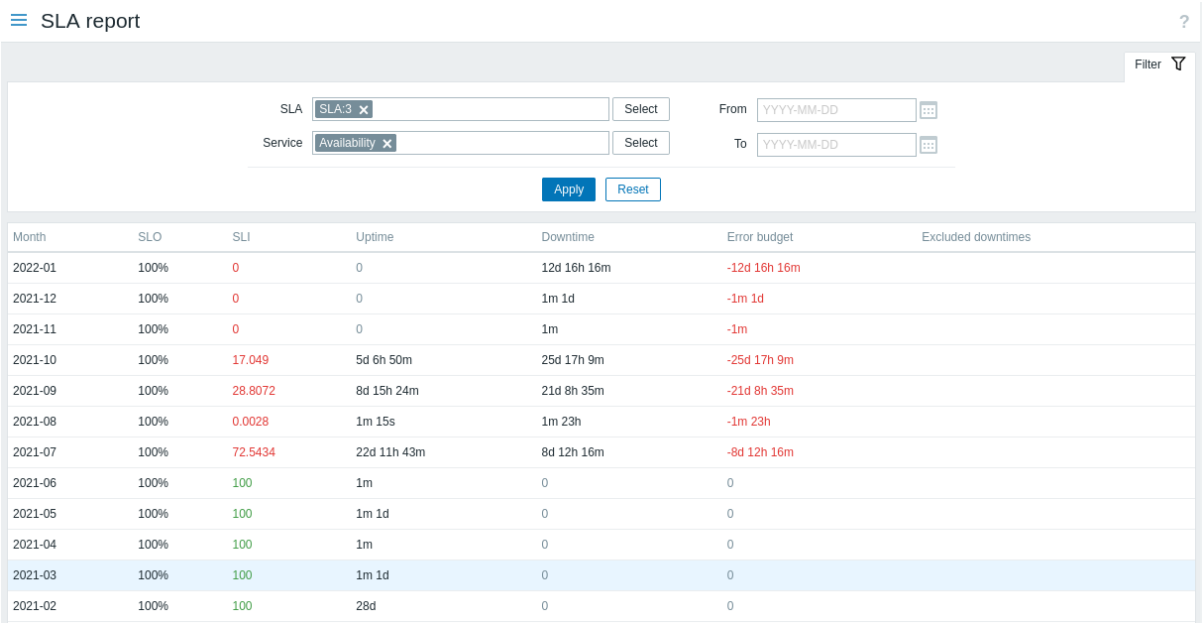


Cada columna (período) muestra el SLI correspondiente a ese período. Los SLI que incumplen el SLO establecido se resaltan en rojo.

En el informe se muestran 20 períodos. Se puede mostrar un máximo de 100 períodos si se especifican tanto la fecha *Desde* como la fecha *Hasta*.

Detalles del informe

Si hace clic en el nombre del servicio en el informe, puede acceder a otro informe que muestra una vista más detallada.



Tenga en cuenta que la duración negativa del problema no afecta al cálculo ni al informe del SLA.

Inventario

Descripción general

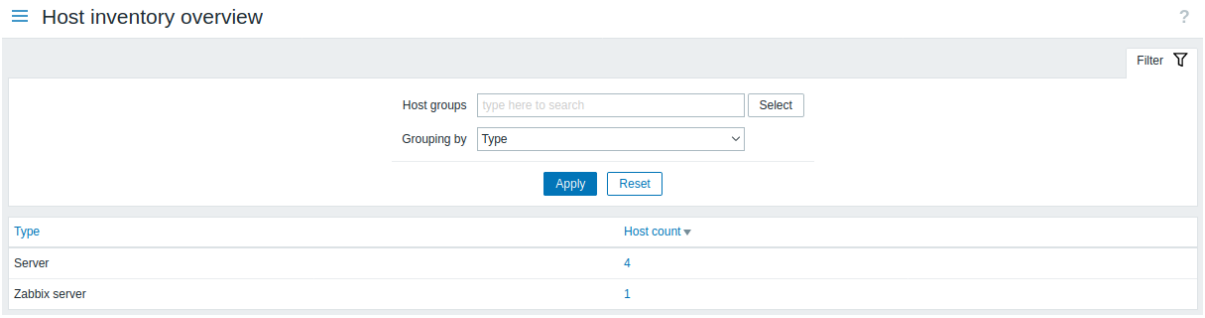
El menú Inventario incluye secciones que proporcionan una visión general de los datos de inventario de equipos por un parámetro elegido, así como la posibilidad de ver los detalles del inventario de equipos.

Descripción general

Resumen

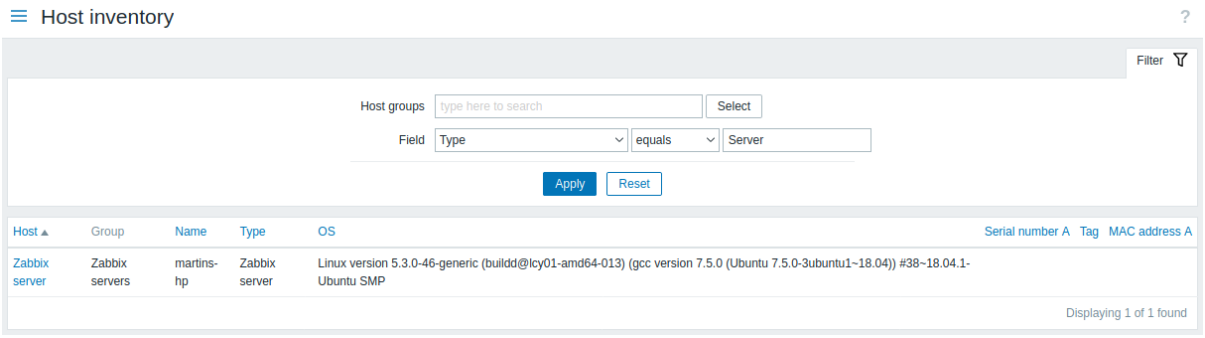
La sección *Inventory* > *Overview* proporciona formas de obtener una vista general de los datos de inventario de host.

Para mostrar un resumen, elija grupos de host (o ninguno) y el campo de inventario por el que se mostrarán los datos. Se mostrará el número de hosts correspondiente a cada entrada del campo elegido.



La completitud de un resumen depende de cuánta información de inventario se mantenga para los hosts.

Los números de la columna *Host count* son enlaces; llevan a que esos hosts se filtren en la tabla *Host Inventories*.

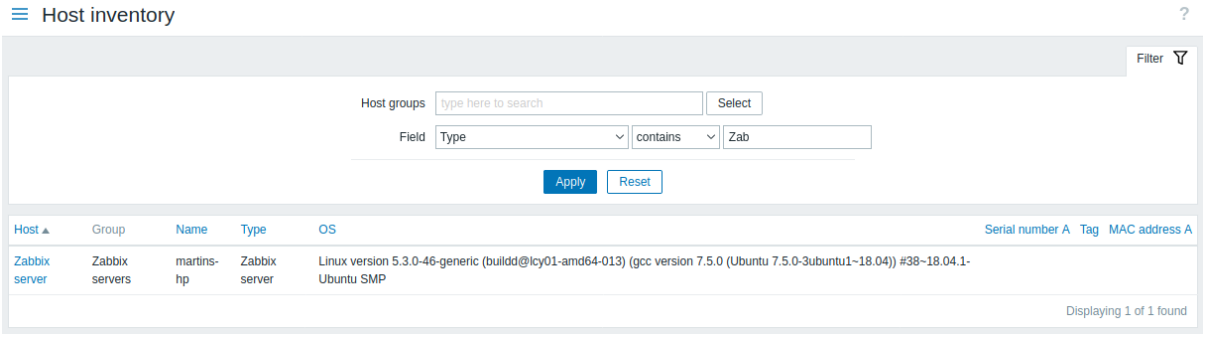


Hosts

Resumen

En la sección *Inventory* > *Hosts* inventory data se muestran los datos de inventario de los hosts.

Puede filtrar los hosts por uno o varios grupos de hosts y por cualquier campo de inventario para mostrar solo los hosts que le interesen.

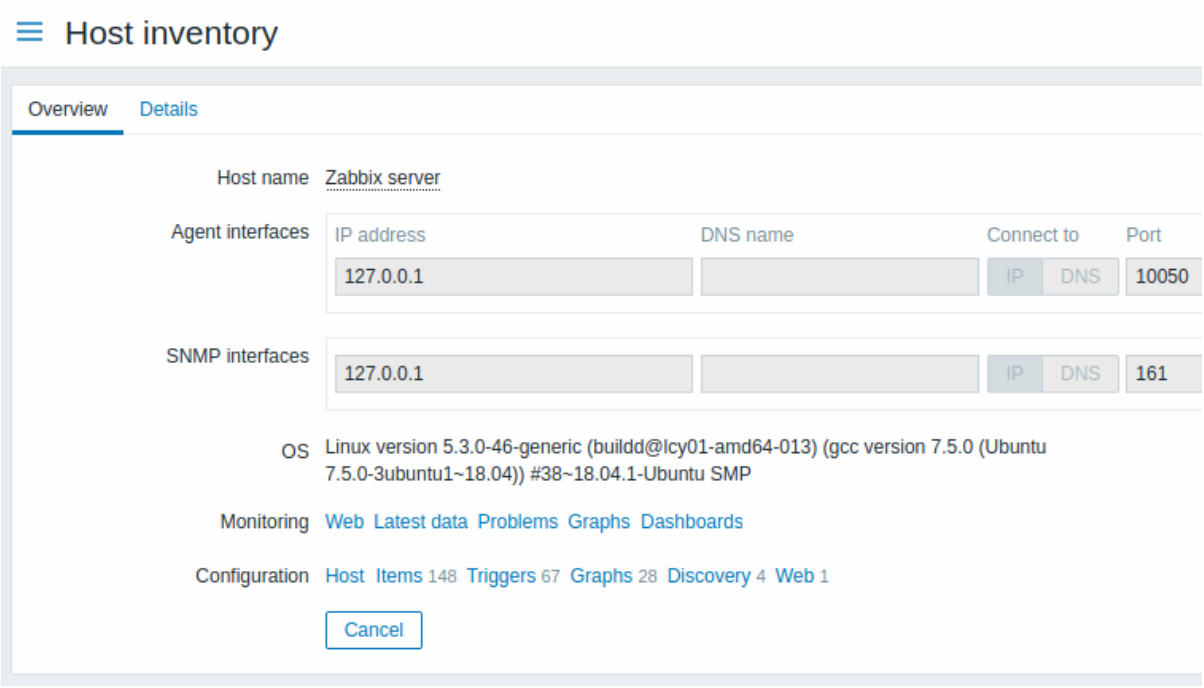


Para mostrar todos los inventarios de hosts, haga clic en el botón *Reset*.

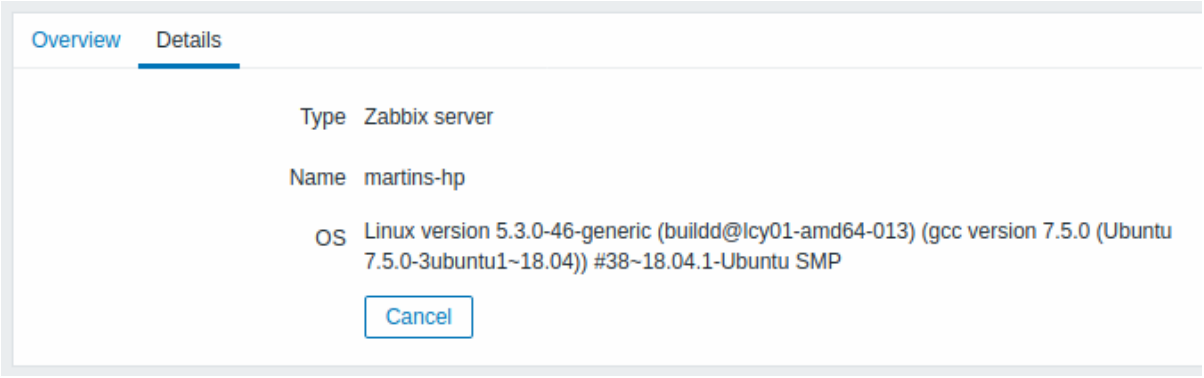
Aunque en la tabla solo se muestran algunos campos clave del inventario, también puede ver toda la información de inventario disponible para ese host. Para ello, haga clic en el nombre del host en la primera columna.

Detalles de inventario

La pestaña **Resumen** contiene información general sobre el equipo, los datos de monitorización más recientes y las opciones de configuración del equipo:



La pestaña **Detalles** contiene todos los detalles de inventario disponibles para el equipo:



La integridad de los datos de inventario depende de cuánta información de inventario se mantenga con el equipo. Si no se mantiene ninguna información, la pestaña *Detalles* está deshabilitada.

Informes

Descripción general

El menú de Informes contiene varias secciones que incluyen una variedad de informes predefinidos y personalizables por el usuario, centrados en mostrar una visión general de parámetros como información del sistema, disparadores y datos recopilados.

Información del sistema

Descripción general

En *Informes* > *Información del sistema*, se muestra un resumen de los datos principales del server de Zabbix y del sistema. Los datos del sistema se recopilan mediante items internos.

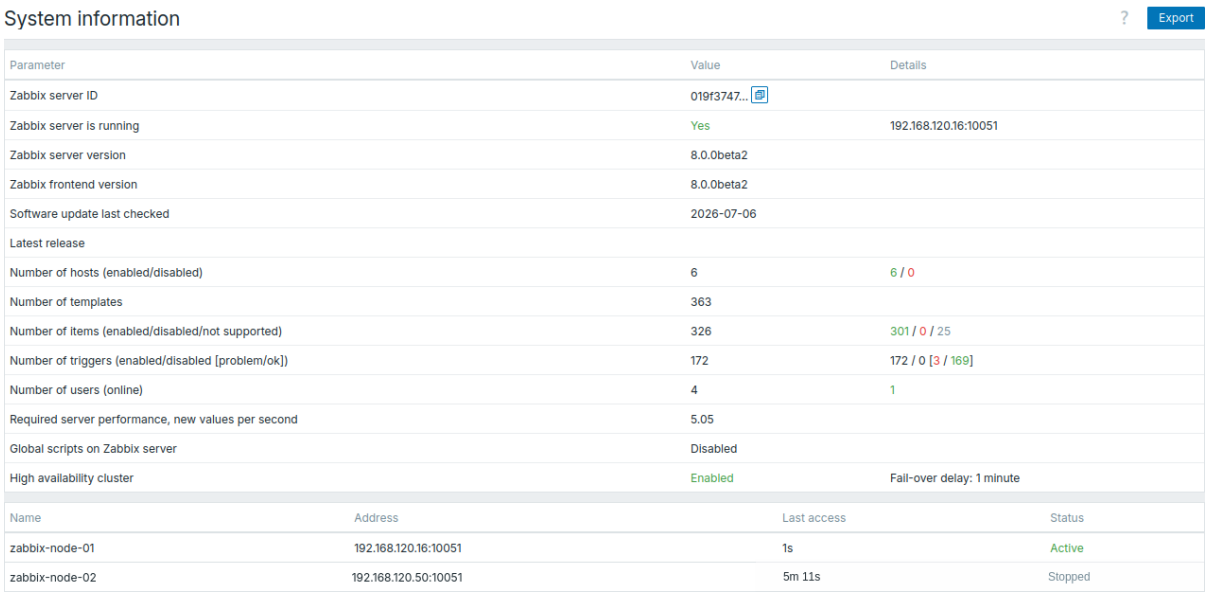
El informe de información del sistema solo está disponible para los usuarios *Super Admin* y *Admin* de Zabbix.

Tenga en cuenta que, en una configuración de alta disponibilidad, es posible redirigir el origen de la información del sistema (instancia del server). Para ello, edite el archivo de configuración del frontend: quite el comentario y establezca `$ZBX_SERVER` o `$ZBX_SERVER` y `$ZBX_SERVER_PORT` en un server distinto del que aparece como activo. Tenga en cuenta que, si solo establece `$ZBX_SERVER`, se utilizará el valor predeterminado (10051) para `$ZBX_SERVER_PORT`.

Con la configuración de alta disponibilidad habilitada, debajo de las estadísticas del sistema se muestra un bloque independiente con información detallada sobre los nodos de alta disponibilidad.

Información del sistema también está disponible como widget de dashboard.

Estadísticas del sistema



Datos mostrados:

Parámetro	Valor	Detalles
ID del server de Zabbix	Identificador único del server de Zabbix (UUIDv7).	El ID del server se genera durante el primer inicio del server. Se muestran los primeros 8 caracteres, seguidos de puntos suspensivos. Al pasar el cursor sobre un valor de ID del server o hacer clic en él, aparece una descripción emergente con el valor completo. Al hacer clic en el icono situado junto al valor, el ID se copia al portapapeles. Si la alta disponibilidad está habilitada, cada nodo tiene su propio ID de nodo para el direccionamiento mediante la línea de comandos, pero todos los nodos se agrupan bajo un único ID de server generado por el nodo que se inició primero (o que anteriormente era un server independiente), independientemente del nodo que esté activo actualmente.
El server de Zabbix está en ejecución	Estado del server de Zabbix: **Sí**: el server está en ejecución **No**: el server no está en ejecución *Nota:* Para mostrar el resto de la información, el web frontend necesita que el server esté en ejecución y debe haber al menos un proceso trapper iniciado en el server (el parámetro StartTrappers en el archivo `zabbix_server.conf` > 0).	Ubicación y puerto del server de Zabbix.

Parámetro	Valor	Detalles
Versión del server de Zabbix	Se muestra el número de versión actual del server. *Nota:* Solo se muestra cuando el server de Zabbix está en ejecución.	Se muestra el estado de la versión del server: **Actualizado**: se utiliza la versión más reciente; **Nueva actualización disponible**: hay disponible una versión más reciente; **Obsoleto**: el período de soporte completo para esta versión ha finalizado. Esta información solo está disponible si la comprobación de actualizaciones de software está habilitada en la configuración del server de Zabbix. No se muestra nada si la última comprobación de actualizaciones de software se realizó hace más de una semana o no existen datos sobre la versión actual.
Versión del frontend de Zabbix	Se muestra el número de versión del frontend de Zabbix.	Se muestra el estado de la versión del frontend de Zabbix: **Actualizado**: se utiliza la versión más reciente; **Nueva actualización disponible**: hay disponible una versión más reciente; **Obsoleto**: el período de soporte completo para esta versión ha finalizado. Esta información solo está disponible si la comprobación de actualizaciones de software está habilitada en la configuración del server de Zabbix. No se muestra nada si la última comprobación de actualizaciones de software se realizó hace más de una semana o no existen datos sobre la versión actual.
Última comprobación de actualización de software	Se muestra la fecha de la última comprobación de actualizaciones de software de Zabbix. Esta información solo está disponible si la comprobación de actualizaciones de software está habilitada en la configuración del server.	
Versión más reciente	Se muestra el número de una versión más reciente (si está disponible) para la versión actual de Zabbix. Esta información solo está disponible si la comprobación de actualizaciones de software está habilitada en la configuración del server. No se muestra nada si la última comprobación de actualizaciones de software se realizó hace más de una semana o no existen datos sobre la versión actual.	Se muestra un enlace a las notas de la versión más reciente de Zabbix disponible.
Número de hosts	Se muestra el número total de hosts configurados.	Número de hosts monitorizados/no monitorizados.
Número de templates	Se muestra el número total de templates.	
Número de items	Se muestra el número total de items.	Número de items de nivel de host monitorizados/deshabilitados/no compatibles. Los items de los hosts deshabilitados se contabilizan como deshabilitados.
Número de triggers	Se muestra el número total de triggers.	Número de triggers de nivel de host habilitados/deshabilitados; división de los triggers habilitados según los estados ”Problema”/”OK”. Los triggers enumerados en el estado ”OK” incluyen triggers con el estado ”Desconocido”. Los triggers que dependen de items deshabilitados o están asignados a hosts deshabilitados se contabilizan como deshabilitados.
Número de usuarios	Se muestra el número total de usuarios configurados.	Número de usuarios conectados.

Parámetro	Valor	Detalles
Rendimiento requerido del server, nuevos valores por segundo	Se muestra el número esperado de nuevos valores procesados por el server de Zabbix por segundo.	*El rendimiento requerido del server* es una estimación y puede resultar útil como referencia. Para obtener cifras precisas de los valores procesados, utilice el `zabbix[wcache,values,all]` item interno. En el cálculo se incluyen los items habilitados de los hosts monitorizados. Los items de registro se contabilizan como un valor por cada intervalo de actualización del item. Se contabilizan los valores de intervalos regulares; no se contabilizan los valores de intervalos flexibles ni programados. El cálculo no se ajusta durante un período de mantenimiento de "nodata". Los items trapper no se contabilizan.
Scripts globales en el server de Zabbix	En este campo se mostrará **Deshabilitado** si los scripts globales están deshabilitados en el server de Zabbix mediante la configuración de EnableGlobalScripts=0 en la configuración del server.	
Clúster de alta disponibilidad	Estado del clúster de alta disponibilidad del server de Zabbix: **Deshabilitado**: server independiente **Habilitado**: existe al menos un nodo de alta disponibilidad	Si está habilitado, se muestra el retraso de conmutación por error.

La información del sistema también mostrará un mensaje de error en las siguientes condiciones:

- La base de datos utilizada no tiene el juego de caracteres o la intercalación requeridos (UTF-8).
- La versión de la base de datos es inferior o superior al rango compatible (disponible únicamente para usuarios con el tipo de *rol de superadministrador*).
- El mantenimiento de TimescaleDB está configurado incorrectamente (las tablas de historial o tendencias contienen fragmentos comprimidos, pero las opciones *Anular el período del historial del item* o *Anular el período de tendencias del item* están deshabilitadas).

Botones

El botón en la esquina superior derecha de la página ofrece la siguiente opción:

Export	Exportar la información del sistema a un archivo JSON. El archivo se guarda como `Zabbix-system-information-{time}.json`, donde `{time}` es la marca de tiempo Unix que indica cuándo se prepararon los datos en el server. Si la alta disponibilidad está habilitada, también se exportan los detalles de los nodos de alta disponibilidad.

Nodos de alta disponibilidad

Si high availability cluster está habilitado, se muestra un bloque adicional de datos que indica el estado de cada nodo de alta disponibilidad.



Name	Address	Last access	Status
node-active	192.168.1.13:10051	12s	Active
node6	192.168.1.10:10053	1h 2m 40s	Unavailable
node7	192.168.1.11:10053	3m 40s	Unavailable
node4	192.168.1.8:10052	1h 34m 29s	Stopped
node5	192.168.1.9:10053	1h 9m 51s	Stopped
node8	192.168.1.12:10051	21m 16s	Stopped
node1	192.168.1.5:10051	17s	Standby
node2	192.168.1.6:10051	16s	Standby
node3	192.168.1.7:10052	16	Standby

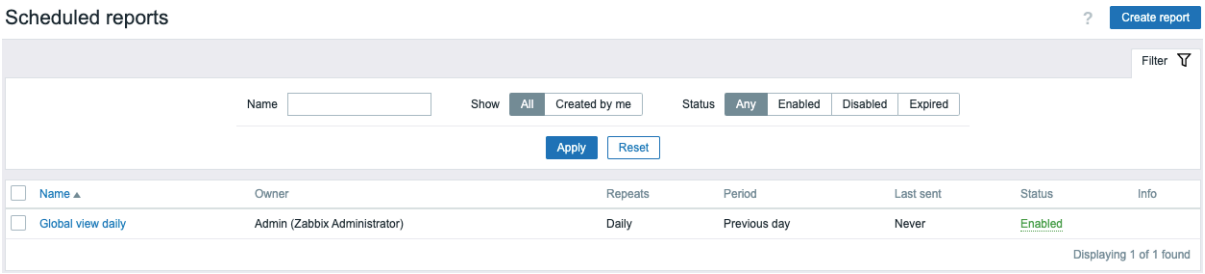
Datos mostrados:

Column	Description
Name	Nombre del nodo, tal como se define en la configuración del server.
Address	Dirección IP y puerto del nodo.
Last access	Hora del último acceso al nodo. Al pasar el cursor sobre la celda se muestra la marca de tiempo del último acceso en formato largo.
Status	Estado del nodo. Las filas de la tabla se ordenan según estos estados en el siguiente orden de prioridad: **Active** - el nodo está en funcionamiento y operando **Unavailable** - el nodo no se ha detectado durante más tiempo que el retraso de failover (quizá desee averiguar por qué) **Stopped** - el nodo se ha detenido o no pudo iniciarse (quizá desee iniciarlo o eliminarlo) **Standby** - el nodo está en funcionamiento y en espera

Informes programados

Descripción general

En *Informes* > *Informes programados*, los usuarios con permisos suficientes pueden configurar la generación programada de versiones en PDF de los paneles, que se enviarán por correo electrónico a los destinatarios especificados.



La pantalla inicial muestra información sobre los informes programados, que se puede filtrar para facilitar la navegación; consulte la sección Uso del filtro a continuación.

Datos mostrados:

Columna	Descripción
Nombre	Nombre del informe. Al hacer clic en él, se abre el formulario de configuración del informe.
Propietario	Usuario que creó el informe.
Repetición	Frecuencia de generación del informe (diaria/semanal/mensual/anual).
Período	Período de tiempo para el que se prepara el informe (día, semana, mes o año anterior).
Último envío	Fecha y hora en que se envió el informe más reciente.
Estado	Estado actual del informe (habilitado/deshabilitado/caducado). Los usuarios con permisos suficientes pueden cambiar el estado haciendo clic en él: de ”Habilitado” a ”Deshabilitado” (y viceversa); de ”Caducado” a ”Deshabilitado” (y viceversa). Para los usuarios sin permisos suficientes, no se puede hacer clic en el estado.
Información	Muestra iconos informativos: Un icono rojo indica que la generación del informe ha fallado; al pasar el cursor sobre él, se muestra una descripción emergente con información sobre el error. Un icono amarillo indica que se ha generado un informe, pero el envío a algunos o a todos los destinatarios ha fallado, o que un informe ha caducado; al pasar el cursor sobre él, se muestra una descripción emergente con información adicional.

Uso del filtro

Puede utilizar el filtro para reducir la lista de informes. Para un mejor rendimiento de búsqueda, los datos se buscan con macros sin resolver.

Las siguientes opciones de filtrado están disponibles:

- *Nombre* - se permite coincidencia parcial de nombre
- *Mostrar* - creados por el usuario actual o todos los informes

- *Estado* - seleccione entre ”Cualquiera” (mostrar todos los informes), ”Habilitado”, ”Deshabilitado” o ”Expirado”

El filtro se encuentra debajo del nombre de la sección *Informes programados*. Puede abrirse y colapsarse haciendo clic en la pestaña *Filtro* en la esquina superior derecha.

Actualización masiva

A veces puede querer eliminar o cambiar el estado de varios informes a la vez. En lugar de abrir cada informe individualmente para editarlo, puede utilizar la función de actualización masiva para ello.

Para actualizar masivamente algunos informes, haga lo siguiente:

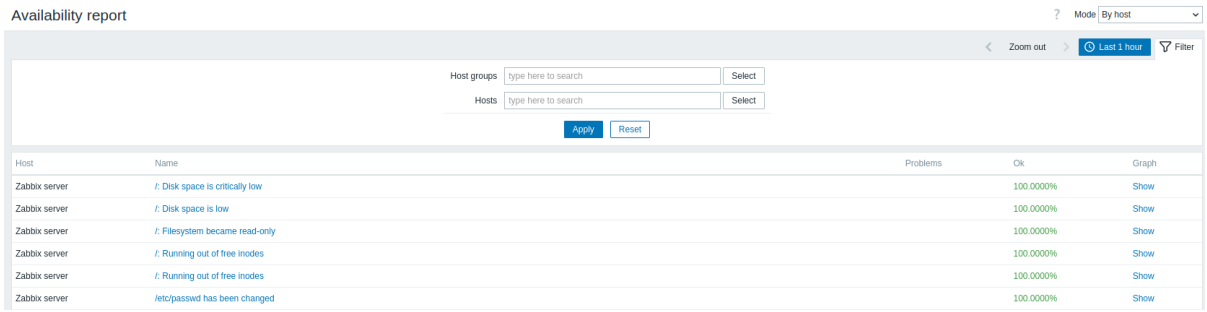
- Marque las casillas de los informes que desea actualizar en la lista
- Haga clic en el botón correspondiente debajo de la lista para realizar los cambios (*Habilitar*, *Deshabilitar* o *Eliminar*)

Informe de disponibilidad

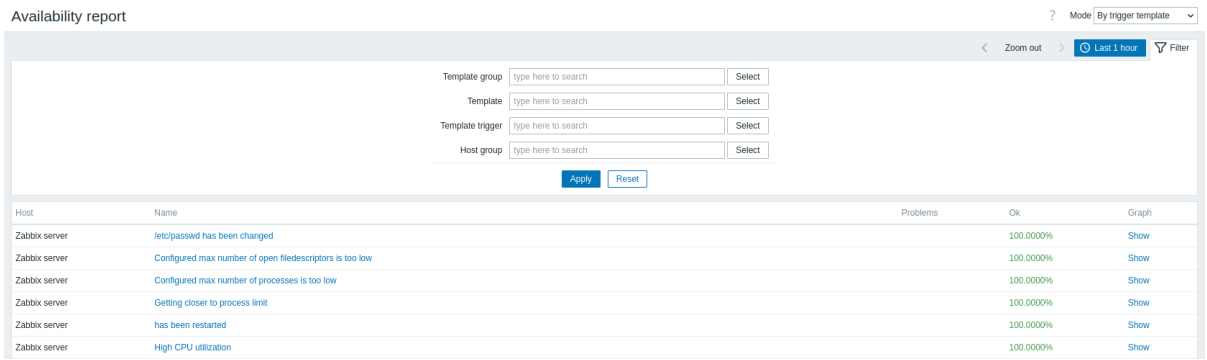
Descripción general

En *Informes* > *Informe de disponibilidad*, puede ver qué proporción de tiempo ha estado cada disparador en el estado de problema/OK.

Para cada estado, se muestra un porcentaje de tiempo, lo que facilita determinar la disponibilidad de varios elementos en su sistema.



En el menú desplegable en la esquina superior derecha, puede elegir el modo de selección: si desea mostrar los disparadores por equipos o por disparadores pertenecientes a una plantilla.



El nombre del disparador es un enlace a los eventos más recientes de ese disparador.

Uso del filtro

El filtro puede ayudar a reducir el número de hosts y/o triggers mostrados. Para mejorar el rendimiento de la búsqueda, los datos se buscan con las macros sin resolver.

El filtro se encuentra debajo del nombre de la sección *Availability report*. Se puede abrir y contraer haciendo clic en la pestaña *Filter* de la derecha.

Filtrado por plantilla de disparador

En el modo *Por plantilla de disparador*, los resultados pueden filtrarse por uno o varios de los parámetros que se enumeran a continuación.

Parámetro	Descripción
Grupo de plantillas	Filtra los equipos por disparadores que se heredan de plantillas pertenecientes al grupo de plantillas seleccionado. Especificar un grupo de plantillas principal selecciona implícitamente todos los grupos de plantillas anidados.
Plantilla	Filtra los equipos por disparadores que se heredan de la plantilla seleccionada, incluidas las plantillas anidadas. Si una plantilla anidada tiene sus propios disparadores, esos disparadores no se mostrarán.
Disparador de plantilla	Filtra los equipos por el disparador seleccionado. Otros disparadores de los equipos filtrados no se mostrarán.
Grupo de equipos	Filtra los equipos que pertenecen al grupo de equipos seleccionado.

Filtrado por equipo

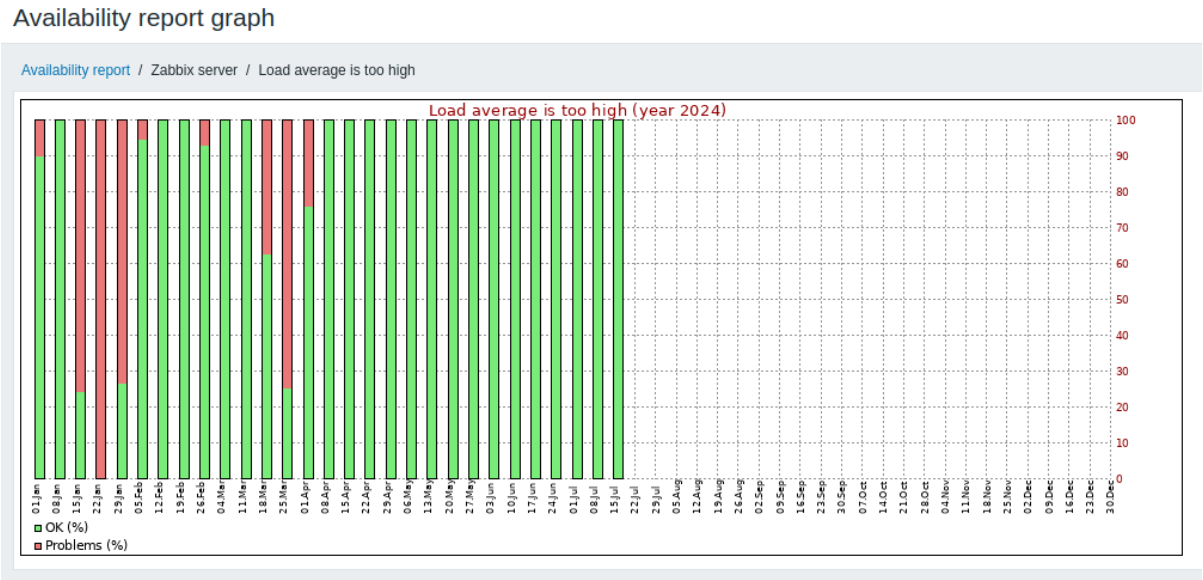
En el modo *Por equipo*, los resultados pueden filtrarse por equipo o grupo de equipos. Al especificar un grupo de equipos principal, se seleccionan implícitamente todos los grupos de equipos anidados.

Selector de período de tiempo

El selector de período de tiempo permite seleccionar períodos de tiempo de uso frecuente con un solo clic. El selector se puede expandir y contraer haciendo clic en la pestaña *Período de tiempo* situada junto al filtro.

Al hacer clic en *Mostrar* en la columna *Gráfico*, se muestra la información de disponibilidad en un gráfico de barras, donde cada barra representa una semana transcurrida del año actual.

Availability report graph



El color verde de una barra representa el tiempo en estado correcto y el rojo, el tiempo con problemas.

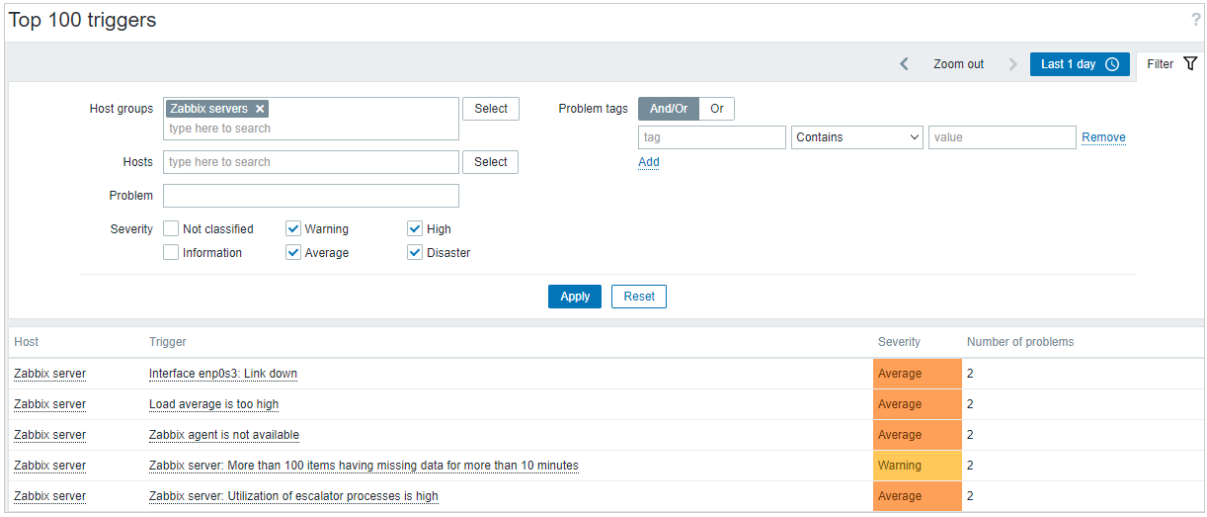
Efecto de los períodos de mantenimiento

Maintenance no excluye automáticamente el tiempo del informe de disponibilidad. Solo el mantenimiento configurado como *No data collection* detiene la sondeo (por lo que no se generan problemas) y, por lo tanto, deja sin cambios el porcentaje de disponibilidad para los triggers afectados.

Los 100 triggers principales

Descripción general

En *Informes* > *Top 100 triggers*, puede ver los triggers con el mayor número de problemas detectados durante el período seleccionado.



Las entradas de las columnas de host y trigger son enlaces que ofrecen algunas opciones útiles:

- para el host: al hacer clic en el nombre del host, se abre el menú del host
- para el trigger: al hacer clic en el nombre del trigger, se muestran enlaces a los eventos más recientes, el gráfico simple de cada item del trigger y los formularios de configuración del propio trigger y de cada item del trigger

Uso del filtro

Puede utilizar el filtro para mostrar triggers por grupo de hosts, host, nombre del problema, etiquetas o gravedad del trigger. Al especificar un grupo de hosts principal, se seleccionan implícitamente todos los grupos de hosts anidados. Para mejorar el rendimiento de la búsqueda, los datos se buscan con las macros sin resolver.

El filtro se encuentra debajo del nombre de la sección *Top 100 triggers*. Puede abrirlo y contraerlo haciendo clic en la pestaña *Filtro* situada a la derecha.

Selector de período

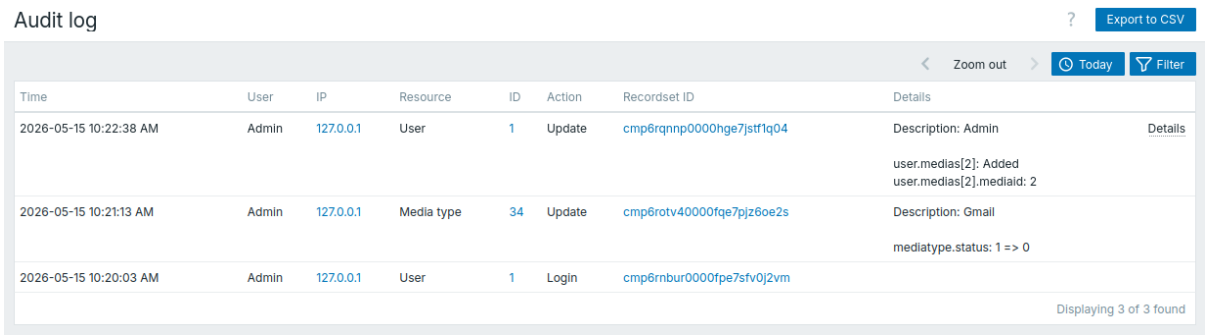
El selector de período situado junto al filtro permite seleccionar con un solo clic los períodos que se utilizan con frecuencia. Para obtener más información, consulte Selectores de período y host.

Registro de auditoría

Resumen

En la sección *Reports* > *Audit log*, se pueden ver los registros de actividad de usuarios y del sistema.

Note:
Para que los registros de auditoría se recopilen y se muestren, la casilla *Enable audit logging* debe estar marcada en la sección *Administration* > *Audit log*. Si esta opción no está habilitada, el historial de actividades no se registrará en la base de datos y no se mostrará en el audit log.



El audit log muestra los siguientes datos:

Column	Description
Time	Marca de tiempo del registro de auditoría.
User	Usuario que realizó la actividad.

Column	Description
IP	IP desde la que se inició la actividad. Al hacer clic en el hipervínculo, se filtrarán los registros del audit log por esta IP.
Resource	Tipo del recurso afectado (*API token*, *Action*, *Authentication*, *Autoregistration*, etc.).
ID	ID del recurso afectado. Al hacer clic en el hipervínculo, se filtrarán los registros del audit log por este ID de recurso.
Action	Tipo de actividad (*Add*, *Configuration refresh*, *Delete*, *Execute*, *Failed login*, *History clear*, *Login*, *Logout*, *Push*, *Update*).
Recordset ID	ID compartido para todos los registros del audit log creados como resultado de la misma operación. Por ejemplo, al vincular un template a un host, se crea un registro de audit log independiente para cada entidad de template heredada (item, trigger, etc.); todos estos registros tendrán el mismo *Recordset ID*. Al hacer clic en el hipervínculo, se filtrarán los registros del audit log por este *Recordset ID*.
Details	Descripción del recurso e información detallada sobre la actividad realizada. Si un registro contiene más de dos filas, se mostrará un enlace adicional *Details*. Haga clic en este enlace para ver la lista completa de cambios.

Note:
Cuando un trapper item o un HTTP agent item (con la captura habilitada) ha recibido algunos datos, solo se añadirá una entrada en el audit log si los datos se enviaron mediante el método de API `history.push`, y no mediante la utilidad Zabbix sender.

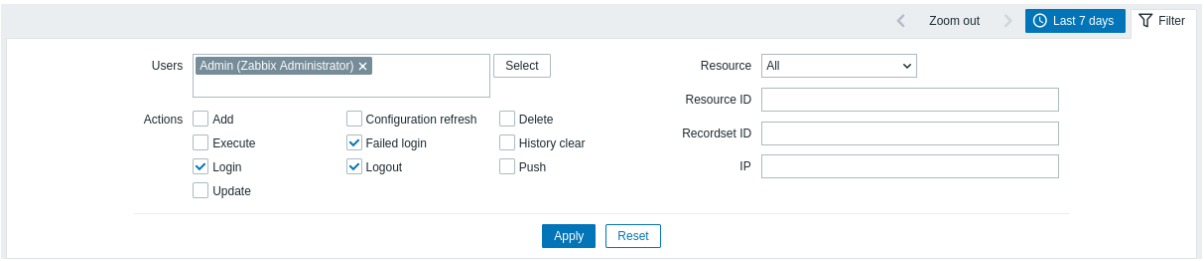
Botones

El botón en la esquina superior derecha de la página ofrece la siguiente opción:

Export to CSV	Exportar los registros del audit log de todas las páginas a un archivo CSV. Si se aplica un filtro, solo se exportarán los registros filtrados.

Usando el filtro

El filtro se encuentra debajo de la barra *Registro de auditoría*. Puede abrirse y colapsarse haciendo clic en la pestaña *Filtro* en la esquina superior derecha.



Puede usar el filtro para limitar los registros por usuario, recurso afectado, ID de recurso, operación realizada (*ID de conjunto de registros*) e IP. Dependiendo del recurso, se pueden seleccionar una o más acciones específicas en el filtro.

Para un mejor rendimiento de búsqueda, todos los datos se buscan con las macros sin resolver.

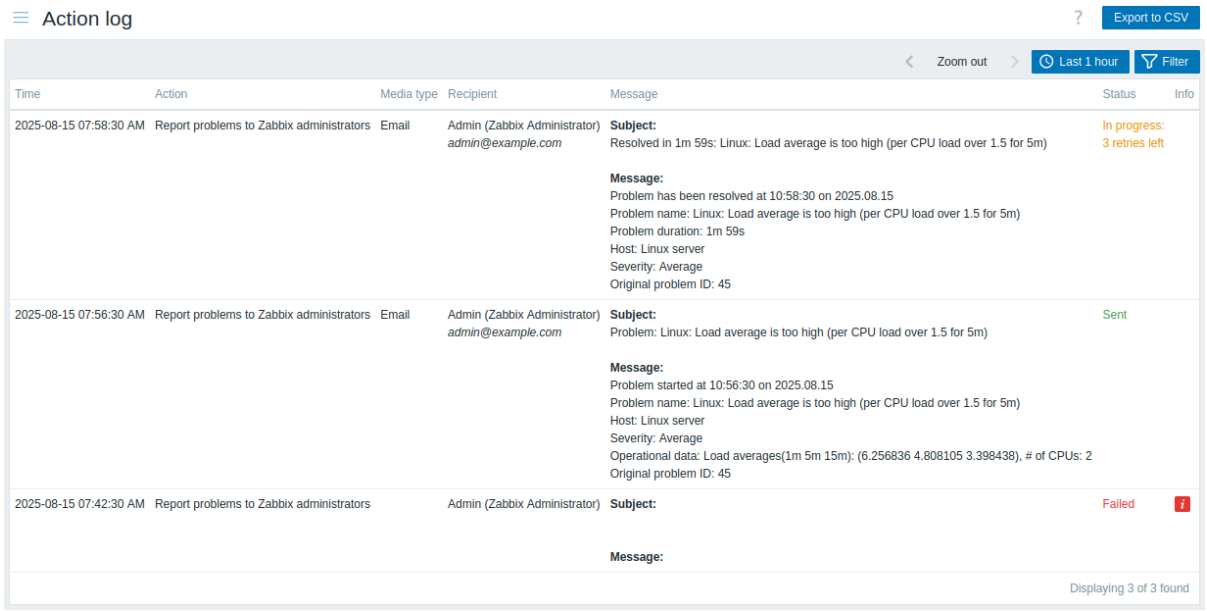
Selector de período de tiempo

El selector de período de tiempo situado junto al filtro permite seleccionar períodos utilizados con frecuencia con un solo clic del ratón. Para obtener más información, consulte Selectores de período de tiempo y host.

Registro de acciones

Descripción general

En *Reports* > *Action log*, puede ver los detalles de las operations (notificaciones, comandos remotos) ejecutadas dentro de una acción.



Datos mostrados:

Column	Description
Time	Marca de tiempo de la operación.
Action	Nombre de la acción que provoca las operaciones.
Media type	Tipo de medio (por ejemplo, Email, Jira, etc.) utilizado para enviar una notificación. Para las operaciones que ejecutaron comandos remotos, esta columna estará vacía.
Recipient	Información sobre el destinatario de la notificación: nombre de usuario, nombre y apellido (entre paréntesis), e información adicional según el tipo de medio (correo electrónico, nombre de usuario, etc.). Para las operaciones que ejecutaron comandos remotos, esta columna estará vacía.
Message	El contenido del mensaje, el comando remoto o el nombre del script global. Un comando remoto se separa del host de destino con el símbolo de dos puntos: `<host>:<command>`. Por ejemplo, si el comando remoto se ejecutó en Zabbix server, la información tendrá el siguiente formato: `Zabbix server:<command>`.
Status	Estado de la operación: *In progress* - la operación de envío de una notificación está en curso (también se muestra el número restante de veces que el server intentará enviar la notificación) *Sent* - la notificación se ha enviado *Executed* - el comando remoto se ha ejecutado *Failed* - la operación no se ha completado
Info	Información de error (si la hay) relacionada con la ejecución de la operación.

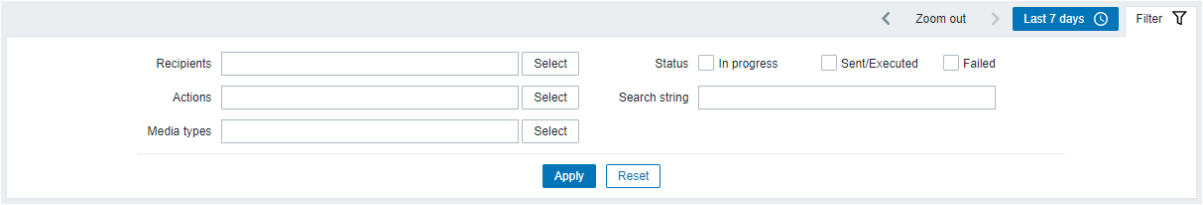
Botones

El botón en la esquina superior derecha de la página ofrece la siguiente opción:

Export to CSV	Exportar los registros del log de acciones de la página actual a un archivo CSV. Si se aplica un filtro, solo se exportarán los registros filtrados. En el archivo CSV exportado, las columnas ”Destinatario” y ”Mensaje” se dividen en varias columnas: ”Nombre de usuario de Zabbix del destinatario”, ”Nombre del destinatario”, ”Apellido del destinatario”, ”Destinatario”, y ”Asunto”, ”Mensaje”, ”Comando”.

Uso del filtro

El filtro se encuentra debajo de la barra *Action log*. Se puede abrir y contraer haciendo clic en la pestaña *Filter* en la esquina superior derecha de la página.



Puede usar el filtro para acotar los registros por destinatarios de notificaciones, acciones, tipos de medios, estado, o por el contenido del mensaje/comando remoto (*Search string*). Para un mejor rendimiento de búsqueda, los datos se buscan con las macros sin resolver.

Selector de período de tiempo

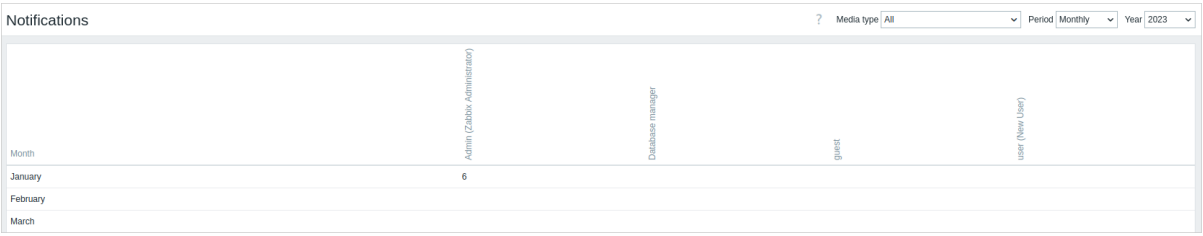
El selector de período de tiempo situado junto al filtro permite seleccionar períodos utilizados con frecuencia con un solo clic del ratón. Para obtener más información, consulte Selectores de período de tiempo y host.

Notificaciones

Resumen

En la sección *Reports* > *Notifications* se muestra un informe sobre el número de notificaciones enviadas a cada usuario.

En las listas desplegables de la esquina superior derecha puede elegir el tipo de medio (o todos), el período (datos de cada día/semana/mes/año) y el año de las notificaciones enviadas.



Cada columna muestra los totales por cada usuario del sistema.

Recopilación de datos

Descripción general

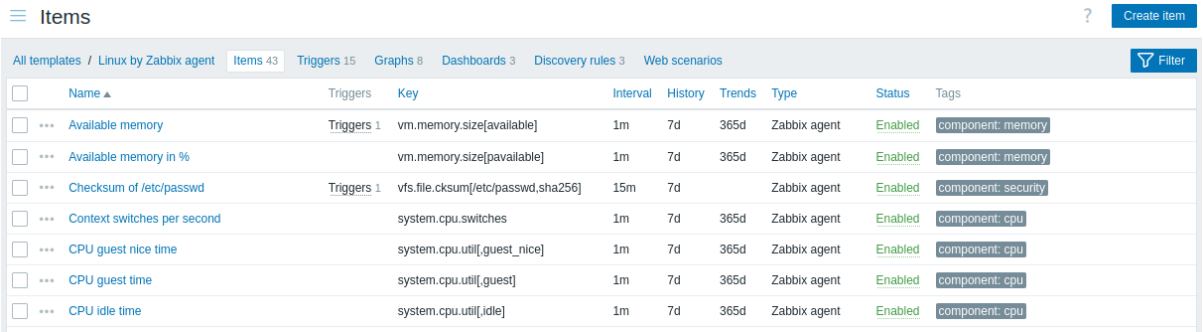
Este menú contiene secciones relacionadas con la configuración de la recopilación de datos.

Items

Descripción general

Se puede acceder a la lista de items de un template desde *Recopilación de datos* > *Templates*, haciendo clic en *Items* para el template correspondiente.

Se muestra una lista de los items existentes.



Datos mostrados:

Columna	Descripción
Menú del item	Haga clic en el icono de tres puntos para abrir el menú del item.
Template	Template al que pertenece el item. Al hacer clic en el nombre del template, se abre el formulario de configuración del template. Esta columna solo se muestra si se seleccionan varios templates o ningún template en el filtro.
Nombre	Nombre del item, mostrado como un enlace azul a los detalles del item. Al hacer clic en el enlace con el nombre del item, se abre el formulario de configuración del item. Si el item se hereda de otro template, el nombre del template se muestra antes del nombre del item como un enlace gris. Al hacer clic en el enlace del template, se abre la lista de items en ese nivel del template.
Triggers	Al pasar el ratón sobre Triggers, se muestra un cuadro informativo con los triggers asociados al item. El número de triggers se muestra en gris.
Clave	Se muestra la clave del item.
Intervalo	Se muestra la frecuencia de la comprobación.
Historial	Se muestra durante cuántos días se conservará el historial de datos del item.
Tendencias	Se muestra durante cuántos días se conservará el historial de tendencias del item.
Tipo	Se muestra el tipo de item (Zabbix agent, SNMP agent, comprobación simple, etc.).
Estado	Se muestra el estado del item: *Habilitado* o *Deshabilitado*. Al hacer clic en el estado, puede cambiarlo de Habilitado a Deshabilitado (y viceversa).
Etiquetas	Se muestran las etiquetas del item. Se pueden mostrar hasta tres etiquetas (pares nombre:valor). Si hay más etiquetas, se muestra un enlace "..." que permite ver todas las etiquetas al pasar el ratón sobre él.

Para configurar un item nuevo, haga clic en el botón *Crear item* situado en la esquina superior derecha.

Opciones de edición masiva

Los botones debajo de la lista ofrecen algunas opciones de edición masiva:

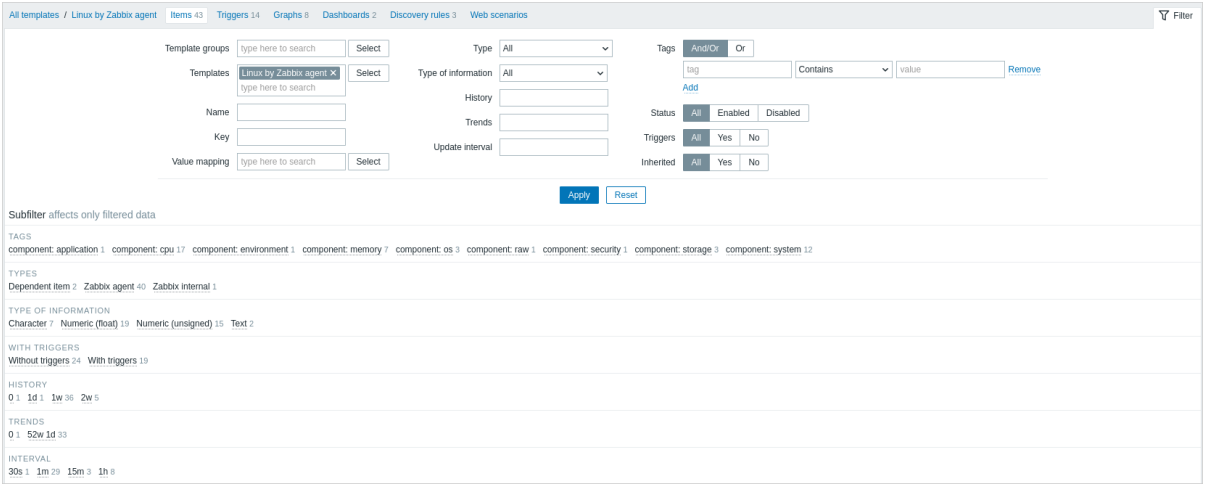
- *Enable* - cambia el estado del item a *Enabled*.
- *Disable* - cambia el estado del item a *Disabled*.
- *Copy* - copia los items a otros hosts o templates.
- *Mass update* - actualiza varias propiedades de varios items a la vez.
- *Delete* - elimina los items.

Para usar estas opciones, marque las casillas antes de los items correspondientes y luego haga clic en el botón requerido.

Uso del filtro

La lista de items puede contener una gran cantidad de items. Mediante el uso del filtro, puede filtrar algunos de ellos para localizar rápidamente los items que está buscando. Para un mejor rendimiento de búsqueda, los datos se buscan con macros sin resolver.

El icono *Filter* está disponible en la esquina superior derecha. Al hacer clic en él se abrirá un filtro donde puede especificar los criterios de filtrado deseados.



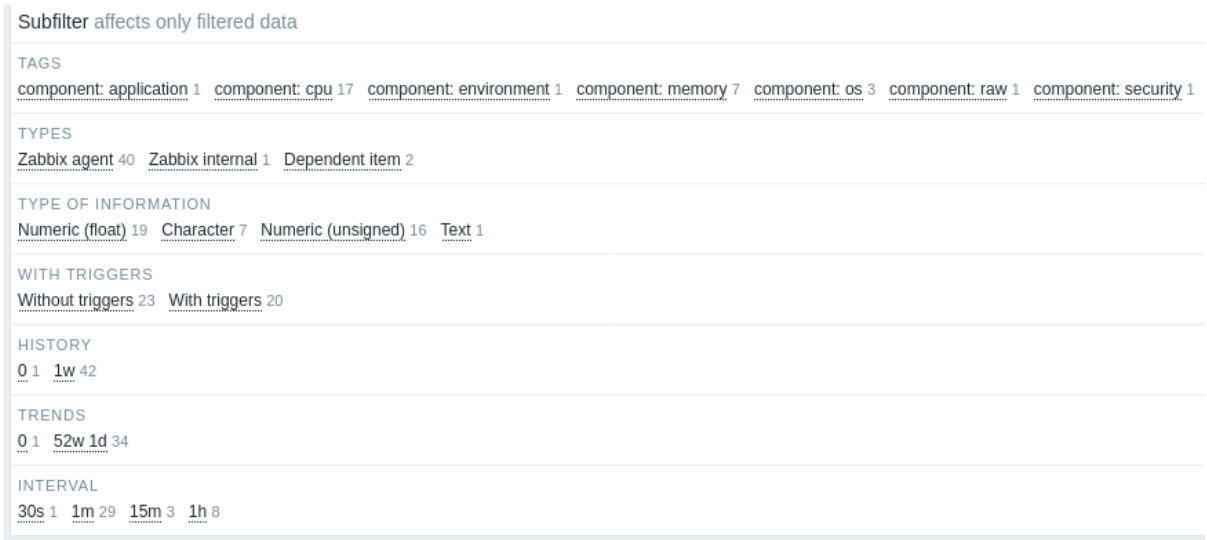
Parameter	Description
Template groups	Filtrar por uno o más grupos de templates. Especificar un grupo de templates padre selecciona implícitamente todos los grupos anidados.

Parameter	Description
Templates	Filtrar por uno o más templates.
Name	Filtrar por nombre del item.
Key	Filtrar por la clave del item.
Value mapping	Filtrar por el mapa de valores utilizado. Este parámetro no se muestra si la opción *Templates* está vacía.
Type	Filtrar por tipo de item (Zabbix agent, SNMP agent, etc.).
Type of information	Filtrar por tipo de información (Numeric unsigned, float, etc.).
History	Filtrar por cuánto tiempo se conserva el historial del item.
Trends	Filtrar por cuánto tiempo se conservan las tendencias del item.
Update interval	Filtrar por el intervalo de actualización del item.
Tags	Especifique etiquetas para limitar el número de items mostrados. Es posible incluir y excluir etiquetas y valores de etiqueta específicos. Se pueden establecer varias condiciones. La coincidencia del nombre de la etiqueta siempre distingue entre mayúsculas y minúsculas. Hay varios operadores disponibles para cada condición: **Exists** - incluir los nombres de etiqueta especificados **Equals** - incluir los nombres y valores de etiqueta especificados (distingue mayúsculas y minúsculas) **Contains** - incluir los nombres de etiqueta especificados cuyos valores de etiqueta contengan la cadena introducida (coincidencia de subcadena, no distingue mayúsculas y minúsculas) **Does not exist** - excluir los nombres de etiqueta especificados **Does not equal** - excluir los nombres y valores de etiqueta especificados (distingue mayúsculas y minúsculas) **Does not contain** - excluir los nombres de etiqueta especificados cuyos valores de etiqueta contengan la cadena introducida (coincidencia de subcadena, no distingue mayúsculas y minúsculas) Hay dos tipos de cálculo para las condiciones: **And/Or** - todas las condiciones deben cumplirse; las condiciones que tengan el mismo nombre de etiqueta se agruparán mediante la condición Or **Or** - basta con que se cumpla una condición
Status	Filtrar por el estado del item: *Enabled* o *Disabled*.
Triggers	Filtrar items con (o sin) triggers.
Inherited	Filtrar items heredados (o no heredados) de templates vinculados.

Uso del subfiltro

El subfiltro permite modificar aún más el filtrado aplicado por el filtro principal.

Contiene enlaces en los que se puede hacer clic para acceder rápidamente a los elementos relacionados. Los elementos están relacionados por una entidad común: etiqueta, tipo de item, intervalo de actualización del item, etc. Al hacer clic en una entidad, esta se resalta con un fondo gris y los elementos se filtran inmediatamente (no es necesario hacer clic en *Aplicar* en el filtro principal). Al hacer clic en otra entidad, esta se añade a los resultados filtrados. Al volver a hacer clic en la entidad, se elimina el filtrado.



Los subfiltros se generan a partir de los datos filtrados, que están limitados a 1000 registros. Si desea ver más registros en el subfiltro, debe aumentar el valor del parámetro *Límite para los resultados de búsqueda y filtrado* (en *Administración* > *General* >

GUI).

A diferencia del filtro principal, el subfiltro se actualiza con cada solicitud de actualización de la tabla para mostrar siempre información actualizada sobre las opciones de filtrado disponibles y sus contadores.

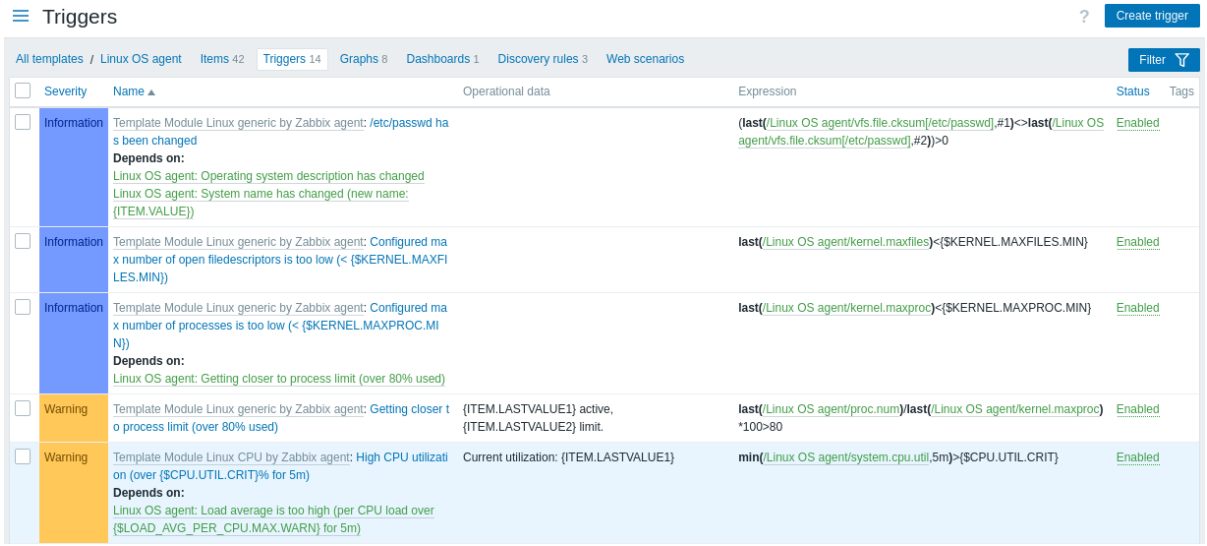
El número de entidades mostradas está limitado a 100 horizontalmente. Si hay más, se muestra un icono de tres puntos al final; no se puede hacer clic en él.

Un número junto a cada entidad en la que se puede hacer clic indica el número de items agrupados en ella (según los resultados del filtro principal). Cuando se hace clic en una entidad, los números de las demás entidades disponibles se muestran con un signo más, que indica cuántos items se pueden añadir a la selección actual. Las entidades sin items no se muestran, a menos que se hayan seleccionado previamente en el subfiltro.

Triggers

Descripción general

Se puede acceder a la lista de triggers de un template desde *Recopilación de datos* > *Templates*, haciendo clic en *Triggers* para el template correspondiente.



Datos mostrados:

Columna	Descripción
Gravedad	La gravedad del trigger se muestra tanto mediante el nombre como mediante el color de fondo de la celda.
Template	Template al que pertenece el trigger. Al hacer clic en el nombre del template, se abre el formulario de configuración del template. Esta columna solo se muestra si se seleccionan varios templates o ningún template en el filtro.
Nombre	Nombre del trigger, mostrado como un enlace azul a los detalles del trigger. Al hacer clic en el enlace del nombre del trigger, se abre el formulario de configuración del trigger. Si el trigger se hereda de otro template, el nombre del template se muestra antes del nombre del trigger como un enlace gris. Al hacer clic en el enlace del template, se abre la lista de triggers en ese nivel del template.
Datos operativos	Definición de los datos operativos del trigger, que contiene cadenas arbitrarias y macros que se resolverán dinámicamente en *Monitorización* > *Problemas*.
Expresión	Se muestra la expresión del trigger. La parte de la expresión correspondiente al item del template se muestra como un enlace que lleva al formulario de configuración del item.
Estado	Se muestra el estado del trigger: *Habilitado* o *Deshabilitado*. Al hacer clic en el estado, puede cambiarlo de Habilitado a Deshabilitado, y viceversa.
Etiquetas	Si un trigger contiene etiquetas, en esta columna se muestran el nombre y el valor de la etiqueta.

Para configurar un nuevo trigger, haga clic en el botón *Crear trigger* situado en la esquina superior derecha.

Opciones de edición masiva

Los botones debajo de la lista ofrecen algunas opciones de edición masiva:

- *Enable* - cambiar el estado del trigger a *Enabled*
- *Disable* - cambiar el estado del trigger a *Disabled*
- *Copy* - copiar los triggers a otros hosts o templates
- *Mass update* - actualizar varias propiedades de varios triggers a la vez
- *Delete* - eliminar los triggers

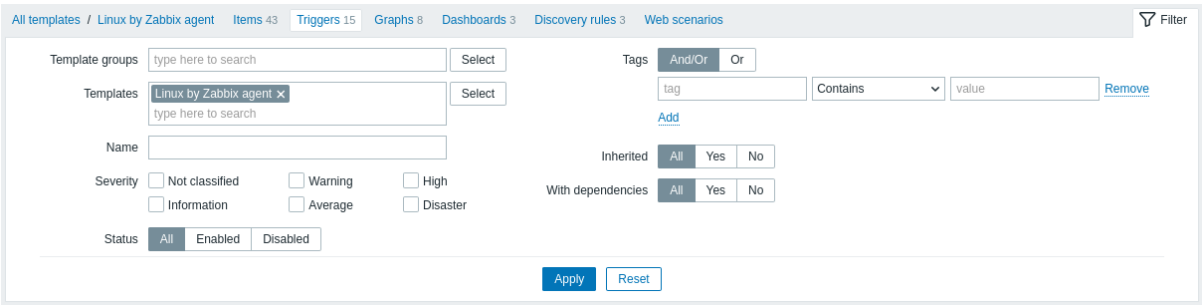
Para usar estas opciones, marque las casillas antes de los triggers correspondientes y luego haga clic en el botón requerido.

Uso del filtro

Puede usar el filtro para mostrar solo los triggers que le interesen.

Para un mejor rendimiento de búsqueda, los datos se buscan con macros sin resolver.

El icono *Filter* está disponible en la esquina superior derecha. Al hacer clic en él, se abrirá un filtro donde podrá especificar los criterios de filtrado deseados.



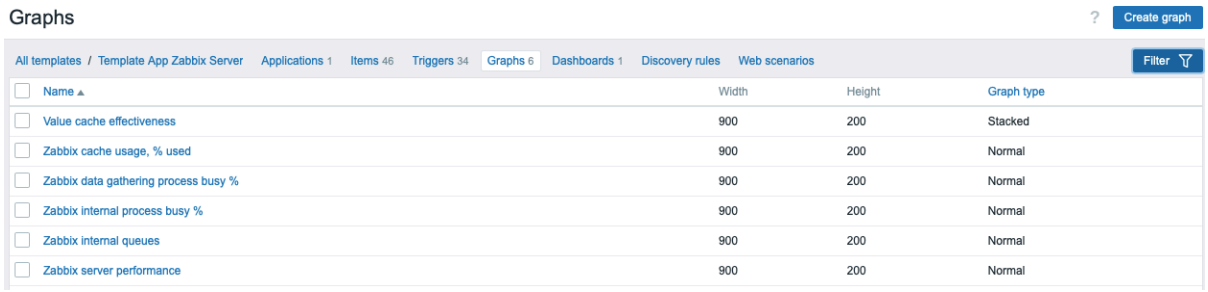
Parameter	Description
Template groups	Filtrar por uno o varios grupos de templates. Especificar un grupo de templates padre selecciona implícitamente todos los grupos anidados.
Templates	Filtrar por uno o varios templates. Si ya se han seleccionado grupos de templates arriba, la selección de templates se limita a esos grupos.
Name	Filtrar por nombre de trigger.
Severity	Seleccione para filtrar por una o varias severidades de trigger.
Status	Filtrar por el estado de activación del trigger (Enabled/Disabled).
Tags	Filtrar por nombre y valor de la etiqueta del trigger. Es posible incluir y excluir etiquetas y valores de etiqueta específicos. Se pueden establecer varias condiciones. La coincidencia del nombre de la etiqueta siempre distingue entre mayúsculas y minúsculas. Hay varios operadores disponibles para cada condición: **Exists** - incluir los nombres de etiqueta especificados **Equals** - incluir los nombres y valores de etiqueta especificados (distingue mayúsculas y minúsculas) **Contains** - incluir los nombres de etiqueta especificados cuyos valores de etiqueta contengan la cadena introducida (coincidencia de subcadena, no distingue mayúsculas y minúsculas) **Does not exist** - excluir los nombres de etiqueta especificados **Does not equal** - excluir los nombres y valores de etiqueta especificados (distingue mayúsculas y minúsculas) **Does not contain** - excluir los nombres de etiqueta especificados cuyos valores de etiqueta contengan la cadena introducida (coincidencia de subcadena, no distingue mayúsculas y minúsculas) Hay dos tipos de cálculo para las condiciones: **And/Or** - deben cumplirse todas las condiciones; las condiciones que tengan el mismo nombre de etiqueta se agruparán mediante la condición Or **Or** - basta con que se cumpla una condición Se admiten macros y macro functions en los campos de nombre y valor de la etiqueta.
Inherited	Filtrar triggers heredados (o no heredados) de templates vinculados.
With dependencies	Filtrar triggers con dependencias (o sin dependencias).

Gráficos

Descripción general

Se puede acceder a la lista de gráficos personalizados de un template desde *Recopilación de datos* > *Templates*, haciendo clic en *Gráficos* para el template correspondiente.

Se muestra una lista de los gráficos existentes.



Datos mostrados:

Columna	Descripción
Template	Template al que pertenece el gráfico. Al hacer clic en el nombre del template, se abre el formulario de configuración del template. Esta columna solo se muestra si se seleccionan varios templates o ningún template en el filtro.
Nombre	Nombre del gráfico personalizado, mostrado como un enlace azul a los detalles del gráfico. Al hacer clic en el enlace del nombre del gráfico, se abre el formulario de configuración del gráfico. Si el gráfico se hereda de otro template, el nombre del template se muestra antes del nombre del gráfico como un enlace gris. Al hacer clic en el enlace del template, se abre la lista de gráficos en ese nivel del template.
Ancho	Se muestra el ancho del gráfico.
Alto	Se muestra el alto del gráfico.
Tipo de gráfico	Se muestra el tipo de gráfico: *Normal*, *Apilado*, *Circular* o *Explotado*.

Para configurar un gráfico nuevo, haga clic en el botón *Crear gráfico* en la esquina superior derecha.

Opciones de edición masiva

Los botones debajo de la lista ofrecen algunas opciones de edición masiva:

- *Copy* - copiar los gráficos a otros hosts o templates
- *Delete* - eliminar los gráficos

Para usar estas opciones, marque las casillas de verificación delante de los gráficos correspondientes y luego haga clic en el botón requerido.

Usando el filtro

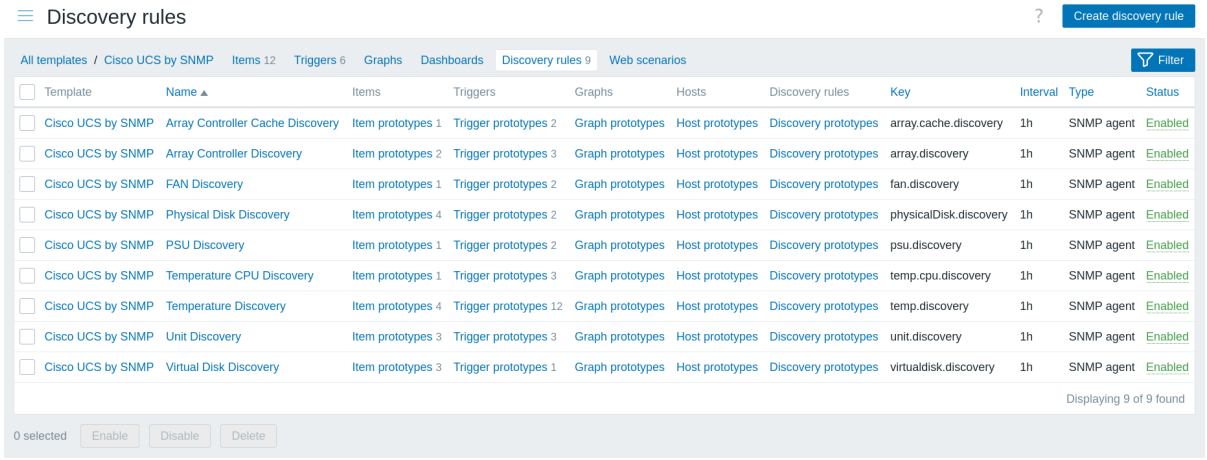
Puede filtrar los gráficos por grupo de plantillas y plantilla. Para un mejor rendimiento de búsqueda, los datos se buscan con las macros sin resolver.

Reglas de descubrimiento

Descripción general

Se puede acceder a la lista de reglas de descubrimiento de bajo nivel de un template desde *Recopilación de datos* > *Templates*, haciendo clic en *Descubrimiento* para el template correspondiente.

Se muestra una lista de las reglas de descubrimiento de bajo nivel existentes. También es posible ver todas las reglas de descubrimiento independientemente del template, o todas las reglas de descubrimiento de un grupo de templates específico, cambiando la configuración del filtro.



Datos mostrados:

Columna	Descripción
Template	El template al que pertenece la regla de descubrimiento. Al hacer clic en el nombre del template, se abre el formulario de configuración del template.
Nombre	Nombre de la regla, mostrado como un enlace azul. Al hacer clic en el nombre de la regla, se abre el formulario de configuración de la regla de descubrimiento de bajo nivel. Si la regla de descubrimiento se hereda de otro template, el nombre del template se muestra antes del nombre de la regla como un enlace gris. Al hacer clic en el enlace del template, se abre la lista de reglas de descubrimiento en ese nivel del template.
Items	Se muestra un enlace a la lista de prototipos de item. El número de prototipos de item existentes se muestra en gris.
Triggers	Se muestra un enlace a la lista de prototipos de trigger. El número de prototipos de trigger existentes se muestra en gris.
Gráficos	Se muestra un enlace a la lista de prototipos de gráficos. El número de prototipos de gráficos existentes se muestra en gris.
Hosts	Se muestra un enlace a la lista de prototipos de host. El número de prototipos de host existentes se muestra en gris.
Clave	Se muestra la clave del item utilizada para el descubrimiento.
Intervalo	Se muestra la frecuencia con la que se realiza el descubrimiento.
Tipo	Se muestra el tipo de item utilizado para el descubrimiento (Zabbix agent, SNMP agent, etc.).
Estado	Se muestra el estado de la regla de descubrimiento: *Habilitada* o *Deshabilitada*. Al hacer clic en el estado, puede cambiarlo de Habilitada a Deshabilitada (y viceversa).

Para configurar una nueva regla de descubrimiento de bajo nivel, haga clic en el botón *Crear regla de descubrimiento*, situado en la esquina superior derecha.

Opciones de edición masiva

Los botones debajo de la lista ofrecen algunas opciones de edición masiva:

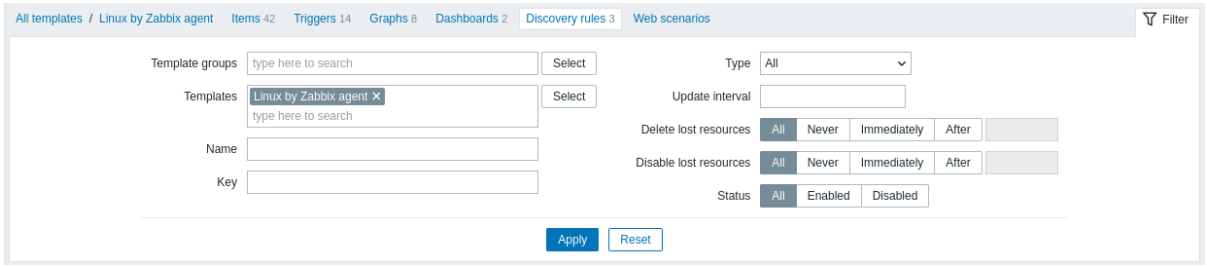
- *Enable* - cambia el estado de la regla de descubrimiento de bajo nivel a *Enabled*
- *Disable* - cambia el estado de la regla de descubrimiento de bajo nivel a *Disabled*
- *Delete* - elimina las reglas de descubrimiento de bajo nivel

Para usar estas opciones, marque las casillas de verificación delante de las respectivas reglas de descubrimiento y, a continuación, haga clic en el botón requerido.

Uso del filtro

Puede usar el filtro para mostrar solo las reglas de descubrimiento que le interesen. Para un mejor rendimiento de búsqueda, los datos se buscan con macros sin resolver.

El icono *Filter* está disponible en la esquina superior derecha. Al hacer clic en él, se abrirá un filtro donde puede especificar los criterios de filtrado deseados, como template, nombre de la regla de descubrimiento, clave del item, tipo de item, etc.



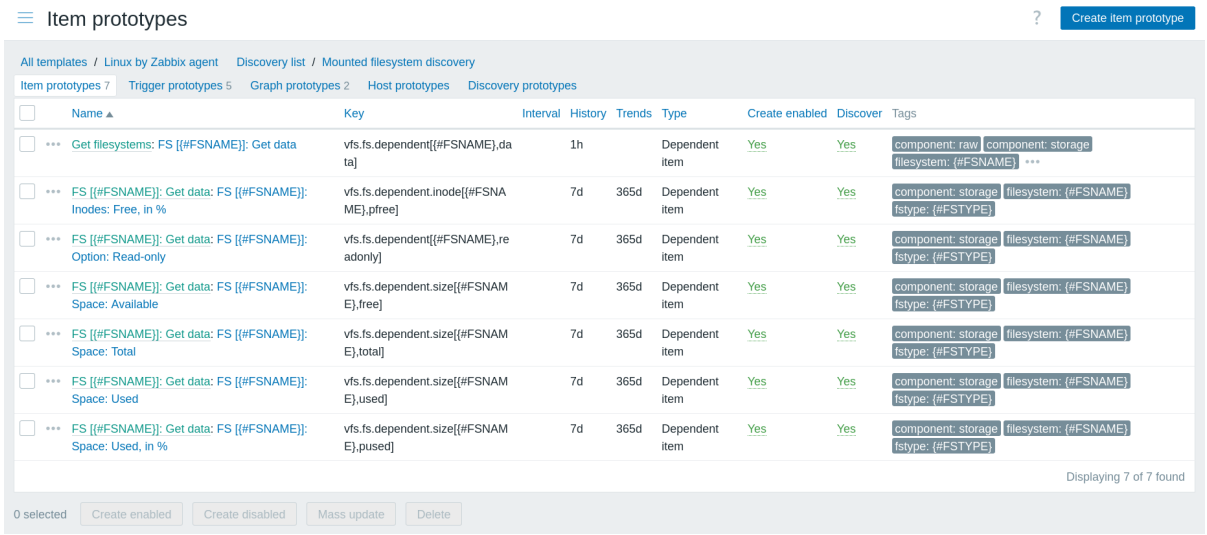
Parameter	Description
Template groups	Filtrar por uno o más grupos de template. Especificar un grupo de template padre selecciona implícitamente todos los grupos anidados.
Templates	Filtrar por uno o más templates.
Name	Filtrar por nombre de la regla de descubrimiento.
Key	Filtrar por clave del item de descubrimiento.
Type	Filtrar por tipo de item de descubrimiento.
Update interval	Filtrar por intervalo de actualización. No disponible para Zabbix trapper y items dependientes.
Delete lost resources	Filtrar por el período de *Delete lost resources*.
Disable lost resources	Filtrar por el período de *Disable lost resources*.
Status	Filtrar por el estado de la regla de descubrimiento (All/Enabled/Disabled).

Prototipos de item

Descripción general

En esta sección se muestran los prototipos de item configurados de una regla de descubrimiento de bajo nivel en el template.

Si el template está vinculado al host, los prototipos de item se utilizarán como base para crear items reales en el host durante el descubrimiento de bajo nivel.



Datos mostrados:

Column	Description
Name	Se muestra el nombre del prototipo de item como un enlace azul. Al hacer clic en el nombre, se abre el formulario de configuración del prototipo de item. Si el prototipo de item pertenece a un template vinculado, el nombre del template se muestra antes del nombre del prototipo de item como un enlace gris. Al hacer clic en el enlace del template, se abre la lista de prototipos de item en el nivel del template vinculado. Si el prototipo de item es del tipo ”Dependent item”, el nombre del item maestro se muestra antes del nombre del prototipo de item en color verde (como en el caso de los prototipos de item en el nivel del host).
Key	Se muestra la clave del prototipo de item.

Column	Description
Interval	Se muestra la frecuencia de la comprobación.
History	Se muestra durante cuántos días se conservará el historial de datos del item.
Trends	Se muestra durante cuántos días se conservará el historial de tendencias del item.
Type	Se muestra el tipo de prototipo de item (Zabbix agent, SNMP agent, comprobación simple, etc.).
Create enabled	Crear el item basándose en este prototipo como: **Yes**: habilitado **No**: deshabilitado. Puede cambiar entre ”Yes” y ”No” haciendo clic en ellos.
Discover	Descubrir el item basándose en este prototipo: **Yes**: descubrir **No**: no descubrir. Puede cambiar entre ”Yes” y ”No” haciendo clic en ellos.
Tags	Se muestran las etiquetas del prototipo de item.

Para configurar un nuevo prototipo de item, haga clic en el botón *Create item prototype* situado en la esquina superior derecha.

Opciones de edición masiva

Los botones debajo de la lista ofrecen algunas opciones de edición masiva:

- *Crear habilitado* - crear estos items como *Habilitados*
- *Crear deshabilitado* - crear estos items como *Deshabilitados*
- *Actualización masiva* - actualizar masivamente estos prototipos de item
- *Eliminar* - eliminar estos prototipos de item

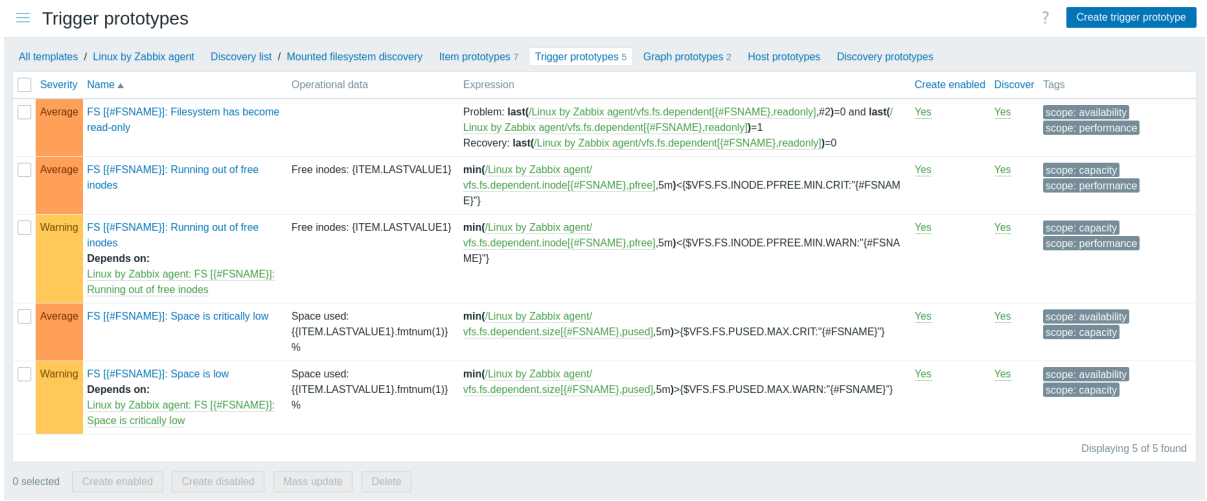
Para usar estas opciones, marque las casillas de verificación delante de los prototipos de item correspondientes y, a continuación, haga clic en el botón requerido.

Prototipos de trigger

Descripción general

En esta sección se muestran los prototipos de trigger configurados de una regla de descubrimiento de bajo nivel en el template.

Si el template está vinculado al host, los prototipos de trigger serán la base para crear triggers reales del host durante el descubrimiento de bajo nivel.



Datos mostrados:

Column	Description
Severity	La gravedad del trigger se muestra tanto mediante el nombre como mediante el color de fondo de la celda.
Name	Nombre del prototipo de trigger, mostrado como un enlace azul. Al hacer clic en el nombre, se abre el formulario de configuración del prototipo de trigger. Si el prototipo de trigger pertenece a un template vinculado, el nombre del template se muestra antes del nombre del trigger como un enlace gris. Al hacer clic en el enlace del template, se abre la lista de prototipos de trigger en el nivel del template vinculado.

Column	Description
Expression	Se muestra la expresión del trigger. La parte de la expresión correspondiente al item del template se muestra como un enlace que lleva al formulario de configuración del item.
Operational data	Se muestra el formato de los datos operativos del trigger, que contiene cadenas arbitrarias y macros que se resolverán dinámicamente en *Monitoring* > *Problems*.
Create enabled	Crear el trigger basándose en este prototipo como: **Yes** - habilitado **No** - deshabilitado. Puede cambiar entre 'Yes' y 'No' haciendo clic en ellos.
Discover	Descubrir el trigger basándose en este prototipo: **Yes** - descubrir **No** - no descubrir. Puede cambiar entre 'Yes' y 'No' haciendo clic en ellos.
Tags	Se muestran las etiquetas del prototipo de trigger.

Para configurar un nuevo prototipo de trigger, haga clic en el botón *Create trigger prototype* situado en la esquina superior derecha.

Opciones de edición masiva

Los botones debajo de la lista ofrecen algunas opciones de edición masiva:

- *Crear habilitado* - crear estos triggers como *Habilitado*
- *Crear deshabilitado* - crear estos triggers como *Deshabilitado*
- *Actualización masiva* - actualizar masivamente estos prototipos de trigger
- *Eliminar* - eliminar estos prototipos de trigger

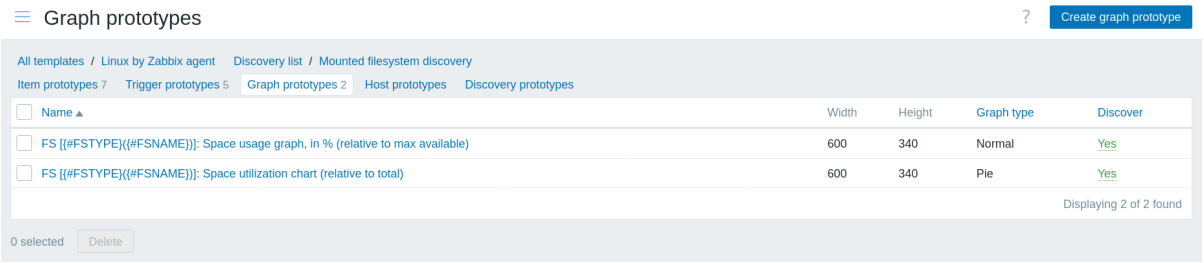
Para usar estas opciones, marque las casillas de verificación delante de los respectivos prototipos de trigger y, a continuación, haga clic en el botón requerido.

Prototipos de gráficos

Descripción general

En esta sección se muestran los prototipos de gráficos configurados de una regla de descubrimiento de bajo nivel en el template.

Si el template está vinculado al host, los prototipos de gráficos serán la base para crear gráficos reales del host durante el descubrimiento de bajo nivel.



Datos mostrados:

Column	Description
Name	Nombre del prototipo de gráfico, mostrado como un enlace azul. Al hacer clic en el nombre, se abre el formulario de configuración del prototipo de gráfico. Si el prototipo de gráfico pertenece a un template vinculado, el nombre del template se muestra antes del nombre del gráfico como un enlace gris. Al hacer clic en el enlace del template, se abre la lista de prototipos de gráficos en el nivel del template vinculado.
Width	Se muestra el ancho del prototipo de gráfico.
Height	Se muestra la altura del prototipo de gráfico.
Type	Se muestra el tipo de prototipo de gráfico: *Normal*, *Stacked*, *Pie* o *Exploded*.
Discover	Descubrir el gráfico basándose en este prototipo: **Yes**: descubrir **No**: no descubrir. Puede alternar entre «Yes» y «No» haciendo clic en ellas.

Para configurar un nuevo prototipo de gráfico, haga clic en el botón *Create graph prototype* situado en la esquina superior derecha.

Opciones de edición masiva

Los botones debajo de la lista ofrecen algunas opciones de edición masiva:

- *Eliminar* - elimina estos prototipos de gráfico

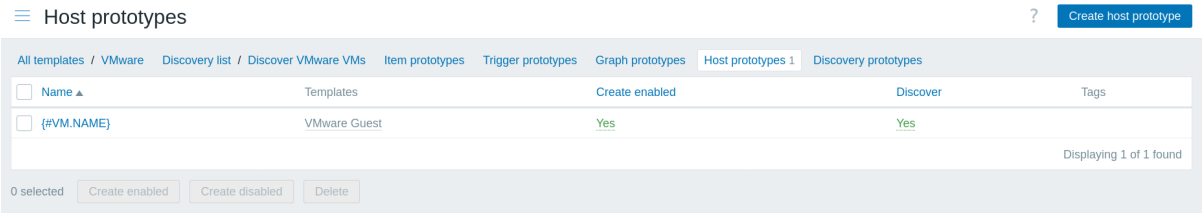
Para usar estas opciones, marque las casillas de verificación delante de los respectivos prototipos de gráfico y, a continuación, haga clic en el botón requerido.

Prototipos de host

Descripción general

En esta sección se muestran los prototipos de host configurados de una regla de descubrimiento de bajo nivel en el template.

Si el template está vinculado al host, los prototipos de host serán la base para crear hosts reales durante el descubrimiento de bajo nivel.



Datos mostrados:

Column	Description
Name	Nombre del prototipo de host, mostrado como un enlace azul. Al hacer clic en el nombre, se abre el formulario de configuración del prototipo de host. Si el prototipo de host pertenece a un template vinculado, el nombre del template se muestra antes del nombre del host como un enlace gris. Al hacer clic en el enlace del template, se abre la lista de prototipos de host en el nivel del template vinculado.
Templates	Se muestran los templates del prototipo de host.
Create enabled	Crear el host basándose en este prototipo como: **Yes** - habilitado **No** - deshabilitado. Puede cambiar entre «Yes» y «No» haciendo clic en ellas.
Discover	Descubrir el host basándose en este prototipo: **Yes** - descubrir **No** - no descubrir. Puede cambiar entre «Yes» y «No» haciendo clic en ellas.
Tags	Se muestran las etiquetas del prototipo de host.

Para configurar un nuevo prototipo de host, haga clic en el botón *Create host prototype* situado en la esquina superior derecha.

Opciones de edición masiva

Los botones debajo de la lista ofrecen algunas opciones de edición masiva:

- *Crear habilitado* - crear estos hosts como *Habilitado*
- *Crear deshabilitado* - crear estos hosts como *Deshabilitado*
- *Eliminar* - eliminar estos prototipos de host

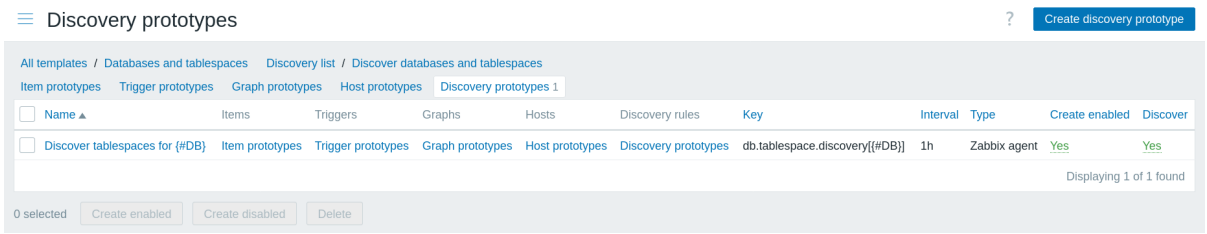
Para usar estas opciones, marque las casillas de verificación delante de los respectivos prototipos de host y, a continuación, haga clic en el botón requerido.

Prototipos de descubrimiento

Descripción general

En esta sección se muestran los prototipos de descubrimiento configurados de una regla de descubrimiento de bajo nivel en el template.

Si el template está vinculado al host, los prototipos de descubrimiento serán la base para crear prototipos de descubrimiento reales de la regla de descubrimiento principal.



Datos mostrados:

Column	Description
Name	Nombre del prototipo de descubrimiento, mostrado como un enlace azul. Al hacer clic en el nombre, se abre el formulario de configuración del prototipo de descubrimiento. Si el prototipo de descubrimiento pertenece a un template vinculado, el nombre del template se muestra antes del nombre del prototipo de descubrimiento como un enlace gris. Al hacer clic en el enlace del template, se abre la lista de prototipos de descubrimiento en el nivel del template vinculado.
Items	Se muestra un enlace a la lista de prototipos de item. El número de prototipos de item existentes se muestra en gris.
Triggers	Se muestra un enlace a la lista de prototipos de trigger. El número de prototipos de trigger existentes se muestra en gris.
Graphs	Se muestra un enlace a la lista de prototipos de gráficos. El número de prototipos de gráficos existentes se muestra en gris.
Hosts	Se muestra un enlace a la lista de prototipos de host. El número de prototipos de host existentes se muestra en gris.
Discovery rules	Se muestra un enlace a la lista de prototipos de descubrimiento. El número de prototipos de descubrimiento existentes se muestra en gris.
Key	Se muestra la clave del item utilizada para el descubrimiento.
Interval	Se muestra la frecuencia con la que se realiza el descubrimiento.
Type	Se muestra el tipo de item utilizado para el descubrimiento (Zabbix agent, SNMP agent, etc.).
Create enabled	Se muestra el estado de creación del prototipo de descubrimiento: creación habilitada (*Yes*) o creación deshabilitada (*No*). Al hacer clic en el estado, puede cambiarlo de Yes a No y viceversa.
Discover	Se muestra el estado de descubrimiento del prototipo de descubrimiento: descubrir (*Yes*) o no descubrir (*No*). Al hacer clic en el estado, puede cambiarlo de Yes a No y viceversa.

Para configurar un nuevo prototipo de descubrimiento, haga clic en el botón *Create discovery prototype* situado en la esquina superior derecha.

Opciones de edición masiva

Los botones debajo de la lista ofrecen algunas opciones de edición masiva:

- *Crear habilitado* - crear estos prototipos de descubrimiento como *Habilitado*
- *Crear deshabilitado* - crear estos prototipos de descubrimiento como *Deshabilitado*
- *Eliminar* - eliminar estos prototipos de descubrimiento

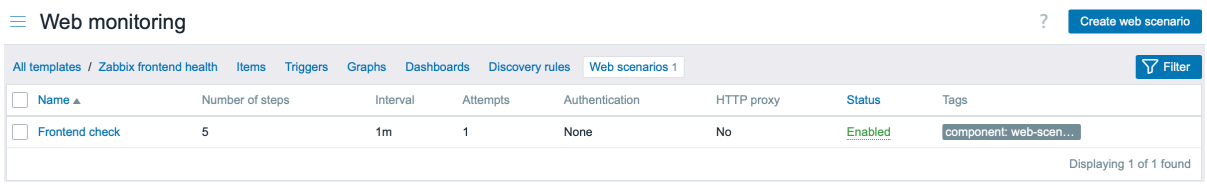
Para usar estas opciones, marque las casillas de verificación delante de los respectivos prototipos de descubrimiento y, a continuación, haga clic en el botón requerido.

Escenarios web

Descripción general

Se puede acceder a la lista de escenarios web de un template desde *Recopilación de datos* > *Templates*, haciendo clic en *Web* para el template correspondiente.

Se muestra una lista de los escenarios web existentes.



Datos mostrados:

Columna	Descripción
Nombre	Nombre del escenario web. Al hacer clic en el nombre del escenario web, se abre el formulario de configuración del escenario web. Si el escenario web se hereda de otro template, el nombre del template se muestra antes del nombre del escenario web como un enlace gris. Al hacer clic en el enlace del template, se abre la lista de escenarios web en ese nivel del template.
Número de pasos	Número de pasos que contiene el escenario.
Intervalo de actualización	Frecuencia con la que se ejecuta el escenario.
Intentos	Número de intentos realizados para ejecutar los pasos del escenario web.
Autenticación	Se muestra el método de autenticación: Basic, NTLM, Kerberos, Digest o None.
Proxy HTTP	Muestra el proxy HTTP o «No» si no se utiliza.
Estado	Se muestra el estado del escenario web: *Enabled* o *Disabled*. Al hacer clic en el estado, puede cambiarlo.
Etiquetas	Se muestran las etiquetas del escenario web. Se pueden mostrar hasta tres etiquetas (pares nombre:valor). Si hay más etiquetas, se muestra un enlace «...» que permite ver todas las etiquetas al pasar el cursor del mouse.

Para configurar un nuevo escenario web, haga clic en el botón *Create web scenario* situado en la esquina superior derecha.

Opciones de edición masiva

Los botones debajo de la lista ofrecen algunas opciones de edición masiva:

- *Enable* - cambia el estado del escenario a *Enabled*
- *Disable* - cambia el estado del escenario a *Disabled*
- *Delete* - elimina los escenarios web

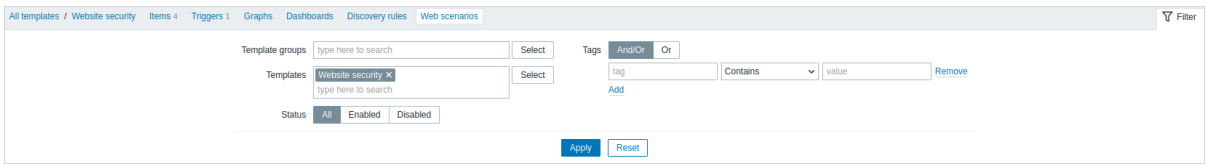
Para usar estas opciones, marque las casillas de verificación delante de los escenarios web correspondientes y, a continuación, haga clic en el botón requerido.

Uso del filtro

Puede usar el filtro para mostrar solo los escenarios que le interesan.

Para mejorar el rendimiento de la búsqueda, los datos se buscan con macros sin resolver.

El enlace *Filter* está disponible encima de la lista de escenarios web. Si hace clic en él, se mostrará un filtro en el que puede filtrar escenarios por grupo de template, template, estado y etiquetas.

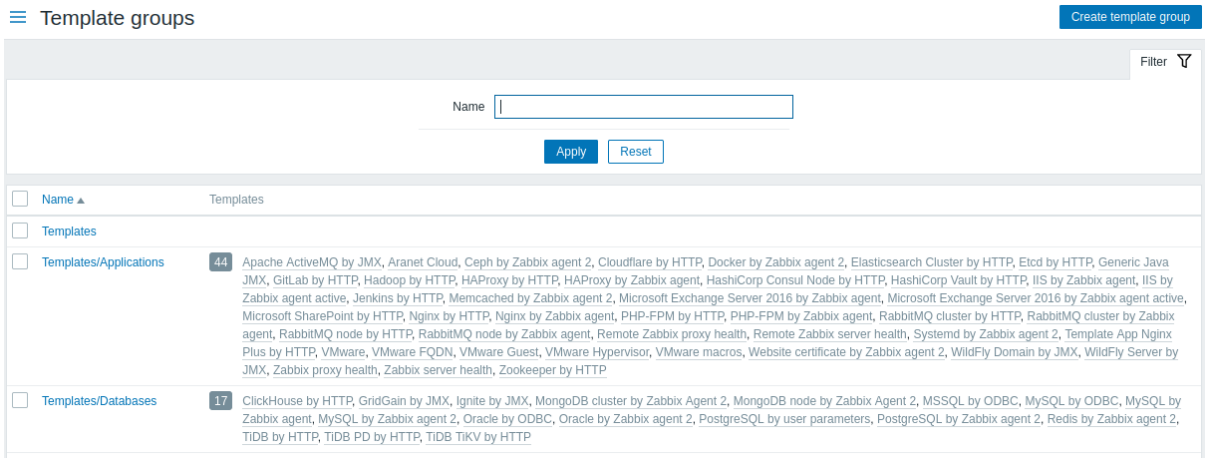


Grupos de template

Descripción general

En la sección *Recopilación de datos* > *Grupos de templates*, los usuarios pueden configurar y mantener grupos de templates.

Se muestra una lista de los grupos de templates existentes con sus detalles. Puede buscar y filtrar los grupos de templates por nombre.



Datos mostrados:

Columna	Descripción
Nombre	Nombre del grupo de templates. Al hacer clic en el nombre del grupo, se abre el formulario de configuración del grupo de templates.
Templates	Número de templates del grupo (mostrado en gris), seguido de la lista de miembros del grupo. Al hacer clic en el nombre de un template, se abre el formulario de configuración del template. Al hacer clic en el número, se abre la lista de templates de este grupo.

Opciones de edición masiva

Para eliminar varios grupos de templates a la vez, marque las casillas de verificación delante de los grupos correspondientes y, a continuación, haga clic en el botón Delete debajo de la lista.

Usando el filtro

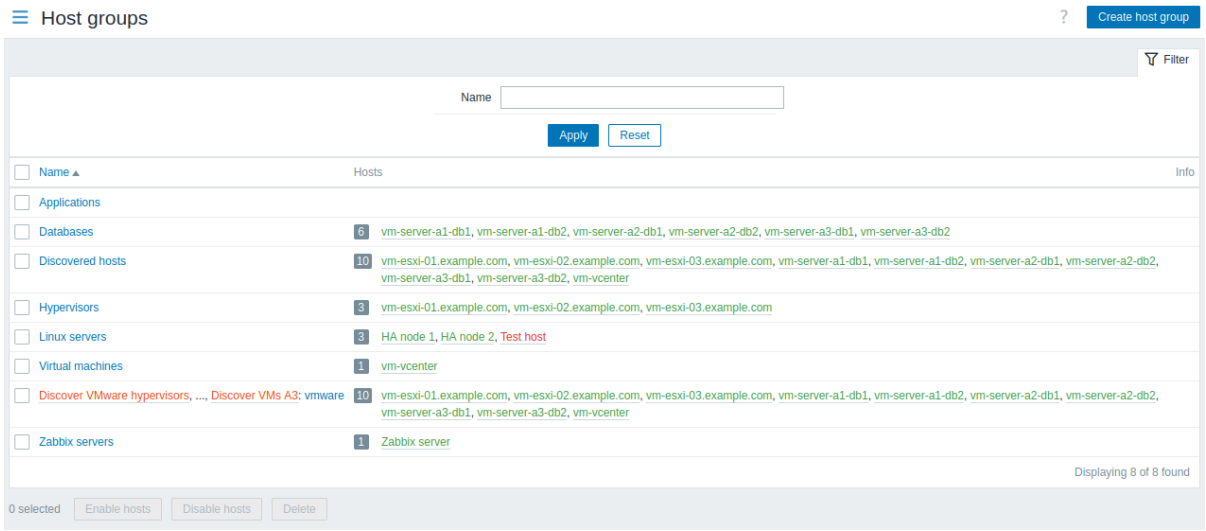
Puede usar el filtro para mostrar solo los grupos de plantillas que le interesan. Para un mejor rendimiento de búsqueda, los datos se buscan con las macros sin resolver.

Grupos de hosts

Descripción general

En la sección *Recopilación de datos* > *Grupos de hosts*, los usuarios pueden configurar y mantener grupos de hosts.

Se muestra una lista de los grupos de hosts existentes con sus detalles. Puede buscar y filtrar grupos de hosts por nombre.



Datos mostrados:

Columna	Descripción
Nombre	Nombre del grupo de hosts. Al hacer clic en el nombre del grupo, se abre el formulario de configuración del grupo de hosts. Los grupos de hosts descubiertos se muestran con los nombres de las reglas de descubrimiento de bajo nivel como prefijo. Al hacer clic en el nombre de la regla LLD, se abre el formulario de configuración del prototipo de host. Tenga en cuenta que los grupos de hosts descubiertos se eliminan cuando se eliminan todas las reglas LLD que los descubrieron.
Hosts	Número de hosts del grupo (mostrado en gris), seguido de la lista de miembros del grupo. Al hacer clic en el nombre de un host, se abre el formulario de configuración del host. Al hacer clic en el número, en la lista completa de hosts se filtran aquellos que pertenecen al grupo.
Información	Se muestra la información de error, si existe, relacionada con el grupo de hosts.

Opciones de edición masiva

Los botones debajo de la lista ofrecen algunas opciones de edición masiva:

- *Enable hosts* - cambia el estado de todos los hosts del grupo a "Monitored"
- *Disable hosts* - cambia el estado de todos los hosts del grupo a "Not monitored"
- *Delete* - elimina los grupos de hosts

Para usar estas opciones, marque las casillas de verificación delante de los grupos de hosts correspondientes y, a continuación, haga clic en el botón necesario.

Usando el filtro

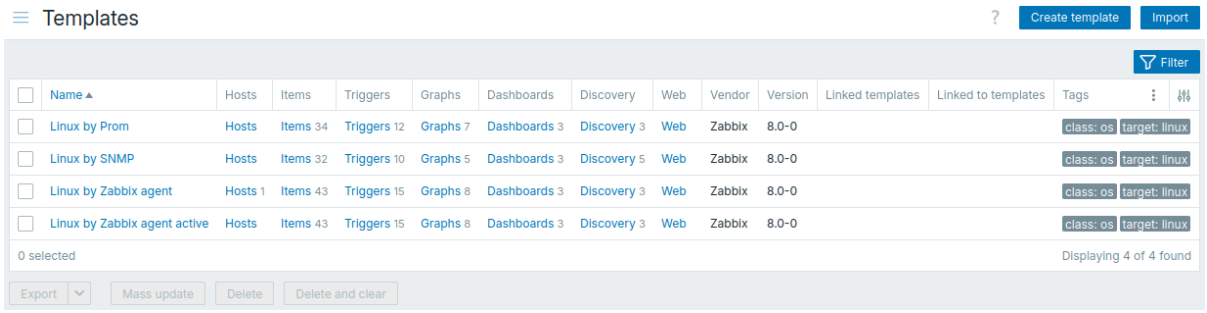
Puede usar el filtro para mostrar solo los grupos de equipos que le interesan. Para un mejor rendimiento de búsqueda, los datos se buscan con las macros sin resolver.

Plantillas

Descripción general

En la sección *Recopilación de datos* > *Templates*, los usuarios pueden configurar y mantener templates.

Se muestra una lista de los templates existentes con sus detalles.



Los detalles de los templates se muestran en una tabla personalizable. El botón *Personalizar tabla*, situado en el encabezado de la tabla, permite mostrar, ocultar o reordenar columnas, así como restablecer el diseño predeterminado. También puede cambiar el tamaño de cada columna arrastrando su borde derecho; haga doble clic en el borde para restablecerlo. Todas las personalizaciones se aplican inmediatamente y se recuerdan siempre para el usuario actual.

Están disponibles las siguientes columnas:

Columna	Descripción
Nombre	Nombre del template. Al hacer clic en el nombre del template, se abre el formulario de configuración del template.
Hosts	Número de hosts editables a los que está vinculado el template; los hosts de solo lectura no se incluyen. Al hacer clic en *Hosts*, se abre la lista de hosts con un filtro que muestra únicamente los hosts vinculados al template.

Columna	Descripción
Entidades (Items, Triggers, Gráficos, Dashboards, Descubrimiento, Web)	Número de las entidades correspondientes en el template (se muestra en gris). Al hacer clic en el nombre de la entidad, en la lista completa de esa entidad se filtran las que pertenecen al template.
Templates vinculados	Templates que están vinculados al template.
Vinculado a templates	Templates a los que el template está vinculado.
Proveedor, *Versión*	Proveedor y versión del template; se muestran si la configuración del template contiene esta información, y únicamente para templates incluidos de fábrica, templates importados o templates modificados mediante la API de Template. Para los templates incluidos de fábrica, la versión se muestra de la siguiente manera: versión principal de Zabbix, delimitador (”-”), número de revisión (que aumenta con cada nueva versión del template y se restablece con cada versión principal de Zabbix). Por ejemplo, 7.0-0, 7.0-3, 8.0-0, 8.0-3.
Etiquetas	Se muestran las etiquetas del template, con las macros sin resolver. Al pasar el cursor sobre el icono de tres puntos o hacer clic en él después de las etiquetas mostradas, se abre una ventana emergente con todas las etiquetas. Al hacer clic en el botón *Opciones* del encabezado de la columna, puede establecer el número de etiquetas mostradas (1–3), elegir cómo se muestran los nombres de las etiquetas (completos, abreviados o ninguno), establecer la prioridad de visualización de las etiquetas o duplicar la columna (las columnas duplicadas se pueden renombrar o eliminar).
Valor de etiqueta	Se muestran los valores de las etiquetas del template, con las macros sin resolver. Al hacer clic en el botón *Opciones* del encabezado de la columna, puede renombrar la columna, elegir qué valores de etiquetas mostrar o duplicar la columna (las columnas duplicadas se pueden renombrar o eliminar).
Texto personalizado	Se muestra texto personalizado, valores de macros o una combinación de ambos. Al hacer clic en el botón *Opciones* del encabezado de la columna, puede renombrar la columna, introducir el texto personalizado o las macros, o duplicar la columna (las columnas duplicadas se pueden renombrar o eliminar). Macros compatibles: `{HOST.ID}`, `{HOST.HOST}`, `{HOST.NAME}`, `{HOST.DESCRIPTION}` y macros de usuario.

Para configurar un nuevo template, haga clic en el botón *Crear template* situado en la esquina superior derecha.

Para importar un template desde un archivo YAML, XML o JSON, haga clic en el botón *Importar* situado en la esquina superior derecha.

Opciones de edición masiva

Los botones debajo de la lista ofrecen algunas opciones de edición masiva:

- *Export* - exportar el template a un archivo YAML, XML o JSON;
- *Mass update* - actualizar varias propiedades de varios templates a la vez;
- *Delete* - eliminar el template dejando sus entidades vinculadas (items, triggers, etc.) en los hosts;
- *Delete and clear* - eliminar el template y sus entidades vinculadas de los hosts.

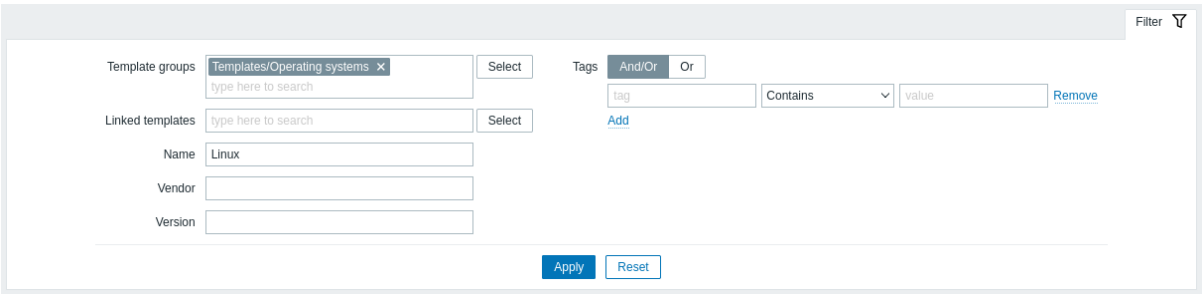
Para usar estas opciones, marque las casillas antes de los respectivos templates y luego haga clic en el botón requerido.

Uso del filtro

Puede usar el filtro para mostrar solo los templates que le interesen.

Para un mejor rendimiento de búsqueda, los datos se buscan con macros sin resolver.

El enlace *Filter* está disponible debajo de los botones *Create template* e *Import*. Si hace clic en él, se mostrará un filtro que le permitirá filtrar templates por grupo de templates, templates vinculados, nombre y etiquetas.



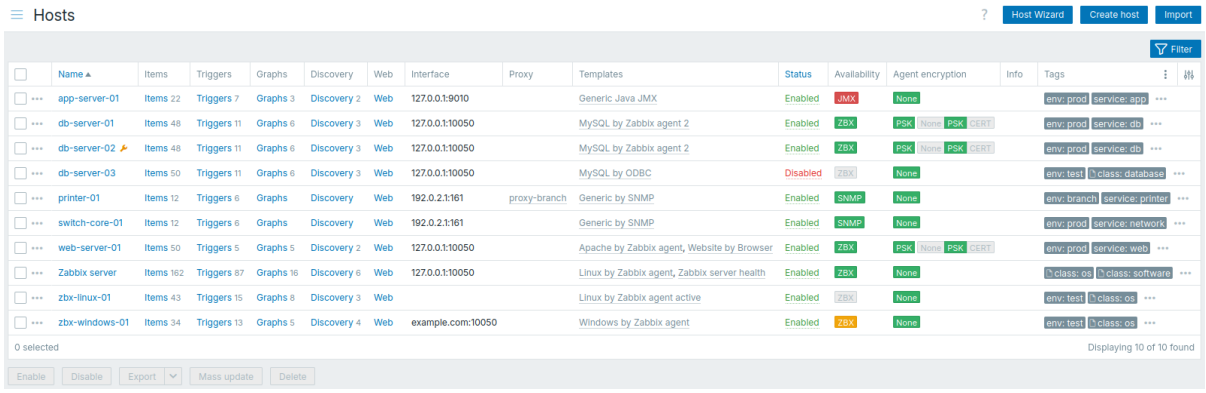
Parameter	Description
Template groups	Filtrar por uno o varios grupos de templates. Especificar un grupo de templates padre selecciona implícitamente todos los grupos anidados.
Linked templates	Filtrar por templates vinculados directamente.
Name	Filtrar por nombre del template.
Vendor	Filtrar por proveedor del template.
Version	Filtrar por versión del template.
Tags	Filtrar por nombre y valor de la etiqueta del template. El filtrado solo es posible por etiquetas a nivel de template (no por las heredadas). Es posible incluir y excluir etiquetas y valores de etiqueta específicos. Se pueden establecer varias condiciones. La coincidencia del nombre de la etiqueta siempre distingue entre mayúsculas y minúsculas. Hay varios operadores disponibles para cada condición: **Exists** - incluir los nombres de etiqueta especificados; **Equals** - incluir los nombres y valores de etiqueta especificados (distingue entre mayúsculas y minúsculas); **Contains** - incluir los nombres de etiqueta especificados cuyos valores de etiqueta contengan la cadena introducida (coincidencia de subcadena, sin distinción entre mayúsculas y minúsculas); **Does not exist** - excluir los nombres de etiqueta especificados; **Does not equal** - excluir los nombres y valores de etiqueta especificados (distingue entre mayúsculas y minúsculas); **Does not contain** - excluir los nombres de etiqueta especificados cuyos valores de etiqueta contengan la cadena introducida (coincidencia de subcadena, sin distinción entre mayúsculas y minúsculas). Hay dos tipos de cálculo para las condiciones: **And/Or** - todas las condiciones deben cumplirse; las condiciones que tengan el mismo nombre de etiqueta se agruparán mediante la condición Or; **Or** - basta con que se cumpla una condición.

Hosts

Descripción general

En la sección *Recopilación de datos* > *Hosts*, los usuarios pueden configurar y mantener hosts.

Se muestra una lista de los hosts existentes con sus detalles.



Los detalles de los hosts se muestran en una tabla personalizable. El botón *Personalizar tabla*, situado en el encabezado de la tabla, permite mostrar, ocultar o reordenar columnas, así como restablecer el diseño predeterminado. También puede cambiar el tamaño de cada columna arrastrando su borde derecho; haga doble clic en el borde para restablecerlo. Todas las personalizaciones se aplican inmediatamente y se recuerdan siempre para el usuario actual.

Están disponibles las siguientes columnas:

Columna	Descripción
Nombre	Nombre del host. Al hacer clic en el nombre del host, se abre el formulario de configuración del host. Una llave inglesa naranja  después del nombre del host indica que este host está en mantenimiento. Los detalles del mantenimiento se muestran al colocar el puntero del ratón sobre el icono.
Entidades (Items, Triggers, Gráficos, Descubrimiento, Web)	Al hacer clic en el nombre de la entidad, se muestran los items, triggers, etc., del host. El número de las entidades correspondientes se muestra en gris.
Interfaz	Se muestra la interfaz principal del host.
Proxy	En esta columna se muestran los proxy asignados: <Nombre del proxy> - el host es monitorizado por un proxy independiente (incluso si el proxy forma parte de un grupo de proxies); <Nombre del grupo de proxies: nombre del proxy> - el host es monitorizado por un grupo de proxies y el Zabbix server ha asignado un proxy para monitorizar el host; <Nombre del grupo de proxies> - el host es monitorizado por un grupo de proxies sin proxies, o el Zabbix server no ha asignado un proxy para monitorizar el host; Nada - el host no es monitorizado por un proxy ni por un grupo de proxies. Esta columna solo se muestra si la opción de filtro *Monitorizado por* está establecida en «Cualquiera», «Proxy» o «Grupo de proxies».
Templates	Se muestran los templates vinculados al host. Si el template vinculado contiene otros templates, estos se muestran entre paréntesis y separados por comas. Al hacer clic en el nombre de un template, se abre su formulario de configuración.
Estado	Se muestra el estado del host: *Habilitado* o *Deshabilitado*. Al hacer clic en el estado, puede cambiarlo manualmente. Los hosts descubiertos que se han perdido se marcan con un icono de información. El texto de la información emergente proporciona detalles sobre su estado.

Columna	Descripción
Disponibilidad	Se muestra la disponibilidad del host por cada interfaz configurada. Los iconos de disponibilidad representan el estado actual de la interfaz del host en el Zabbix server. Por lo tanto, si deshabilita un host en el frontend, su disponibilidad se actualizará después de que el Zabbix server haya sincronizado los cambios de configuración. Del mismo modo, si habilita un host, su disponibilidad se actualizará después de que el Zabbix server haya sincronizado los cambios de configuración y haya consultado el host. Los iconos de disponibilidad representan únicamente los tipos de interfaz (Agent, SNMP, IPMI, JMX) que están configurados. Al pasar el puntero sobre el icono, se muestra una ventana emergente con una lista de todas las interfaces del mismo tipo, junto con sus detalles, estado y errores. Para la interfaz Agent, la ventana emergente muestra las interfaces (pasivas) y las comprobaciones activas. Si un host solo tiene comprobaciones activas, se muestra el icono de la interfaz Agent aunque el host no tenga configurada una interfaz Agent. La columna está vacía para los hosts sin interfaces. El estado de una única interfaz del host se determina mediante la conexión entre un item habilitado que utiliza la interfaz y el host. El estado puede ser: **Disponible**: la conexión con el host se realizó correctamente; **No disponible**: la conexión con el host no se realizó correctamente (tiempo de espera agotado, problemas de firewall, etc.); **Desconocido**: no se ha intentado establecer la conexión con el host o se desconoce el resultado. Para obtener más información sobre cómo el Zabbix server determina el estado de la interfaz, consulte Estado de interfaz desconocido y Configuración de interfaces de host inalcanzables/no disponibles. El estado de todas las interfaces del host de un mismo tipo (Agent, SNMP, IPMI, JMX) se determina mediante las interfaces utilizadas por al menos un item habilitado. El estado se indica mediante el color del icono: **Verde**: todas las interfaces están disponibles; **Amarillo**: al menos una interfaz no está disponible y al menos una está disponible o tiene un estado desconocido; **Rojo**: ninguna interfaz está disponible; **Gris**: al menos una interfaz tiene un estado desconocido, pero ninguna está no disponible. **Disponibilidad de las comprobaciones activas.** Si hay al menos una comprobación activa habilitada en el host, las comprobaciones activas también afectan a la disponibilidad total de la interfaz Agent, tal como se ha descrito anteriormente. Para determinar la disponibilidad de las comprobaciones activas, se envían mensajes de latido en el hilo de comprobaciones activas del agent. La frecuencia de los mensajes de latido se controla mediante el parámetro `HeartbeatFrequency` en la configuración del agent o agent 2 de Zabbix (60 segundos de forma predeterminada, rango de 0 a 3600). Las comprobaciones activas se consideran no disponibles cuando el latido de la comprobación activa tiene una antigüedad superior a 2 x HeartbeatFrequency segundos. **Nota:** los agentes de Zabbix anteriores a la versión 6.2.x no envían latidos de comprobaciones activas, por lo que la disponibilidad de sus hosts permanece desconocida.
Cifrado del agent	Se muestra el estado del cifrado de las conexiones hacia el host y desde el host: **Ninguno**: sin cifrado; **PSK**: se utiliza una clave precompartida; **Cert**: se utiliza un certificado.
Información	Se muestra la información de error, si existe, relacionada con el host.
Etiquetas	Se muestran las etiquetas del host, con las macros sin resolver. Al pasar el puntero sobre el icono de tres puntos ••• o hacer clic en él después de las etiquetas mostradas, se abre una ventana emergente con todas las etiquetas. Al hacer clic en el botón ⋮ *Opciones* del encabezado de la columna, puede establecer el número de etiquetas mostradas (1–3), elegir cómo se muestran los nombres de las etiquetas (completos, abreviados o ninguno), establecer la prioridad de visualización de las etiquetas o duplicar la columna (las columnas duplicadas se pueden renombrar o eliminar).

Columna	Descripción
Valor de etiqueta	Se muestran los valores de las etiquetas del host, con las macros sin resolver. Al hacer clic en el botón *Opciones* del encabezado de la columna, puede renombrar la columna, elegir qué valores de etiqueta mostrar o duplicar la columna (las columnas duplicadas se pueden renombrar o eliminar).
Texto personalizado	Se muestran textos personalizados, valores de macros o una combinación de ambos. Al hacer clic en el botón *Opciones* del encabezado de la columna, puede renombrar la columna, introducir el texto personalizado o las macros, o duplicar la columna (las columnas duplicadas se pueden renombrar o eliminar). Macros compatibles: `{HOST.*}` (excepto `{HOST.METADATA}`), `{INVENTORY.*}` y macros de usuario.

Para abrir el menú del host, haga clic en el botón *Opciones* situado antes del nombre del host.

Para configurar un host nuevo, haga clic en el botón *Asistente de hosts* o *Crear host* en la esquina superior derecha.

Para importar un host desde un archivo YAML, XML o JSON, haga clic en el botón *Importar* en la esquina superior derecha.

Opciones de edición masiva

Los botones debajo de la lista ofrecen algunas opciones de edición masiva:

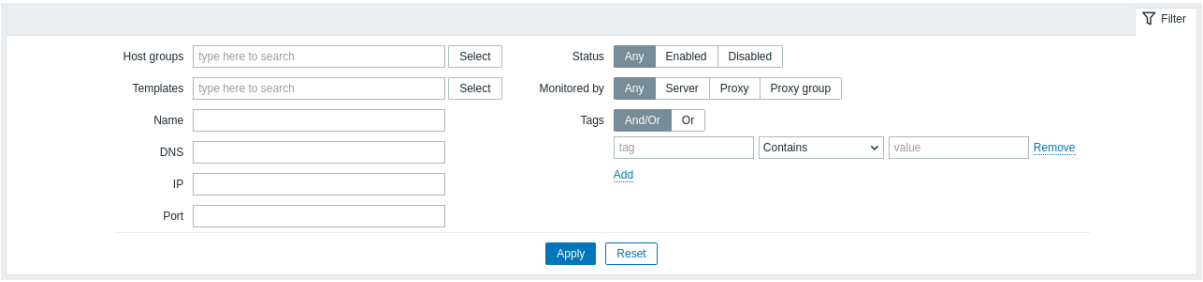
- *Enable* - cambiar el estado del host a *Monitored*;
- *Disable* - cambiar el estado del host a *Not monitored*;
- *Export* - exportar los hosts a un archivo YAML, XML o JSON;
- *Mass update* - actualizar varias propiedades de varios hosts a la vez;
- *Delete* - eliminar los hosts.

Para usar estas opciones, marque las casillas de verificación delante de los hosts correspondientes y luego haga clic en el botón requerido.

Uso del filtro

Puede usar el filtro para mostrar solo los hosts que le interesan. Para un mejor rendimiento de búsqueda, los datos se buscan con las macros sin resolver.

El icono *Filter* está disponible en la esquina superior derecha. Al hacer clic en él, se abrirá un filtro donde puede especificar los criterios de filtrado deseados.



Parameter	Description
Host groups	Filtrar por uno o varios grupos de hosts. Especificar un grupo de hosts padre selecciona implícitamente todos los grupos de hosts anidados.
Templates	Filtrar por templates vinculadas.
Name	Filtrar por el nombre visible del host.
DNS	Filtrar por el nombre DNS.
IP	Filtrar por la dirección IP.
Port	Filtrar por el número de puerto.
Status	Filtrar por el estado del host.
Monitored by	Filtrar hosts que son monitorizados por Zabbix server, proxy o grupo de proxy.
Proxies	Filtrar hosts que son monitorizados por los proxies especificados aquí. Este campo solo está disponible si se selecciona ”Proxy” en el campo *Monitored by*.
Proxy groups	Filtrar hosts que son monitorizados por los grupos de proxy especificados aquí. Este campo solo está disponible si se selecciona ”Proxy group” en el campo *Monitored by*.

Parameter	Description
Tags	Filtrar por el nombre y el valor de la etiqueta del host. Es posible incluir y excluir etiquetas y valores de etiqueta específicos. Se pueden establecer varias condiciones. La coincidencia del nombre de la etiqueta siempre distingue entre mayúsculas y minúsculas. Hay varios operadores disponibles para cada condición: **Exists** - incluir los nombres de etiqueta especificados; **Equals** - incluir los nombres y valores de etiqueta especificados (distingue entre mayúsculas y minúsculas); **Contains** - incluir los nombres de etiqueta especificados cuyos valores de etiqueta contengan la cadena introducida (coincidencia de subcadena, sin distinción entre mayúsculas y minúsculas); **Does not exist** - excluir los nombres de etiqueta especificados; **Does not equal** - excluir los nombres y valores de etiqueta especificados (distingue entre mayúsculas y minúsculas); **Does not contain** - excluir los nombres de etiqueta especificados cuyos valores de etiqueta contengan la cadena introducida (coincidencia de subcadena, sin distinción entre mayúsculas y minúsculas). Hay dos tipos de cálculo para las condiciones: **And/Or** - todas las condiciones deben cumplirse; las condiciones que tengan el mismo nombre de etiqueta se agruparán mediante la condición Or; **Or** - basta con que se cumpla una condición.

Estado de interfaz desconocido

El server de Zabbix muestra el estado ”Unknown” para una interfaz de host (Agent, SNMP, IPMI, JMX) en los siguientes casos:

- El host está deshabilitado.
- El host está configurado para ser monitoreado por proxy, por un proxy diferente o por el server si anteriormente era monitoreado por proxy.
- El host es monitoreado por un proxy que parece estar fuera de línea (no se han recibido actualizaciones del proxy durante el intervalo máximo de latido: 1 hora).
- Todos los items del host con ese tipo de interfaz están deshabilitados.
- No se han configurado pollers para ese tipo de interfaz (por ejemplo, el parámetro de configuración del server `StartAgentPollers` o `StartSNMPPollers` está establecido en 0).

La disponibilidad de la interfaz se establece en ”Unknown” después de la sincronización de la caché de configuración del server de Zabbix.

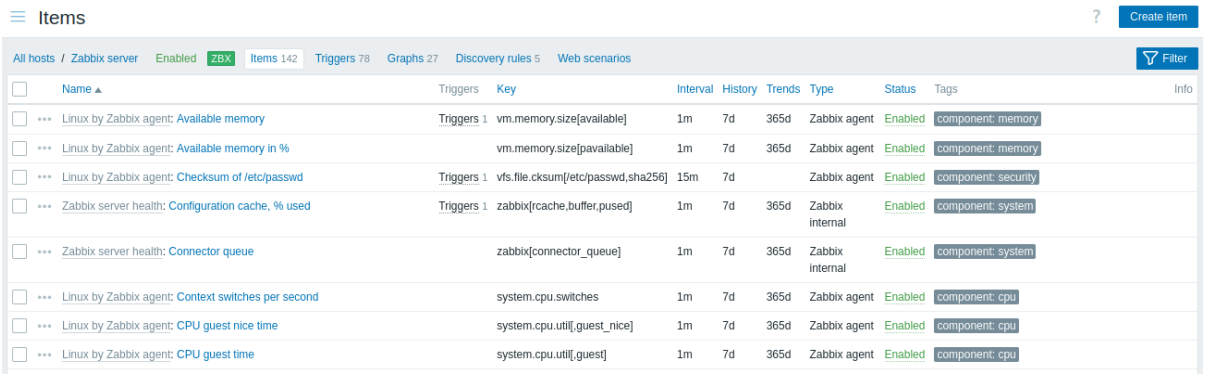
La disponibilidad de la interfaz (Available/Not available) en hosts monitoreados por proxies se restaura después de la sincronización de la caché de configuración del proxy.

Elementos

Descripción general

Se puede acceder a la lista de items de un host desde *Recopilación de datos* > *Hosts*, haciendo clic en *Items* para el host correspondiente.

Se muestra una lista de los items existentes.



Datos mostrados:

Columna	Descripción
Menú del item	Haga clic en el icono de tres puntos para abrir el menú del item.
Host	Host del item. Al hacer clic en el nombre del host, se abre el formulario de configuración del host. Esta columna solo se muestra si se seleccionan varios hosts o ningún host en el filtro.
Nombre	Nombre del item, mostrado como un enlace azul a los detalles del item. Al hacer clic en el enlace del nombre del item, se abre el formulario de configuración del item. Si el item del host pertenece a un template, el nombre del template se muestra antes del nombre del item como un enlace gris. Al hacer clic en el enlace del template, se abre la lista de items en el nivel del template. Si el item se ha creado a partir de un prototipo de item, su nombre está precedido por el nombre de la regla de descubrimiento de bajo nivel, en color naranja. Al hacer clic en el nombre de la regla de descubrimiento, se abre la lista de prototipos de item.
Triggers	Al mover el ratón sobre Triggers, se muestra un cuadro informativo con los triggers asociados al item. El número de triggers se muestra en gris.
Clave	Se muestra la clave del item.
Intervalo	Se muestra la frecuencia de la comprobación. *Nota*: los items pasivos también se pueden comprobar inmediatamente mediante el botón *Ejecutar ahora*.
Historial	Se muestra durante cuántos días se conservará el historial de datos del item.
Tendencias	Se muestra durante cuántos días se conservará el historial de tendencias del item.
Tipo	Se muestra el tipo de item (Zabbix agent, SNMP agent, comprobación simple, etc.).
Estado	Se muestra el estado del item: *Habilitado*, *Deshabilitado* o *No compatible*. Puede cambiar el estado manualmente haciendo clic en él: de Habilitado a Deshabilitado (y viceversa), o de No compatible a Deshabilitado (y viceversa). Los items descubiertos que se han perdido se marcan con un icono de información. El texto de la descripción emergente proporciona detalles sobre su estado.
Etiquetas	Se muestran las etiquetas del item. Se pueden mostrar hasta tres etiquetas (pares nombre:valor). Si hay más etiquetas, se muestra un enlace "..." que permite ver todas las etiquetas al pasar el ratón sobre él.
Información	Si el item funciona correctamente, no se muestra ningún icono en esta columna. En caso de errores, se muestra un icono cuadrado con la letra "i". Pase el ratón sobre el icono para ver una descripción emergente con la descripción del error.

Para configurar un nuevo item, haga clic en el botón *Crear item* situado en la esquina superior derecha.

Opciones de edición masiva

Los botones debajo de la lista ofrecen algunas opciones de edición masiva:

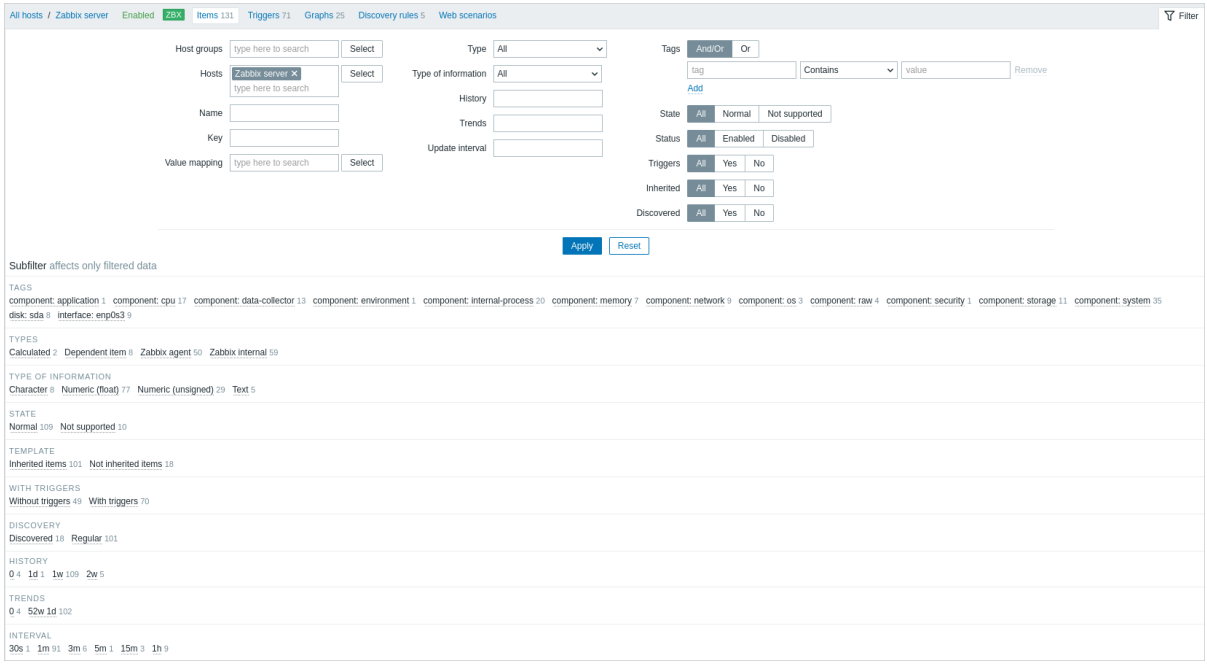
- *Enable* - cambiar el estado del item a *Enabled*
- *Disable* - cambiar el estado del item a *Disabled*
- *Execute now* - ejecutar inmediatamente una comprobación de nuevos valores del item. Compatible solo con comprobaciones **pasivas** (consulte más detalles). Tenga en cuenta que, al comprobar los valores inmediatamente, la caché de configuración no se actualiza, por lo que los valores no reflejarán cambios muy recientes en la configuración del item.
- *Clear history and trends* - eliminar los datos de historial y tendencias de los items.
- *Copy* - copiar los items a otros hosts o templates.
- *Mass update* - actualizar varias propiedades de varios items a la vez.
- *Delete* - eliminar los items.

Para usar estas opciones, marque las casillas antes de los items correspondientes y luego haga clic en el botón requerido.

Uso del filtro

Puede usar el filtro para mostrar solo los items que le interesan. Para un mejor rendimiento de búsqueda, los datos se buscan con macros sin resolver.

El icono *Filter* está disponible en la esquina superior derecha. Al hacer clic en él se abrirá un filtro donde puede especificar los criterios de filtrado deseados.

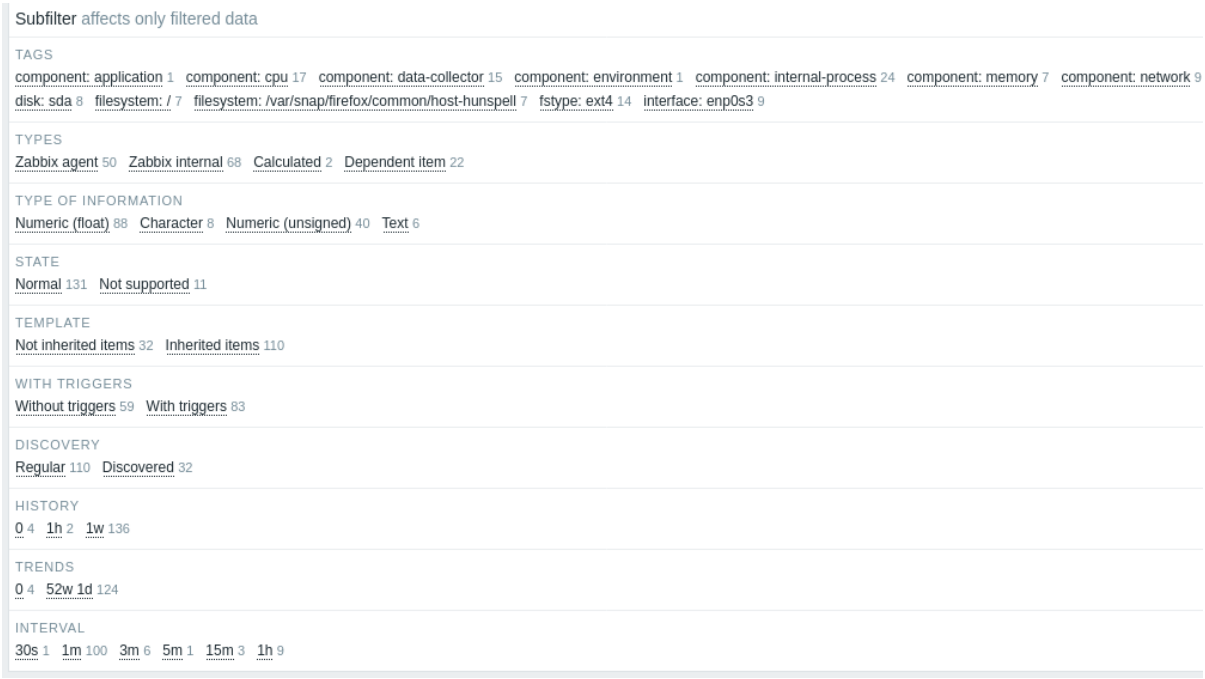


Parameter	Description
Host groups	Filtrar por uno o más grupos de host. Al especificar un grupo de host padre, se seleccionan implícitamente todos los grupos de host anidados. No se pueden seleccionar grupos de host que contengan solo templates.
Hosts	Filtrar por uno o más hosts.
Name	Filtrar por nombre del item.
Key	Filtrar por la clave del item.
Value mapping	Filtrar por el mapa de valores utilizado. Este parámetro no se muestra si la opción *Hosts* está vacía.
Type	Filtrar por tipo de item (Zabbix agent, SNMP agent, etc.).
Type of information	Filtrar por tipo de información (entero sin signo, flotante, etc.).
History	Filtrar por cuánto tiempo se conserva el historial del item.
Trends	Filtrar por cuánto tiempo se conservan las tendencias del item.
Update interval	Filtrar por el intervalo de actualización del item.
Tags	Especifique etiquetas para limitar el número de items mostrados. Es posible incluir y excluir etiquetas y valores de etiqueta específicos. Se pueden establecer varias condiciones. La coincidencia del nombre de la etiqueta siempre distingue entre mayúsculas y minúsculas. Hay varios operadores disponibles para cada condición: **Exists** - incluir los nombres de etiqueta especificados **Equals** - incluir los nombres y valores de etiqueta especificados (distingue entre mayúsculas y minúsculas) **Contains** - incluir los nombres de etiqueta especificados cuyos valores de etiqueta contengan la cadena introducida (coincidencia de subcadena, no distingue entre mayúsculas y minúsculas) **Does not exist** - excluir los nombres de etiqueta especificados **Does not equal** - excluir los nombres y valores de etiqueta especificados (distingue entre mayúsculas y minúsculas) **Does not contain** - excluir los nombres de etiqueta especificados cuyos valores de etiqueta contengan la cadena introducida (coincidencia de subcadena, no distingue entre mayúsculas y minúsculas) Hay dos tipos de cálculo para las condiciones: **And/Or** - deben cumplirse todas las condiciones; las condiciones que tengan el mismo nombre de etiqueta se agruparán mediante la condición Or **Or** - basta con que se cumpla una condición
State	Filtrar por estado del item: *Normal* o *Not supported*.
Status	Filtrar por estado del item: *Enabled* o *Disabled*.
Triggers	Filtrar items con (o sin) triggers.
Inherited	Filtrar items heredados (o no heredados) de un template.
Discovery	Filtrar items descubiertos (o no descubiertos) mediante la detección de bajo nivel.

Uso del subfiltro

El subfiltro permite modificar aún más el filtrado aplicado por el filtro principal.

Contiene enlaces en los que se puede hacer clic para acceder rápidamente a los elementos relacionados. Los elementos están relacionados por una entidad común: etiqueta, tipo de item, estado del item, intervalo de actualización del item, etc. Al hacer clic en una entidad, esta se resalta con un fondo gris y los elementos se filtran inmediatamente (no es necesario hacer clic en *Aplicar* en el filtro principal). Al hacer clic en otra entidad, esta se añade a los resultados filtrados. Al volver a hacer clic en la entidad, se elimina el filtrado.



Los subfiltros se generan en función de los datos filtrados, que están limitados a 1000 registros. Si desea ver más registros en el subfiltro, debe aumentar el valor del parámetro *Límite para los resultados de búsqueda y filtrado* (en *Administración > General > GUI*).

A diferencia del filtro principal, el subfiltro se actualiza con cada solicitud de actualización de la tabla para mostrar siempre información actualizada sobre las opciones de filtrado disponibles y sus contadores.

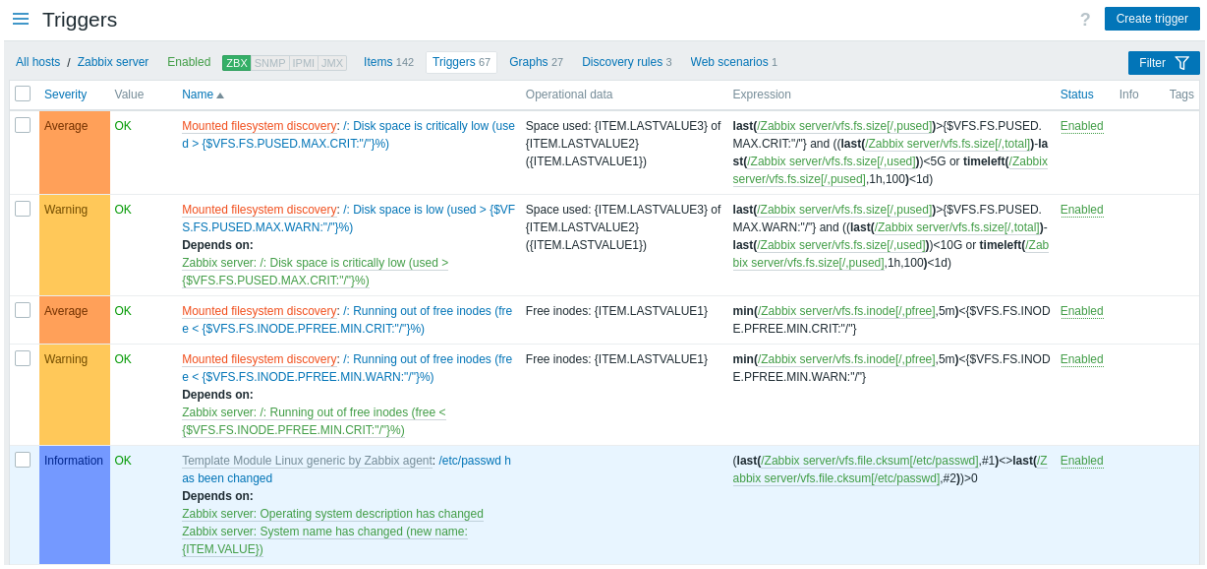
El número de entidades mostradas está limitado a 100 horizontalmente. Si hay más, se muestra un icono de tres puntos al final; no se puede hacer clic en él.

El número situado junto a cada entidad en la que se puede hacer clic indica el número de items agrupados en ella (según los resultados del filtro principal). Al hacer clic en una entidad, los números de las demás entidades disponibles se muestran con un signo más, que indica cuántos items se pueden añadir a la selección actual. Las entidades sin items no se muestran, a menos que se hayan seleccionado previamente en el subfiltro.

Triggers

Descripción general

Se puede acceder a la lista de triggers de un host desde *Recopilación de datos > Hosts*, haciendo clic en *Triggers* para el host correspondiente.



Datos mostrados:

Columna	Descripción
Gravedad	La gravedad del trigger se muestra tanto mediante el nombre como mediante el color de fondo de la celda.
Valor	Se muestra el estado del trigger: **OK** - estado OK **PROBLEM** - estado Problem
Host	Host del trigger. Al hacer clic en el nombre del host, se abre el formulario de configuración del host. Esta columna solo se muestra si se seleccionan varios hosts o ningún host en el filtro.
Nombre	Nombre del trigger, mostrado como un enlace azul a los detalles del trigger. Al hacer clic en el enlace del nombre del trigger, se abre el formulario de configuración del trigger. Si el trigger pertenece a un template, el nombre del template se muestra antes del nombre del trigger como un enlace gris. Al hacer clic en el enlace del template, se abre la lista de triggers en el nivel del template. Si el trigger se ha creado a partir de un prototipo de trigger, su nombre va precedido por el nombre de la regla de descubrimiento de bajo nivel, en naranja. Al hacer clic en el nombre de la regla de descubrimiento, se abre la lista de prototipos de trigger.
Datos operativos	Definición de los datos operativos del trigger, que contiene cadenas arbitrarias y macros que se resolverán dinámicamente en *Monitorización* > *Problemas*.
Expresión	Se muestra la expresión del trigger. La parte host-item de la expresión se muestra como un enlace que lleva al formulario de configuración del item.
Estado	Se muestra el estado del trigger: *Habilitado*, *Deshabilitado* o *Desconocido*. Al hacer clic en el estado, puede cambiarlo manualmente: de Habilitado a Deshabilitado (y viceversa); de Desconocido a Deshabilitado (y viceversa). Los problemas de un trigger deshabilitado ya no se muestran en el frontend, pero no se eliminan. Los triggers descubiertos que se han perdido se marcan con un icono de información. El texto de la información emergente proporciona detalles sobre su estado.
Información	Si todo funciona correctamente, no se muestra ningún icono en esta columna. En caso de errores, se muestra un icono cuadrado con la letra ”i”. Pase el cursor sobre el icono para ver una información emergente con la descripción del error.
Etiquetas	Si un trigger contiene etiquetas, en esta columna se muestran el nombre y el valor de la etiqueta.

Para configurar un nuevo trigger, haga clic en el botón *Crear trigger* situado en la esquina superior derecha.

Opciones de edición masiva

Los botones debajo de la lista ofrecen algunas opciones de edición masiva:

- *Enable* - cambia el estado del trigger a *Enabled*.
- *Disable* - cambia el estado del trigger a *Disabled*.
- *Copy* - copia los triggers a otros hosts o templates.
- *Mass update* - actualiza varias propiedades de un número de triggers a la vez.

- *Delete* - elimina los triggers.

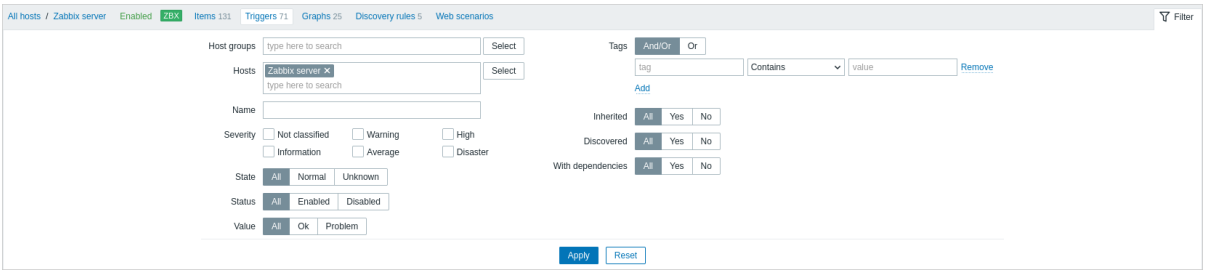
Para usar estas opciones, marque las casillas antes de los triggers correspondientes y luego haga clic en el botón requerido.

Uso del filtro

Puede usar el filtro para mostrar solo los triggers que le interesan.

Para un mejor rendimiento de búsqueda, los datos se buscan con macros sin resolver.

El icono *Filter* está disponible en la esquina superior derecha. Al hacer clic en él, se abrirá un filtro donde puede especificar los criterios de filtrado deseados.



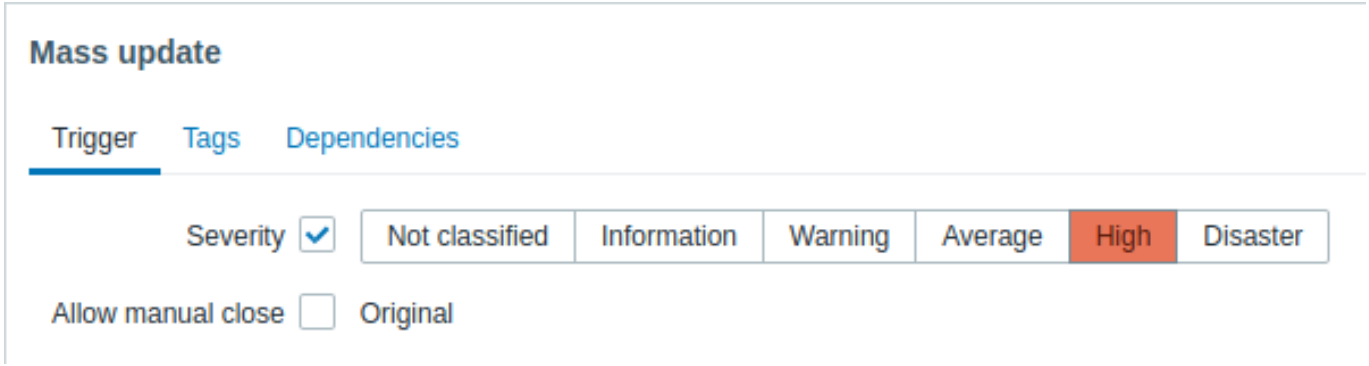
Parameter	Description
Host groups	Filtrar por uno o varios grupos de hosts. Especificar un grupo de hosts padre selecciona implícitamente todos los grupos de hosts anidados. No se pueden seleccionar grupos de hosts que contengan solo templates.
Hosts	Filtrar por uno o varios hosts. Si ya se han seleccionado grupos de hosts arriba, la selección de hosts se limita a esos grupos.
Name	Filtrar por nombre del trigger.
Severity	Seleccione para filtrar por una o varias severidades del trigger.
State	Filtrar por el estado de la expresión del trigger (Normal/Unknown).
Status	Filtrar por el estado de activación del trigger (Enabled/Disabled).
Value	Filtrar por el valor del trigger.
Tags	Filtrar por nombre y valor de la etiqueta del trigger. Es posible incluir y excluir etiquetas y valores de etiqueta específicos. Se pueden establecer varias condiciones. La coincidencia del nombre de la etiqueta siempre distingue entre mayúsculas y minúsculas. Hay varios operadores disponibles para cada condición: **Exists** - incluir los nombres de etiqueta especificados **Equals** - incluir los nombres y valores de etiqueta especificados (distingue entre mayúsculas y minúsculas) **Contains** - incluir los nombres de etiqueta especificados cuyos valores de etiqueta contengan la cadena introducida (coincidencia de subcadena, no distingue entre mayúsculas y minúsculas) **Does not exist** - excluir los nombres de etiqueta especificados **Does not equal** - excluir los nombres y valores de etiqueta especificados (distingue entre mayúsculas y minúsculas) **Does not contain** - excluir los nombres de etiqueta especificados cuyos valores de etiqueta contengan la cadena introducida (coincidencia de subcadena, no distingue entre mayúsculas y minúsculas) Hay dos tipos de cálculo para las condiciones: **And/Or** - todas las condiciones deben cumplirse; las condiciones que tengan el mismo nombre de etiqueta se agruparán mediante la condición Or **Or** - basta con que se cumpla una condición Se admiten macros y funciones de macro tanto en los campos de nombre de etiqueta como de valor de etiqueta.
Inherited	Filtrar triggers heredados (o no heredados) de un template.
Discovered	Filtrar triggers descubiertos (o no descubiertos) por el descubrimiento de bajo nivel.
With dependencies	Filtrar triggers con (o sin) dependencias.

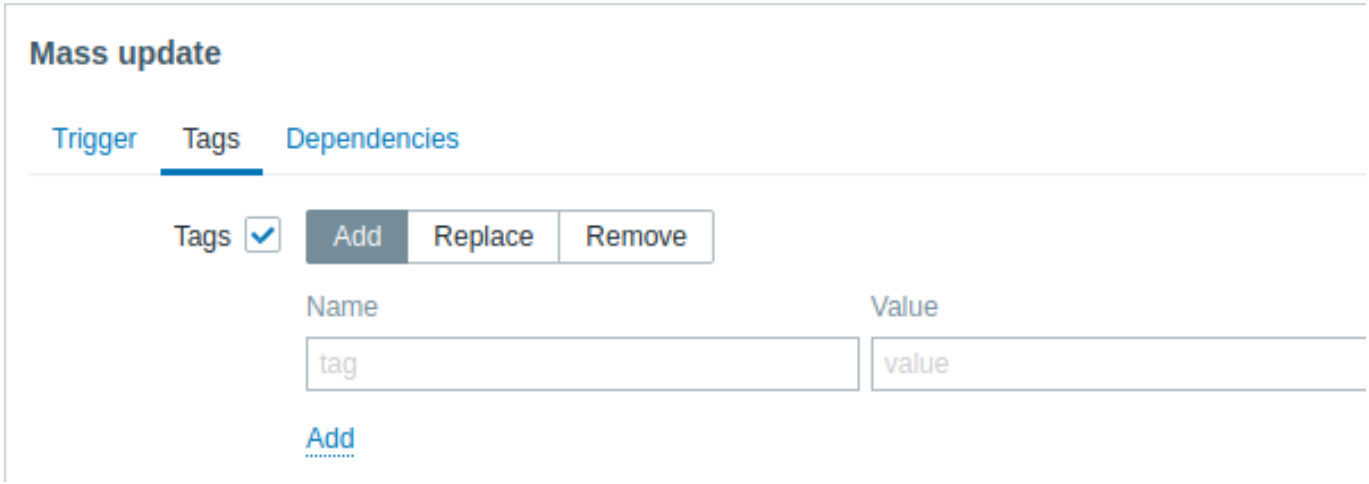
Uso de la actualización masiva

Con la actualización masiva puede cambiar algún atributo de varios triggers a la vez, sin necesidad de abrir cada trigger individual para editarlo.

Para actualizar varios triggers de forma masiva, haga lo siguiente:

- Marque las casillas de los triggers que desea actualizar en la lista
- Haga clic en *Actualización masiva* debajo de la lista
- Vaya a la pestaña con los atributos necesarios (*Trigger*, *Etiquetas* o *Dependencias*)
- Marque las casillas de los atributos que desea actualizar

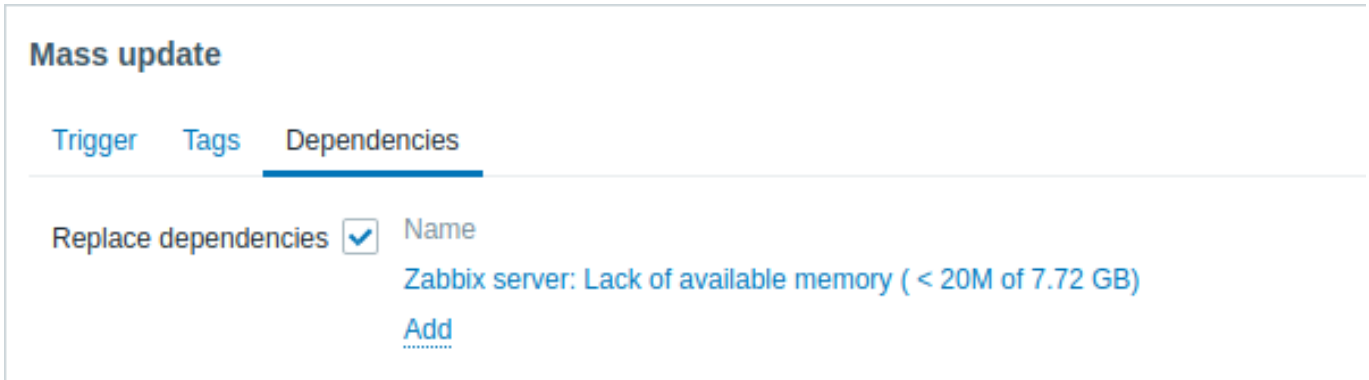




Las siguientes opciones están disponibles al seleccionar el botón correspondiente para actualizar las etiquetas:

- *Añadir*: añade nuevas etiquetas a los triggers;
- *Reemplazar*: elimina todas las etiquetas existentes del trigger y las reemplaza por las especificadas a continuación;
- *Eliminar*: elimina las etiquetas especificadas de los triggers.

Tenga en cuenta que las etiquetas con el mismo nombre pero con valores diferentes no se consideran «duplicadas» y pueden añadirse al mismo trigger.



Reemplazar dependencias: elimina todas las dependencias existentes del trigger y las reemplaza por las especificadas.

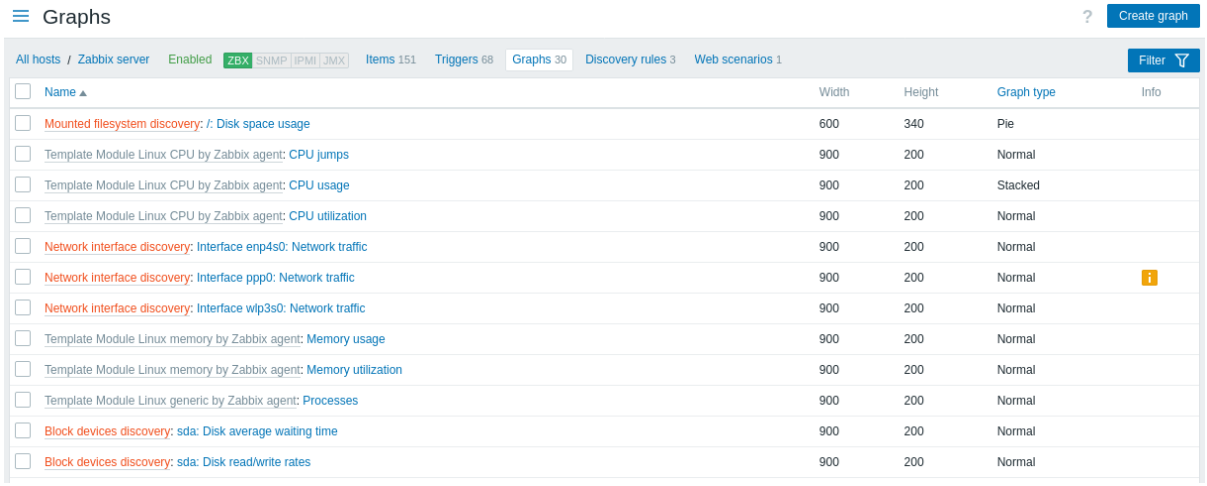
Haga clic en *Actualizar* para aplicar los cambios.

Gráficos

Descripción general

Se puede acceder a la lista de gráficos personalizados de un host desde *Recopilación de datos* > *Hosts*, haciendo clic en *Gráficos* para el host correspondiente.

Se muestra una lista de los gráficos existentes.



Datos mostrados:

Columna	Descripción
Host	Host del gráfico. Al hacer clic en el nombre del host, se abre el formulario de configuración del host. Esta columna solo se muestra si se seleccionan varios hosts o ningún host en el filtro.
Nombre	Nombre del gráfico personalizado, mostrado como un enlace azul a los detalles del gráfico. Al hacer clic en el enlace del nombre del gráfico, se abre el formulario de configuración del gráfico. Si el gráfico del host pertenece a un template, el nombre del template se muestra antes del nombre del gráfico como un enlace gris. Al hacer clic en el enlace del template, se abre la lista de gráficos en el nivel del template. Si el gráfico se ha creado a partir de un prototipo de gráfico, su nombre va precedido por el nombre de la regla de descubrimiento de bajo nivel, en naranja. Al hacer clic en el nombre de la regla de descubrimiento, se abre la lista de prototipos de gráficos.
Ancho	Se muestra el ancho del gráfico.
Alto	Se muestra el alto del gráfico.
Tipo de gráfico	Se muestra el tipo de gráfico: *Normal*, *Apilado*, *Circular* o *Explotado*.
Información	Si el gráfico funciona correctamente, no se muestra ningún icono en esta columna. En caso de errores, se muestra un icono cuadrado con la letra "i". Pase el cursor sobre el icono para ver una descripción emergente con la descripción del error.

Para configurar un gráfico nuevo, haga clic en el botón *Crear gráfico* situado en la esquina superior derecha.

Opciones de edición masiva

Los botones debajo de la lista ofrecen algunas opciones de edición masiva:

- *Copiar* - copiar los gráficos a otros hosts o templates
- *Eliminar* - eliminar los gráficos

Para usar estas opciones, marque las casillas de verificación delante de los gráficos correspondientes y, a continuación, haga clic en el botón requerido.

Usando el filtro

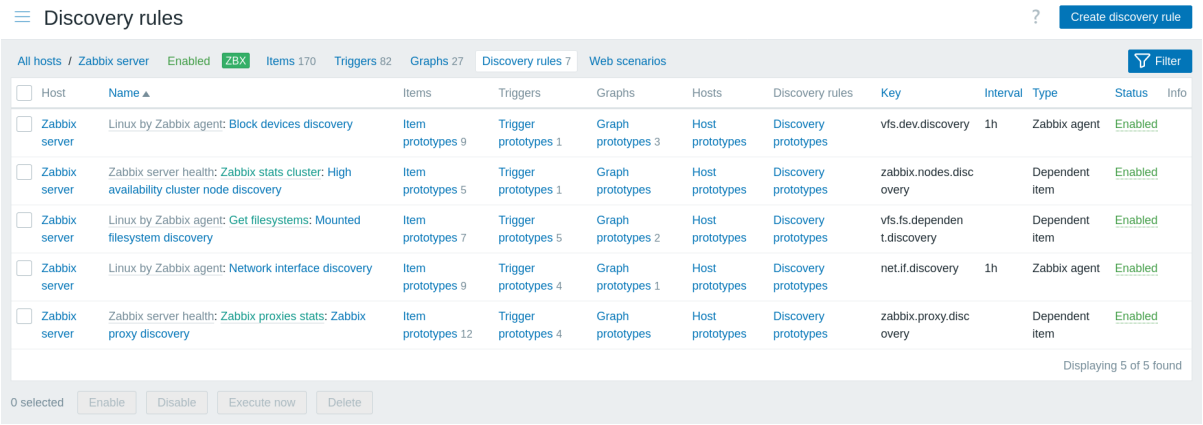
Puede filtrar los gráficos por grupo de equipos y equipo. Para un mejor rendimiento de búsqueda, los datos se buscan con macros sin resolver.

Reglas de descubrimiento

Descripción general

Se puede acceder a la lista de reglas de descubrimiento de bajo nivel de un host desde *Recopilación de datos* > *Hosts*, haciendo clic en *Descubrimiento* para el host correspondiente.

Se muestra una lista de las reglas de descubrimiento de bajo nivel existentes. También es posible ver todas las reglas de descubrimiento independientemente del host, o todas las reglas de descubrimiento de un grupo de hosts específico, cambiando la configuración de los filtros.



Datos mostrados:

Columna	Descripción
Host	Se muestra el nombre visible del host. Al hacer clic en el nombre del host, se abre el formulario de configuración del host. Si no existe un nombre visible del host, se muestra el nombre técnico del host.
Nombre	Nombre de la regla, mostrado como un enlace azul. Al hacer clic en el nombre de la regla, se abre el formulario de configuración de la regla de descubrimiento de bajo nivel. Si la regla de descubrimiento pertenece a un template, el nombre del template se muestra antes del nombre de la regla como un enlace gris. Al hacer clic en el enlace del template, se abre la lista de reglas en el nivel del template.
Items	Se muestra un enlace a la lista de prototipos de item. El número de prototipos de item existentes se muestra en gris.
Triggers	Se muestra un enlace a la lista de prototipos de trigger. El número de prototipos de trigger existentes se muestra en gris.
Gráficos	Se muestra un enlace a la lista de prototipos de gráficos. El número de prototipos de gráficos existentes se muestra en gris.
Hosts	Se muestra un enlace a la lista de prototipos de host. El número de prototipos de host existentes se muestra en gris.
Clave	Se muestra la clave del item utilizada para el descubrimiento.
Intervalo	Se muestra la frecuencia con la que se realiza el descubrimiento. *Nota*: el descubrimiento también se puede realizar inmediatamente pulsando el botón *Ejecutar ahora* situado debajo de la lista.
Tipo	Se muestra el tipo de item utilizado para el descubrimiento (Zabbix agent, SNMP agent, etc.).
Estado	Se muestra el estado de la regla de descubrimiento: *Habilitada*, *Deshabilitada* o *No compatible*. Al hacer clic en el estado, puede cambiarlo: de Habilitada a Deshabilitada (y viceversa); de No compatible a Deshabilitada (y viceversa).
Información	Si todo funciona correctamente, no se muestra ningún icono en esta columna. En caso de errores, se muestra un icono cuadrado con la letra "i". Pase el cursor sobre el icono para ver una descripción emergente con la descripción del error.

Para configurar una nueva regla de descubrimiento de bajo nivel, haga clic en el botón *Crear regla de descubrimiento* situado en la esquina superior derecha.

Opciones de edición masiva

Los botones debajo de la lista ofrecen algunas opciones de edición masiva:

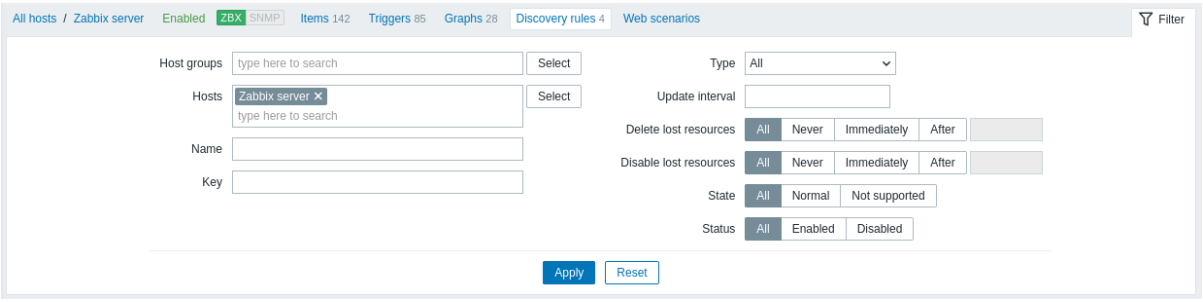
- *Enable* - cambia el estado de la regla de descubrimiento de bajo nivel a *Enabled*.
- *Disable* - cambia el estado de la regla de descubrimiento de bajo nivel a *Disabled*.
- *Execute now* - realiza el descubrimiento basado en las reglas de descubrimiento inmediatamente. Consulte more details. Tenga en cuenta que, al realizar el descubrimiento inmediatamente, la caché de configuración no se actualiza, por lo que el resultado no reflejará cambios muy recientes en la configuración de la regla de descubrimiento.
- *Delete* - elimina las reglas de descubrimiento de bajo nivel.

Para usar estas opciones, marque las casillas antes de las respectivas reglas de descubrimiento y luego haga clic en el botón requerido.

Uso del filtro

Puede usar el filtro para mostrar solo las reglas de descubrimiento que le interesen. Para mejorar el rendimiento de la búsqueda, los datos se buscan con las macros sin resolver.

El enlace *Filter* está disponible encima de la lista de reglas de descubrimiento. Si hace clic en él, estará disponible un filtro con el que puede filtrar las reglas de descubrimiento por grupo de host, host, nombre, clave de item, tipo de item y otros parámetros.

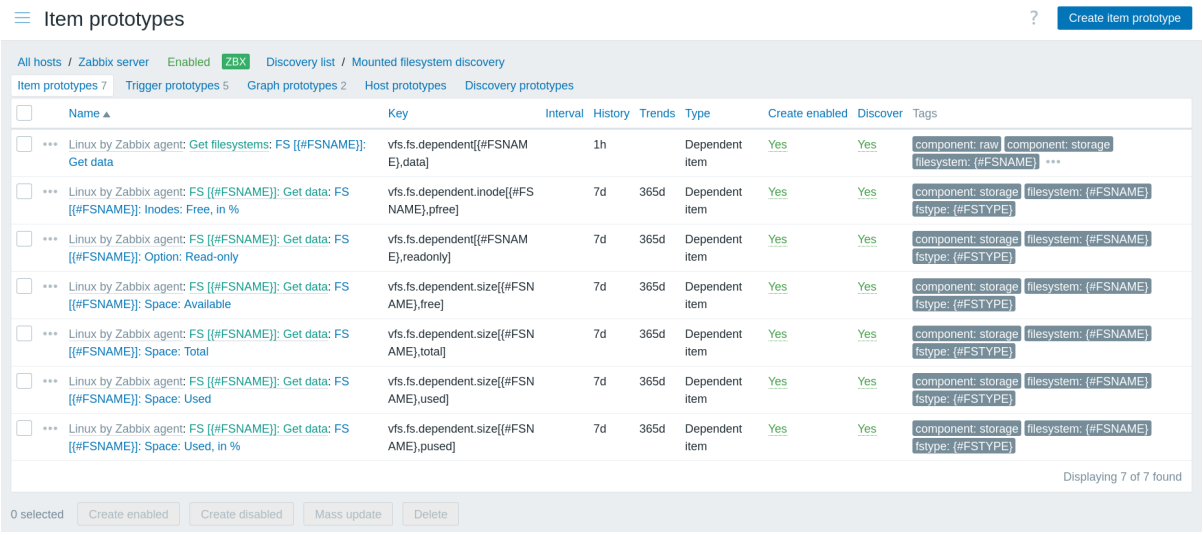


Parameter	Description
Host groups	Filtrar por uno o más grupos de host. Especificar un grupo de host padre selecciona implícitamente todos los grupos de host anidados.
Hosts	Filtrar por uno o más hosts.
Name	Filtrar por el nombre de la regla de descubrimiento.
Key	Filtrar por la clave del item de descubrimiento.
Type	Filtrar por el tipo de item de descubrimiento.
Update interval	Filtrar por el intervalo de actualización. No disponible para Zabbix trapper y items dependientes.
Delete lost resources	Filtrar por el período de *Delete lost resources*.
Disable lost resources	Filtrar por el período de *Disable lost resources*.
SNMP OID	Filtrar por SNMP OID. Solo disponible si *SNMP agent* está seleccionado como tipo.
State	Filtrar por el estado de la regla de descubrimiento (All/Normal/Not supported).
Status	Filtrar por el estado de la regla de descubrimiento (All/Enabled/Disabled).

Prototipos de item

Descripción general

En esta sección se muestran los prototipos de item de una regla de descubrimiento de bajo nivel en el host. Los prototipos de item son la base de los items reales del host que se crean durante el descubrimiento de bajo nivel.



Datos mostrados:

Column	Description
Name	Se muestra el nombre del prototipo de item como un enlace azul. Al hacer clic en el nombre, se abre el formulario de configuración del prototipo de item. Si el prototipo de item pertenece a un template, el nombre del template se muestra antes del nombre de la regla como un enlace gris. Al hacer clic en el enlace del template, se abre la lista de prototipos de item en el nivel del template. Si el prototipo de item es del tipo ”Dependent item”, el nombre del item maestro se muestra antes del nombre del prototipo de item en color verde.
Key	Se muestra la clave del prototipo de item.
Interval	Se muestra la frecuencia de la comprobación.
History	Se muestra durante cuántos días se conservará el historial de datos del item.
Trends	Se muestra durante cuántos días se conservará el historial de tendencias del item.
Type	Se muestra el tipo de prototipo de item (Zabbix agent, SNMP agent, comprobación simple, etc.).
Create enabled	Crea el item basándose en este prototipo como: **Yes** - habilitado **No** - deshabilitado. Puede cambiar entre ’Yes’ y ’No’ haciendo clic en ellos.
Discover	Descubre el item basándose en este prototipo: **Yes** - descubrir **No** - no descubrir. Puede cambiar entre ’Yes’ y ’No’ haciendo clic en ellos.
Tags	Se muestran las etiquetas del prototipo de item.

Para configurar un nuevo prototipo de item, haga clic en el botón *Create item prototype* situado en la esquina superior derecha.

Opciones de edición masiva

Los botones debajo de la lista ofrecen algunas opciones de edición masiva:

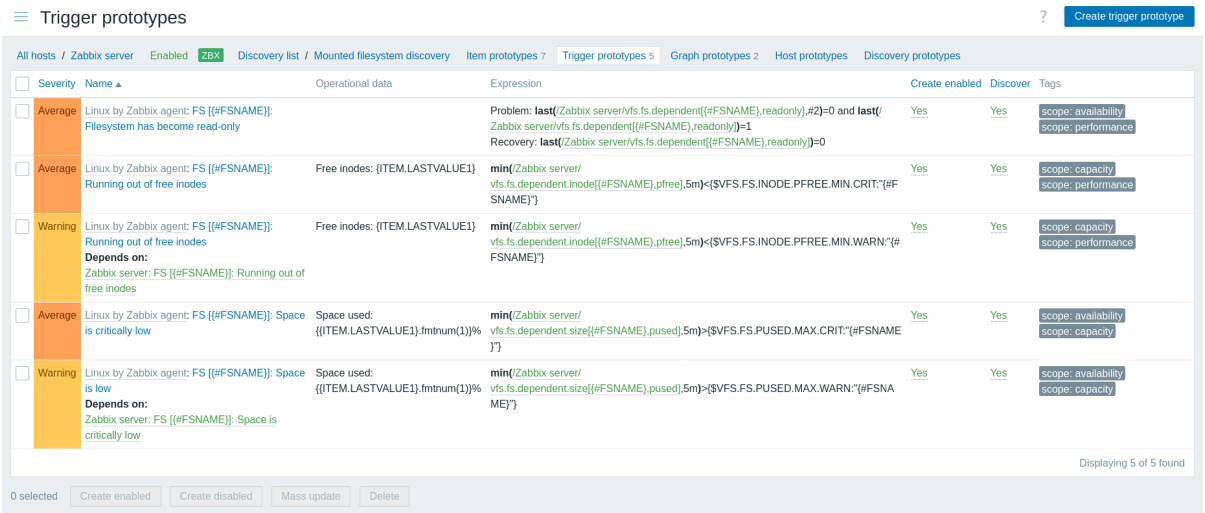
- *Crear habilitado* - crear estos items como *Habilitado*
- *Crear deshabilitado* - crear estos items como *Deshabilitado*
- *Actualización masiva* - actualizar masivamente estos prototipos de item
- *Eliminar* - eliminar estos prototipos de item

Para usar estas opciones, marque las casillas de verificación delante de los respectivos prototipos de item y luego haga clic en el botón requerido.

Prototipos de trigger

Descripción general

En esta sección se muestran los prototipos de trigger de una regla de descubrimiento de bajo nivel en el host. Los prototipos de trigger son la base de los triggers reales del host que se crean durante el descubrimiento de bajo nivel.



Datos mostrados:

Columna	Descripción
Nombre	Nombre del prototipo de trigger, mostrado como un enlace azul. Al hacer clic en el nombre, se abre el formulario de configuración del prototipo de trigger. Si el prototipo de trigger pertenece a un template vinculado, el nombre del template se muestra antes del nombre del trigger como un enlace gris. Al hacer clic en el enlace del template, se abre la lista de prototipos de trigger en el nivel del template vinculado.
Datos operativos	Se muestra el formato de los datos operativos del trigger, que contiene cadenas arbitrarias y macros que se resolverán dinámicamente en *Monitoring* > *Problems*.
Creación habilitada	Crear el trigger basándose en este prototipo como: **Sí**: habilitado **No**: deshabilitado. Puede cambiar entre «Sí» y «No» haciendo clic en ellos.
Descubrir	Descubrir el trigger basándose en este prototipo: **Sí**: descubrir **No**: no descubrir. Puede cambiar entre «Sí» y «No» haciendo clic en ellos.
Etiquetas	Se muestran las etiquetas del prototipo de trigger.

Para configurar un nuevo prototipo de trigger, haga clic en el botón *Crear prototipo de trigger* en la esquina superior derecha.

Opciones de edición masiva

Los botones debajo de la lista ofrecen algunas opciones de edición masiva:

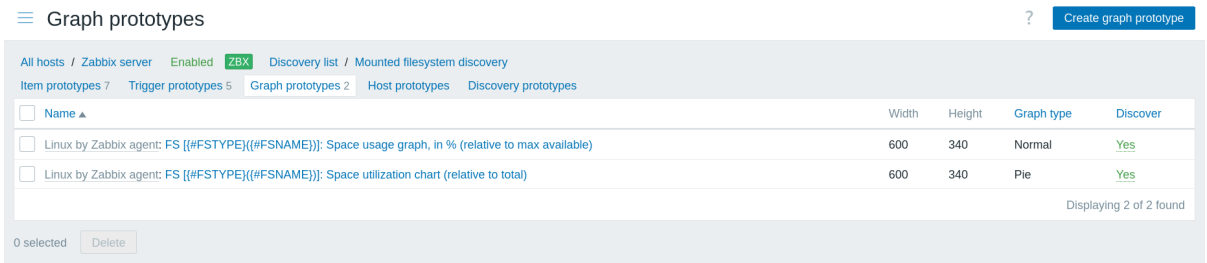
- *Crear habilitado* - crear estos triggers como *Habilitado*
- *Crear deshabilitado* - crear estos triggers como *Deshabilitado*
- *Actualización masiva* - actualizar masivamente estos prototipos de trigger
- *Eliminar* - eliminar estos prototipos de trigger

Para usar estas opciones, marque las casillas de verificación delante de los respectivos prototipos de trigger y, a continuación, haga clic en el botón requerido.

Prototipos de gráficos

Descripción general

En esta sección se muestran los prototipos de gráficos de una regla de descubrimiento de bajo nivel en el host. Los prototipos de gráficos son la base de los gráficos reales del host que se crean durante el descubrimiento de bajo nivel.



Datos mostrados:

Column	Description
Name	Nombre del prototipo de gráfico, mostrado como un enlace azul. Al hacer clic en el nombre, se abre el formulario de configuración del prototipo de gráfico. Si el prototipo de gráfico pertenece a un template vinculado, el nombre del template se muestra antes del nombre del gráfico como un enlace gris. Al hacer clic en el enlace del template, se abre la lista de prototipos de gráficos en el nivel del template vinculado.
Width	Se muestra el ancho del prototipo de gráfico.
Height	Se muestra la altura del prototipo de gráfico.
Type	Se muestra el tipo de prototipo de gráfico: *Normal*, *Apilado*, *Circular* o *Explotado*.
Discover	Descubrir el gráfico basándose en este prototipo: **Yes**: descubrir **No**: no descubrir. Puede cambiar entre «Yes» y «No» haciendo clic en ellas.

Para configurar un nuevo prototipo de gráfico, haga clic en el botón *Create graph prototype* situado en la esquina superior derecha.

Opciones de edición masiva

Los botones debajo de la lista ofrecen algunas opciones de edición masiva:

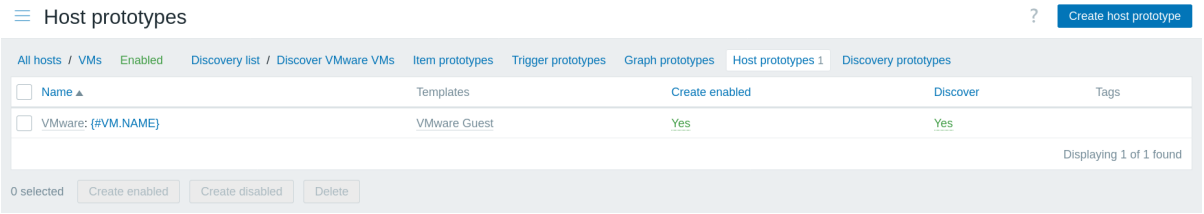
- *Eliminar* - elimina estos prototipos de gráfico

Para usar estas opciones, marque las casillas de verificación delante de los respectivos prototipos de gráfico y, a continuación, haga clic en el botón requerido.

Prototipos de host

Descripción general

En esta sección se muestran los prototipos de host de una regla de descubrimiento de bajo nivel en el host. Los prototipos de host son la base de los hosts reales que se crean durante el descubrimiento de bajo nivel.



Datos mostrados:

Column	Description
Name	Nombre del prototipo de host, mostrado como un enlace azul. Al hacer clic en el nombre, se abre el formulario de configuración del prototipo de host. Si el prototipo de host pertenece a un template vinculado, el nombre del template se muestra antes del nombre del host como un enlace gris. Al hacer clic en el enlace del template, se abre la lista de prototipos de host en el nivel del template vinculado.
Templates	Se muestran los templates del prototipo de host.
Create enabled	Crear el host basándose en este prototipo como: **Yes** - habilitado **No** - deshabilitado. Puede cambiar entre «Yes» y «No» haciendo clic en ellos.
Discover	Descubrir el host basándose en este prototipo: **Yes** - descubrir **No** - no descubrir. Puede cambiar entre «Yes» y «No» haciendo clic en ellos.
Tags	Se muestran las etiquetas del prototipo de host.

Para configurar un nuevo prototipo de host, haga clic en el botón *Create host prototype* situado en la esquina superior derecha.

Opciones de edición masiva

Los botones debajo de la lista ofrecen algunas opciones de edición masiva:

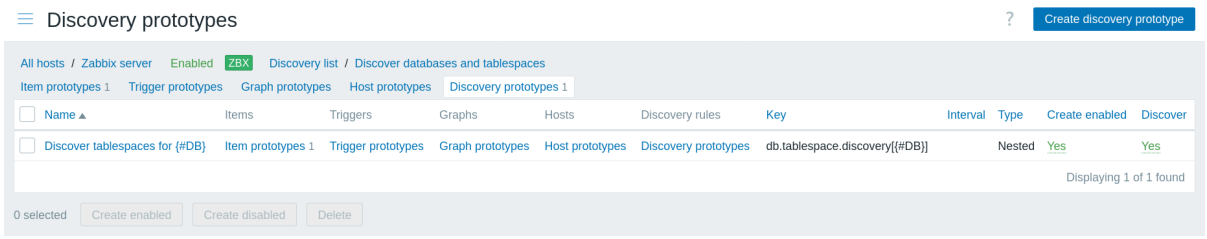
- *Crear habilitado* - crear estos hosts como *Habilitados*
- *Crear deshabilitado* - crear estos hosts como *Deshabilitados*
- *Eliminar* - eliminar estos prototipos de host

Para utilizar estas opciones, marque las casillas de verificación antes de los respectivos prototipos de host y luego haga clic en el botón requerido.

Prototipos de descubrimiento

Descripción general

En esta sección se muestran los prototipos de descubrimiento de una regla de descubrimiento de bajo nivel en el host. Los prototipos de descubrimiento son reglas de descubrimiento anidadas de la regla de descubrimiento principal.



Datos mostrados:

Column	Description
Name	Se muestra el nombre del prototipo de descubrimiento como un enlace azul. Al hacer clic en el nombre, se abre el formulario de configuración del prototipo de descubrimiento. Si el prototipo de descubrimiento pertenece a un template vinculado, el nombre del template se muestra antes del nombre del host como un enlace gris. Al hacer clic en el enlace del template, se abre la lista de prototipos de descubrimiento en el nivel del template vinculado.
Items	Se muestra un enlace a la lista de prototipos de item. El número de prototipos de item existentes se muestra en gris.
Triggers	Se muestra un enlace a la lista de prototipos de trigger. El número de prototipos de trigger existentes se muestra en gris.
Graphs	Se muestra un enlace a la lista de prototipos de gráficos. El número de prototipos de gráficos existentes se muestra en gris.
Hosts	Se muestra un enlace a la lista de prototipos de host. El número de prototipos de host existentes se muestra en gris.
Discovery rules	Se muestra un enlace a la lista de prototipos de reglas de descubrimiento. El número de prototipos de reglas de descubrimiento existentes se muestra en gris.
Key	Se muestra la clave del item utilizada para el descubrimiento.
Interval	Se muestra la frecuencia con la que se realiza el descubrimiento. *Nota*: el descubrimiento también se puede realizar inmediatamente haciendo clic en el botón *Execute now* situado debajo de la lista.
Type	Se muestra el tipo de item utilizado para el descubrimiento (Zabbix agent, SNMP agent, etc.).
Create enabled	Se muestra el estado de creación del prototipo de descubrimiento: creación habilitada (*Yes*) o creación deshabilitada (*No*). Al hacer clic en el estado, puede cambiarlo de Yes a No (y viceversa).
Discover	Se muestra el estado de descubrimiento del prototipo de descubrimiento: descubrir (*Yes*) o no descubrir (*No*). Al hacer clic en el estado, puede cambiarlo de Yes a No (y viceversa).

Para configurar un nuevo prototipo de descubrimiento, haga clic en el botón *Create discovery prototype* situado en la esquina superior derecha.

Opciones de edición masiva

Los botones debajo de la lista ofrecen algunas opciones de edición masiva:

- *Crear habilitado* - crear estos prototipos de descubrimiento como *Habilitado*
- *Crear deshabilitado* - crear estos prototipos de descubrimiento como *Deshabilitado*
- *Eliminar* - eliminar estos prototipos de descubrimiento

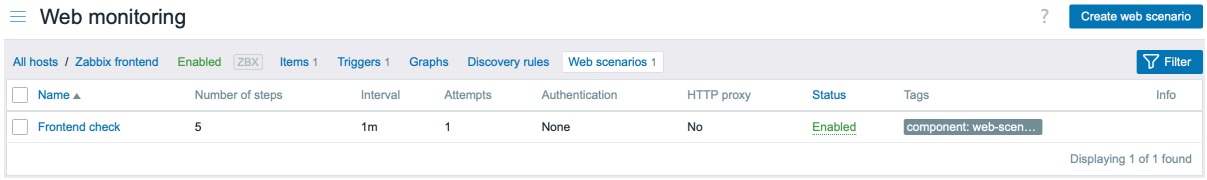
Para usar estas opciones, marque las casillas de verificación delante de los respectivos prototipos de descubrimiento y, a continuación, haga clic en el botón requerido.

Escenarios web

Descripción general

La lista de escenarios web de un host se puede abrir desde *Recopilación de datos* > *Hosts*, haciendo clic en *Web* para el host correspondiente.

Se muestra una lista de los escenarios web existentes.



Datos mostrados:

Columna	Descripción
Nombre	Nombre del escenario web. Al hacer clic en el nombre del escenario web, se abre el formulario de configuración del escenario web. Si el escenario web del host pertenece a un template, el nombre del template se muestra antes del nombre del escenario web como un enlace gris. Al hacer clic en el enlace del template, se abrirá la lista de escenarios web en el nivel del template.
Número de pasos	Número de pasos que contiene el escenario.
Intervalo de actualización	Frecuencia con la que se ejecuta el escenario.
Intentos	Número de intentos realizados para ejecutar los pasos del escenario web.
Autenticación	Se muestra el método de autenticación: Basic, NTLM, Kerberos, Digest o None.
Proxy HTTP	Muestra el proxy HTTP o «No» si no se utiliza.
Estado	Se muestra el estado del escenario web: *Habilitado* o *Deshabilitado*. Al hacer clic en el estado, puede cambiarlo.
Etiquetas	Se muestran las etiquetas del escenario web. Se pueden mostrar hasta tres etiquetas (pares nombre:valor). Si hay más etiquetas, se muestra un enlace «...» que permite ver todas las etiquetas al pasar el ratón por encima.
Información	Si todo funciona correctamente, no se muestra ningún icono en esta columna. En caso de errores, se muestra un icono cuadrado con la letra «i». Pase el ratón por encima del icono para ver una descripción del error en una información emergente.

Para configurar un nuevo escenario web, haga clic en el botón *Crear escenario web* situado en la esquina superior derecha.

Opciones de edición masiva

Los botones debajo de la lista ofrecen algunas opciones de edición masiva:

- *Enable* - cambia el estado del escenario a *Enabled*
- *Disable* - cambia el estado del escenario a *Disabled*
- *Clear history and trends* - borra los datos de historial y tendencias de los escenarios
- *Delete* - elimina los escenarios web

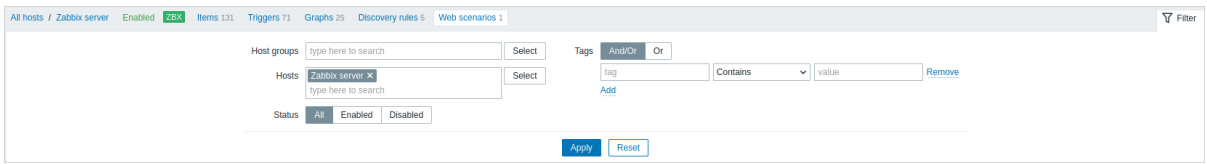
Para usar estas opciones, marque las casillas de verificación delante de los respectivos escenarios web y luego haga clic en el botón requerido.

Uso del filtro

Puede usar el filtro para mostrar solo los escenarios que le interesen.

Para un mejor rendimiento de búsqueda, los datos se buscan con macros sin resolver.

El enlace *Filter* está disponible encima de la lista de escenarios web. Al hacer clic en él, se abrirá un filtro donde puede especificar los criterios de filtrado deseados.



Parameter	Description
Host groups	Filtrar por uno o más grupos de hosts. Al especificar un grupo de hosts padre, se seleccionan implícitamente todos los grupos de hosts anidados.
Hosts	Filtrar por uno o más hosts. Si ya se han seleccionado grupos de hosts arriba, la selección de hosts se limita a esos grupos.
Status	Filtrar por el estado del escenario web.

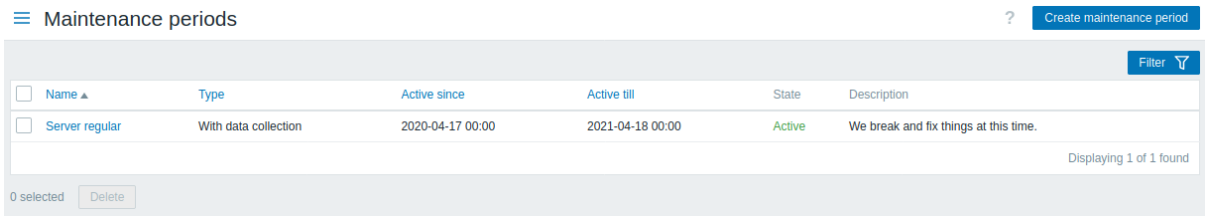
Parameter	Description
Tags	Filtrar por el nombre y el valor de la etiqueta del escenario web. Es posible incluir y excluir etiquetas y valores de etiqueta específicos. Se pueden establecer varias condiciones. La coincidencia del nombre de la etiqueta siempre distingue entre mayúsculas y minúsculas. Hay varios operadores disponibles para cada condición: **Exists** - incluir los nombres de etiqueta especificados; **Equals** - incluir los nombres y valores de etiqueta especificados (distingue entre mayúsculas y minúsculas); **Contains** - incluir los nombres de etiqueta especificados cuyos valores de etiqueta contengan la cadena introducida (coincidencia de subcadena, sin distinguir entre mayúsculas y minúsculas); **Does not exist** - excluir los nombres de etiqueta especificados; **Does not equal** - excluir los nombres y valores de etiqueta especificados (distingue entre mayúsculas y minúsculas); **Does not contain** - excluir los nombres de etiqueta especificados cuyos valores de etiqueta contengan la cadena introducida (coincidencia de subcadena, sin distinguir entre mayúsculas y minúsculas). Hay dos tipos de cálculo para las condiciones: **And/Or** - deben cumplirse todas las condiciones; las condiciones que tengan el mismo nombre de etiqueta se agruparán mediante la condición Or; **Or** - basta con que se cumpla una condición.

Mantenimiento

Descripción general

En la sección *Recopilación de datos* > *Mantenimiento*, los usuarios pueden configurar y mantener períodos de mantenimiento para los hosts.

Se muestra una lista de los períodos de mantenimiento existentes con sus detalles.



Datos mostrados:

Columna	Descripción
Nombre	Nombre del período de mantenimiento. Al hacer clic en el nombre del período de mantenimiento, se abre el formulario de configuración del período de mantenimiento.
Tipo	Se muestra el tipo de mantenimiento: *Con recopilación de datos* o *Sin recopilación de datos*
Activo desde	Fecha y hora en que la ejecución de los períodos de mantenimiento pasa a estar activa. Nota: esta hora no activa un período de mantenimiento; los períodos de mantenimiento deben establecerse por separado.
Activo hasta	Fecha y hora en que la ejecución de los períodos de mantenimiento deja de estar activa.
Estado	Estado del período de mantenimiento: **Próximo**: se activará pronto **Activo**: está activo **Caducado**: ya no está activo
Descripción	Se muestra la descripción del período de mantenimiento.

Para configurar un nuevo período de mantenimiento, haga clic en el botón *Crear período de mantenimiento* situado en la esquina superior derecha.

Opciones de edición masiva

Un botón debajo de la lista ofrece una opción de edición masiva:

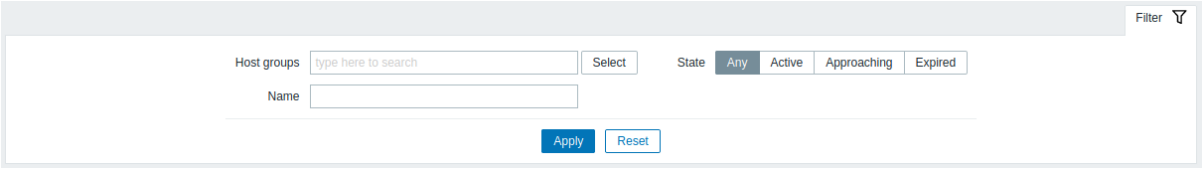
- *Eliminar* - elimina los periodos de mantenimiento

Para usar esta opción, marque las casillas de verificación delante de los respectivos periodos de mantenimiento y haga clic en *Eliminar*.

Uso del filtro

Puede usar el filtro para mostrar solo los periodos de mantenimiento que le interesen. Para un mejor rendimiento de búsqueda, los datos se buscan con las macros sin resolver.

El enlace *Filter* está disponible encima de la lista de periodos de mantenimiento. Si hace clic en él, se mostrará un filtro que le permitirá filtrar los periodos de mantenimiento por grupo de host, nombre y estado.



Cálculo de colas durante el mantenimiento

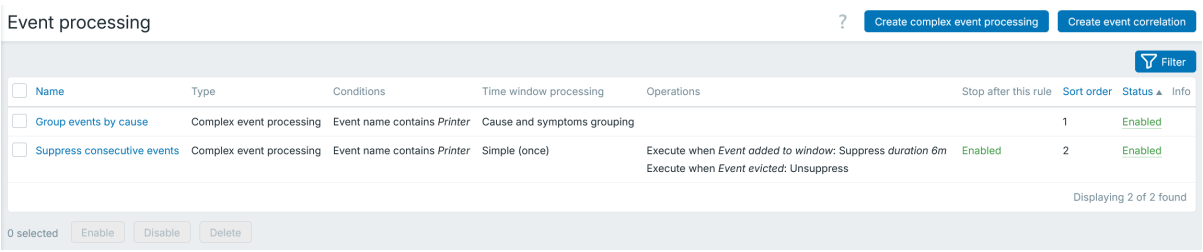
Attention:
El proxy de Zabbix no tiene conocimiento de los períodos de mantenimiento; consulte Calculation of queues during maintenance para obtener más detalles.

Event processing

Overview

In the *Data collection* > *Event processing* section users can configure and maintain:

- Complex event processing rules (CEP rules)
- Global event correlation rules (deprecated)



Displayed data:

Column	Description
Name	Name of the rule. Clicking the name opens the rule configuration form - for CEP rule or global correlation rule respectively.
Type	Type of the rule - *Complex event processing* or *Correlation*.
Conditions	Rule conditions are displayed.
Time window processing	Time window type is displayed.
Operations	Rule operations are displayed.
Stop after this rule	Stop execution after this rule - *Enabled* or *Disabled*.
Sort order	Sort order of the CEP rule.
Status	Rule status is displayed - *Enabled* or *Disabled*. By clicking the status you can change it.
Info	If everything is working correctly, no icon is displayed in this column. In case of errors, a square icon with the letter "i" is displayed. Hover over the icon to see a tooltip with the error description.

To configure a new CEP rule, click the *Create complex event processing* button in the upper-right corner.

To configure a new correlation rule, click the *Create event correlation* button in the upper-right corner.

Mass editing options

Buttons below the list offer some mass-editing options:

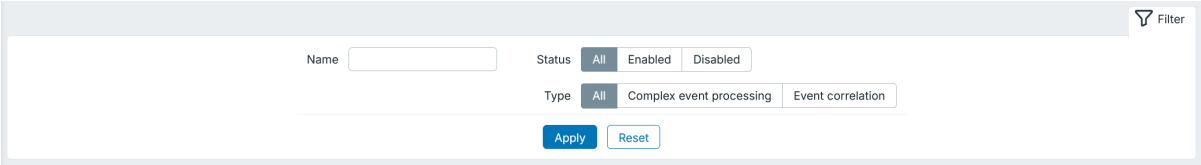
- *Enable* - change the rule status to *Enabled*
- *Disable* - change the rule status to *Disabled*
- *Delete* - delete the rules

To use these options, mark the checkboxes before the respective rules, then click the required button.

Using filter

You can use the filter to display only the rules you are interested in. For better search performance, data is searched with macros unresolved.

The *Filter* link is available above the list of CEP/correlation rules. If you click it, a filter becomes available where you can filter rules by name and status.

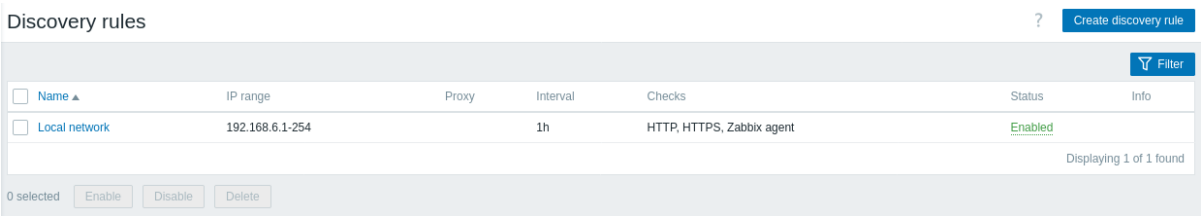


Descubrimiento

Descripción general

En la sección *Recopilación de datos* > *Descubrimiento*, los usuarios pueden configurar y mantener reglas de descubrimiento.

Se muestra una lista de las reglas de descubrimiento existentes con sus detalles.



Datos mostrados:

Columna	Descripción
Nombre	Nombre de la regla de descubrimiento. Al hacer clic en el nombre de la regla de descubrimiento, se abre el formulario de configuración de la regla de descubrimiento.
Rango de IP	Se muestra el rango de direcciones IP que se utilizará para el escaneo de red.
Proxy	Se muestra el nombre del proxy, si el descubrimiento lo realiza el proxy.
Intervalo	Se muestra la frecuencia con la que se realiza el descubrimiento.
Comprobaciones	Se muestran los tipos de comprobaciones utilizados para el descubrimiento.
Estado	Se muestra el estado de la regla de descubrimiento: *Habilitada* o *Deshabilitada*. Al hacer clic en el estado, puede cambiarlo.
Información	Si todo funciona correctamente, no se muestra nada en esta columna. En caso de errores, se muestra un icono de información rojo con la letra "i". Pase el cursor sobre el icono para ver una descripción del error en una información sobre herramientas.

Para configurar una nueva regla de descubrimiento, haga clic en el botón *Crear regla de descubrimiento* situado en la esquina superior derecha.

Opciones de edición masiva

Los botones debajo de la lista ofrecen algunas opciones de edición masiva:

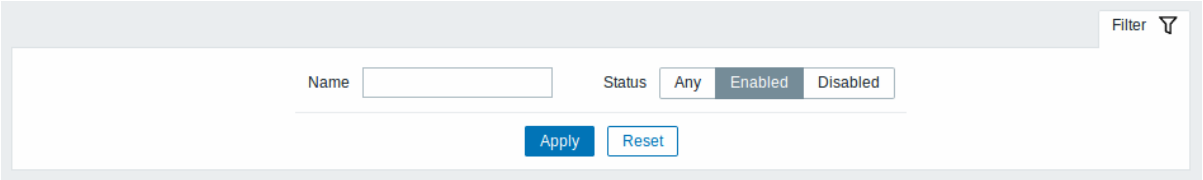
- *Enable* - cambia el estado de la regla de descubrimiento a *Enabled*
- *Disable* - cambia el estado de la regla de descubrimiento a *Disabled*
- *Delete* - elimina las reglas de descubrimiento

Para usar estas opciones, marque las casillas de verificación delante de las respectivas reglas de descubrimiento y luego haga clic en el botón requerido.

Uso del filtro

Puede usar el filtro para mostrar solo las reglas de descubrimiento que le interesan. Para un mejor rendimiento de búsqueda, los datos se buscan con las macros sin resolver.

El enlace *Filter* está disponible encima de la lista de reglas de descubrimiento. Si hace clic en él, se mostrará un filtro donde puede filtrar las reglas de descubrimiento por nombre y estado.



Alertas

Descripción general

Este menú contiene secciones relacionadas con la configuración de alertas en Zabbix.

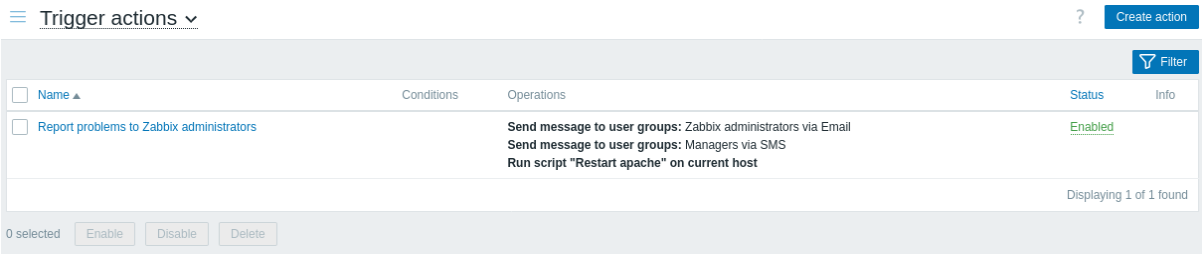
Acciones

Descripción general

En la sección *Alertas* > *Acciones*, los usuarios pueden configurar y mantener acciones.

Las acciones mostradas son las acciones asignadas a la fuente de eventos seleccionada (trigger, servicios, descubrimiento, registro automático, acciones internas).

Para cambiar a otra fuente de eventos, haga clic en *Acciones* en la sección del menú *Alertas*. También es posible cambiar entre fuentes mediante la lista desplegable del título, situada en la esquina superior izquierda.



Datos mostrados:

Columna	Descripción
Nombre	Nombre de la acción. Al hacer clic en el nombre de la acción se abre el formulario de configuración de la acción.
Condiciones	Se muestran las condiciones de la acción.
Operaciones	Se muestran las operaciones de la acción. La lista de operaciones también muestra el tipo de medio (correo electrónico, SMS o script) utilizado para la notificación, así como el nombre y los apellidos (entre paréntesis después del nombre de usuario) del destinatario de la notificación. La operación de acción puede ser una notificación o un comando remoto, según el tipo de operación seleccionado.
Estado	Se muestra el estado de la acción: *Habilitada* o *Deshabilitada*. Al hacer clic en el estado, puede cambiarlo. Consulte la sección Escalaciones para obtener más información sobre lo que ocurre si una acción se deshabilita durante una escalación en curso.
Información	Si todo funciona correctamente, no se muestra ningún icono en esta columna. En caso de errores, como operaciones o condiciones de acción faltantes después de una actualización o de un descubrimiento de bajo nivel, se muestra un icono de advertencia. Pase el cursor sobre el icono para ver una descripción del error en una información emergente.

Para configurar una acción nueva, haga clic en el botón *Crear acción* en la esquina superior derecha.

Para los usuarios sin derechos de Superadministrador, las acciones se muestran según la configuración de permisos. Esto significa que, en algunos casos, un usuario sin derechos de Superadministrador no puede ver la lista completa de acciones debido a ciertas restricciones de permisos. Una acción se muestra al usuario sin derechos de Superadministrador si se cumplen las siguientes condiciones:

- El usuario tiene acceso de lectura y escritura a los grupos de hosts, hosts, templates y triggers incluidos en las condiciones de la acción.
- El usuario tiene acceso de lectura y escritura a los grupos de hosts, hosts y templates incluidos en las operaciones de la acción, las operaciones de recuperación y las operaciones de actualización.
- El usuario tiene acceso de lectura a los grupos de usuarios y usuarios incluidos en las operaciones de la acción, las operaciones de recuperación y las operaciones de actualización.

Opciones de edición masiva

Los botones debajo de la lista ofrecen algunas opciones de edición masiva:

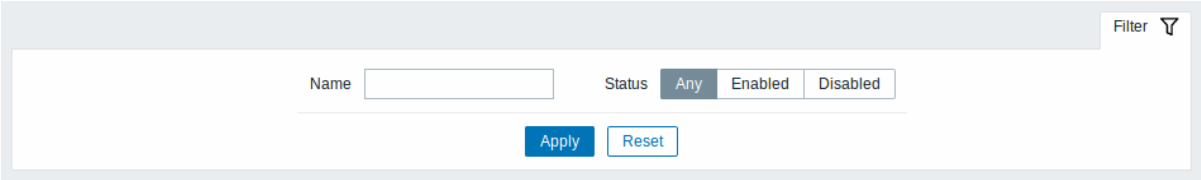
- *Enable* - cambia el estado de la acción a *Enabled*
- *Disable* - cambia el estado de la acción a *Disabled*
- *Delete* - elimina las acciones

Para usar estas opciones, marque las casillas de verificación delante de las acciones correspondientes, y luego haga clic en el botón requerido.

Usando el filtro

Puede usar el filtro para mostrar solo las acciones que le interesan. Para un mejor rendimiento de búsqueda, los datos se buscan con macros sin resolver.

El enlace *Filtro* está disponible sobre la lista de acciones. Si hace clic en él, se habilita un filtro donde puede filtrar acciones por nombre y estado.



Tipos de medios

Descripción general

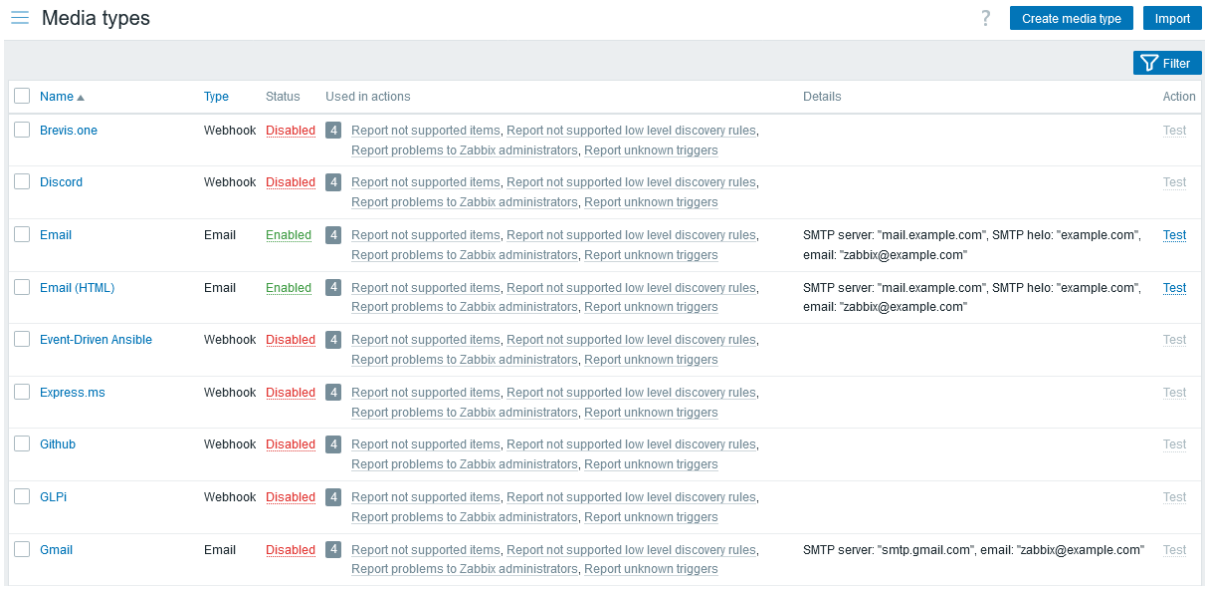
En la sección *Alertas* > *Tipos de medios*, los usuarios pueden configurar y mantener la información de los tipos de medios.

Este menú solo está disponible para los usuarios de tipo SuperAdmin.

La información de los tipos de medios contiene instrucciones generales para utilizar un medio como canal de entrega de notificaciones. Los detalles específicos, como las direcciones de correo electrónico individuales a las que se enviará una notificación, se mantienen con cada usuario.

Se muestra una lista de los tipos de medios existentes con sus detalles.

Note:
Si los métodos de entrega (*Correo electrónico*, *SMS*, *Webhook*, *Script* o *Push*) se especifican en el indicador de funcionalidad `$ZBX_FEATURE_FLAGS['media_type_denylist']` del archivo de configuración del frontend, los tipos de medios preconfigurados correspondientes se deshabilitan en esta sección.



Datos mostrados:

Columna	Descripción
Nombre	Nombre del tipo de medio. Al hacer clic en el nombre, se abre el formulario de configuración del tipo de medio.
Tipo	Se muestra el tipo de medio (correo electrónico, SMS, etc.).
Estado	Se muestra el estado del tipo de medio: *Habilitado* o *Deshabilitado*. Al hacer clic en el estado, puede cambiarlo.
Usado en acciones	Se muestran las acciones en las que se utiliza el tipo de medio, precedidas por el número total de estas acciones. Al hacer clic en el nombre de la acción, se abre el formulario de configuración de la acción. Si el usuario no tiene permisos para la acción, no se puede hacer clic en el nombre.
Detalles	Se muestra información detallada del tipo de medio.
Acciones	La siguiente acción está disponible: **Probar**: haga clic para abrir un formulario de prueba en el que puede introducir los parámetros del tipo de medio (por ejemplo, una dirección de destinatario con el asunto y el cuerpo de prueba) y enviar un mensaje de prueba para verificar que el tipo de medio configurado funciona. Consulte también: Prueba de tipos de medios.

Para configurar un nuevo tipo de medio, haga clic en el botón *Crear tipo de medio* en la esquina superior derecha.

Para importar un tipo de medio, haga clic en el botón *Importar* en la esquina superior derecha.

Opciones de edición masiva

Los botones debajo de la lista ofrecen algunas opciones de edición masiva:

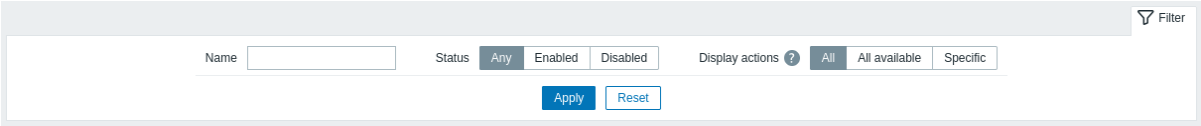
- *Enable* - cambia el estado del media type a *Enabled*
- *Disable* - cambia el estado del media type a *Disabled*
- *Export* - exporta los media types a un archivo YAML, XML o JSON
- *Delete* - elimina los media types

Para usar estas opciones, marque las casillas de verificación delante de los media types correspondientes y luego haga clic en el botón requerido.

Uso del filtro

Puede usar el filtro para mostrar solo los tipos de medio que le interesan. Para un mejor rendimiento de búsqueda, los datos se buscan con las macros sin resolver.

El enlace *Filter* está disponible encima de la lista de tipos de medio. Si hace clic en él, estará disponible un filtro con el que puede filtrar los tipos de medio por nombre y estado. Además, puede usar el filtro para mostrar acciones en la columna *Used in actions* en función del ámbito de uso de su tipo de medio (definido por el parámetro *Send to media type* en los detalles de la operación de la acción).



Scripts

Descripción general

En la sección *Alertas* > *Scripts* se pueden configurar y mantener scripts globales definidos por el usuario.

Este menú solo está disponible para usuarios del tipo SuperAdmin.

Los scripts globales, según el alcance configurado y los permisos del usuario, están disponibles para su ejecución:

- Desde el menú del host en varias ubicaciones del frontend (*Dashboard*, *Problems*, *Latest data*, *Maps*, etc.).
- Desde el menú de eventos.
- Se pueden ejecutar como una operación de una acción.

Los scripts se ejecutan mediante Zabbix agent, Zabbix server (proxy) o únicamente Zabbix server. Consulte también Ejecución de comandos.

Los comandos remotos están deshabilitados de forma predeterminada tanto en Zabbix agent como en Zabbix proxy. Se pueden habilitar de la siguiente manera:

- Para los comandos remotos ejecutados por Zabbix agent, añada un parámetro `AllowKey=system.run[<command>,*]` o `AllowKeyRegexp=^system\.run\[<command>,.*\]$` para cada comando permitido en la configuración del agent, donde `*` coincide con cualquiera de los modos de ejecución (`wait` o `nowait`).
- Para los comandos remotos ejecutados por Zabbix proxy, establezca el parámetro `EnableRemoteCommands` en 1 en la configuración del proxy. Tenga en cuenta que no es necesario habilitar los comandos remotos en Zabbix proxy cuando los ejecuta Zabbix agent en un host supervisado por el proxy.

Attention:
No utilice `AllowKey=system.run[*]` ni `AllowKeyRegexp=^system\.run\[.*\]$` a menos que sea necesario, ya que permite ejecutar cualquier comando en el host. Por motivos de seguridad, se recomienda permitir únicamente comandos exactos, mientras que `system.run[*]` está denegado de forma predeterminada (`DenyKey=system.run[*]`).

La ejecución de scripts globales en Zabbix server se puede deshabilitar estableciendo `EnableGlobalScripts=0` en la configuración del server. En las instalaciones nuevas a partir de Zabbix 7.0, la ejecución de scripts globales en Zabbix server está deshabilitada de forma predeterminada.

Se muestra una lista de los scripts existentes con sus detalles.



Datos mostrados:

Column	Description
Name	Nombre del script. Al hacer clic en el nombre del script, se abre el formulario de configuración del script.
Scope	Alcance del script: operación de acción, acción manual del host o acción manual del evento. Esta configuración determina dónde está disponible el script: en los comandos remotos de las operaciones de una acción, desde el menú del host o desde el menú de eventos, respectivamente.

Column	Description
Used in actions	Se muestran todas las acciones en las que se utiliza el script, precedidas por el número total de estas acciones. Al hacer clic en el nombre de la acción, se abre el formulario de configuración de la acción. Si el usuario no tiene permisos para la acción, el nombre no se puede seleccionar.
Type	Se muestra el tipo de script: comando *URL*, *Webhook*, *Script*, *SSH*, *Telnet* o *IPMI*.
Execute on	Se muestra si el script se ejecutará en Zabbix agent, Zabbix proxy o server, o únicamente en Zabbix server.
Commands	Se muestran todos los comandos que se ejecutarán dentro del script. Para los webhooks no se muestra nada aquí.
User group	Se muestra el grupo de usuarios para el que está disponible el script (o *All* para todos los grupos de usuarios).
Host group	Se muestra el grupo de hosts para el que está disponible el script (o *All* para todos los grupos de hosts).
Host access	Se muestra el nivel de permisos para el grupo de hosts: *Read* o *Write*. Solo los usuarios con el nivel de permisos requerido podrán ejecutar el script.

Para configurar un script nuevo, haga clic en el botón *Create script* situado en la esquina superior derecha.

Opciones de edición masiva

Un botón debajo de la lista ofrece una opción de edición masiva:

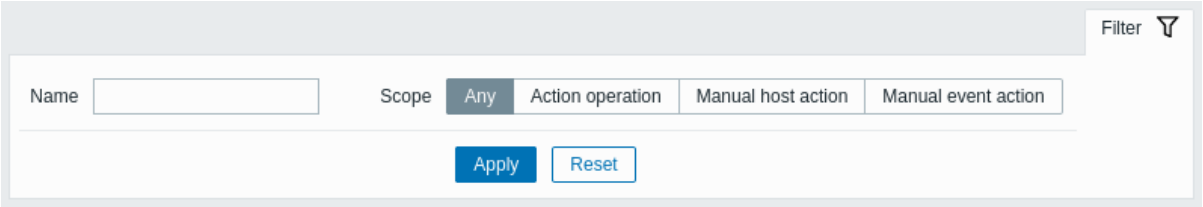
- *Eliminar* - elimina los scripts

Para usar esta opción, marque las casillas de verificación delante de los scripts correspondientes y haga clic en *Eliminar*.

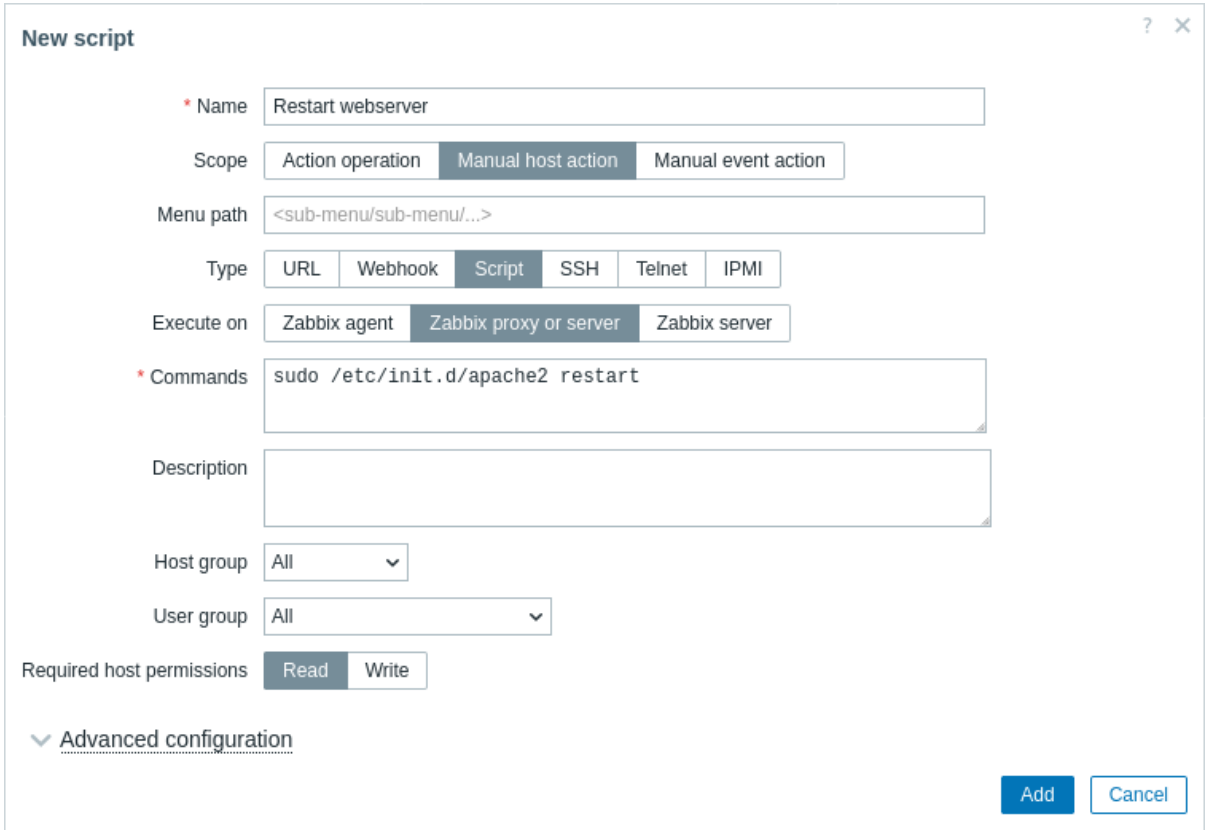
Uso del filtro

Puede usar el filtro para mostrar solo los scripts que le interesan. Para mejorar el rendimiento de la búsqueda, los datos se buscan con las macros sin resolver.

El enlace *Filter* está disponible encima de la lista de scripts. Si hace clic en él, se mostrará un filtro donde puede filtrar scripts por nombre y ámbito.



Configuración de un script global



Note:
Se recomienda utilizar funciones de macro en lugar de macros simples en los scripts globales para mejorar la seguridad, ya que las macros no se escapan automáticamente.

Atributos del script:

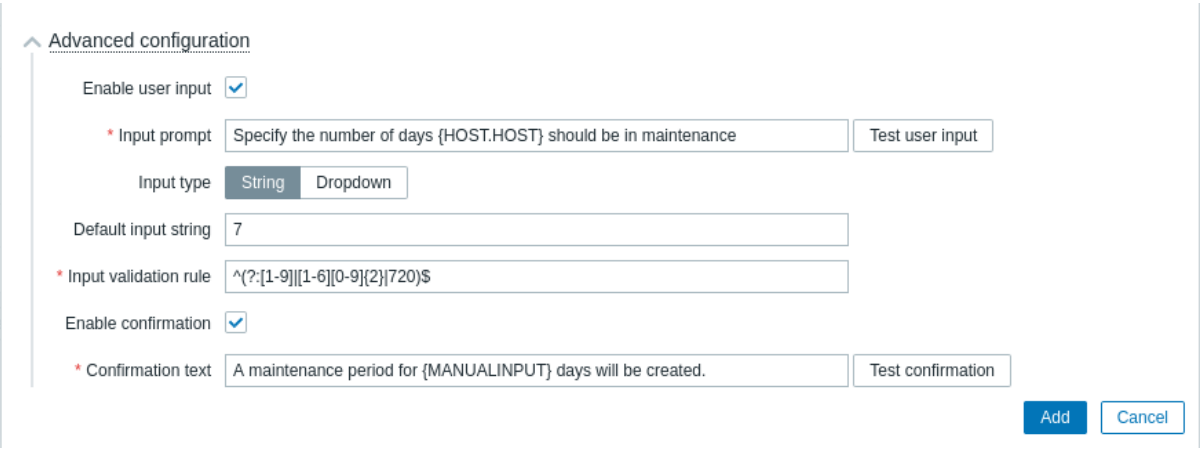
Parámetro	Descripción
Nombre	Nombre único del script. Por ejemplo, `Clear /tmp filesystem`
Ámbito	Ámbito del script: operación de acción, acción manual del host o acción manual del evento. Esta configuración determina dónde se puede utilizar el script: en los comandos remotos de las operaciones de una acción, desde el menú del host o desde el menú del evento, respectivamente. Establecer el ámbito en «Operación de acción» hace que el script esté disponible para todos los usuarios con acceso a *Alertas* > *Acciones*. Si un script se utiliza realmente en una acción, su ámbito no se puede cambiar de «operación de acción». **Compatibilidad con macros** El ámbito afecta al conjunto de macros disponibles. Por ejemplo, las macros relacionadas con el usuario (`{USER.*}`) son compatibles con los scripts para permitir pasar información sobre el usuario que inició el script. Sin embargo, no son compatibles si el ámbito del script es «operación de acción», ya que las operaciones de acción se ejecutan automáticamente. Una macro `{MANUALINPUT}` permite especificar una entrada manual en el momento de ejecutar el script. Es compatible con los scripts de acciones manuales del host y de acciones manuales del evento. Para saber qué otras macros son compatibles, busque «Notificaciones y comandos basados en trigger/Comandos basados en trigger», «Scripts de acciones manuales del host» y «Scripts de acciones manuales del evento» en la tabla de macros compatibles. Tenga en cuenta que, si una macro puede resolverse en un valor con espacios (por ejemplo, el nombre del host), debe incluir comillas cuando sea necesario.

Parámetro	Descripción
Ruta del menú	Ruta de menú deseada para el script. Por ejemplo, `Default` o `Default/` mostrarán el script en el directorio correspondiente. Los menús pueden estar anidados, por ejemplo, `Main menu/Sub menu1/Sub menu2`. Al acceder a los scripts mediante el menú del host/evento en las secciones de monitorización, se organizarán según los directorios indicados. Este campo solo se muestra si se selecciona «Acción manual del host» o «Acción manual del evento» como *Ámbito*.
Tipo	Haga clic en el botón correspondiente para seleccionar el tipo de script: comando **URL**, **Webhook**, **Script**, **SSH**, **Telnet** o **IPMI**. El tipo **URL** solo está disponible cuando se selecciona «Acción manual del host» o «Acción manual del evento» como *Ámbito*.
Tipo de script: **URL**	
URL	Especifique la URL para acceder rápidamente desde el menú del host o el menú del evento. Se admiten macros y macros de usuario personalizadas. La compatibilidad con macros depende del ámbito del script (consulte *Ámbito* más arriba). Utilice la macro {MANUALINPUT} en este campo para poder especificar una entrada manual en el momento de ejecutar el script, por ejemplo: `http://{MANUALINPUT}/zabbix/zabbix.php?action=dashboard.view` Los valores de las macros no deben codificarse como URL.
Abrir en una ventana nueva	Determina si la URL debe abrirse en una pestaña nueva o en la misma pestaña del navegador.
Tipo de script: **Webhook**	
Parámetros	Especifique las variables del webhook como pares atributo-valor. Consulte también: configuración del medio Webhook. Se admiten macros y macros de usuario personalizadas en los valores de los parámetros. La compatibilidad con macros depende del ámbito del script (consulte *Ámbito* más arriba).
Script	Introduzca el código JavaScript en el editor modal que se abre al hacer clic en el campo del parámetro o en el icono del lápiz situado junto a él. La compatibilidad con macros depende del ámbito del script (consulte *Ámbito* más arriba). Consulte también: configuración del medio Webhook, objetos JavaScript adicionales.
Tiempo de espera	Tiempo de espera de ejecución de JavaScript (1-60 s, valor predeterminado: 30 s). Se admiten sufijos de tiempo (por ejemplo, `30s`, `1m`).
Tipo de script: **Script**	
Ejecutar en	Haga clic en el botón correspondiente para ejecutar el script de shell en: **Zabbix agent**: el script será ejecutado por Zabbix agent (si el item system.run está permitido) en el host. **Zabbix proxy o server**: el script será ejecutado por Zabbix proxy o server, según si el host está monitorizado por un proxy o por un server. Se ejecutará en el proxy si está habilitado mediante EnableRemoteCommands. Se ejecutará en el server si los scripts globales están habilitados mediante el parámetro de server EnableGlobalScripts. **Zabbix server**: el script será ejecutado únicamente por Zabbix server. Esta opción no estará disponible si los scripts globales están deshabilitados mediante el parámetro de server EnableGlobalScripts.
Comandos	Introduzca la ruta completa de los comandos que se ejecutarán dentro del script. La compatibilidad con macros depende del ámbito del script (consulte *Ámbito* más arriba). Se admiten macros de usuario personalizadas.
Tipo de script: **SSH**	
Método de autenticación	Seleccione el método de autenticación: contraseña o clave pública.
Nombre de usuario	Introduzca el nombre de usuario.
Contraseña	Introduzca la contraseña. Este campo está disponible si se selecciona «Contraseña» como método de autenticación.
Archivo de clave pública	Introduzca la ruta al archivo de clave pública. Este campo está disponible si se selecciona «Clave pública» como método de autenticación.
Archivo de clave privada	Introduzca la ruta al archivo de clave privada. Este campo está disponible si se selecciona «Clave pública» como método de autenticación.
Frase de contraseña	Introduzca la frase de contraseña. Este campo está disponible si se selecciona «Clave pública» como método de autenticación.
Puerto	Introduzca el puerto del servicio SSH remoto en el host de destino al que se conectará Zabbix.

Parámetro	Descripción
Comandos	Introduzca los comandos. La compatibilidad con macros depende del ámbito del script (consulte *Ámbito* más arriba). Se admiten macros de usuario personalizadas.
Tipo de script: **Telnet**	
Nombre de usuario	Introduzca el nombre de usuario.
Contraseña	Introduzca la contraseña.
Puerto	Introduzca el puerto del servicio Telnet remoto en el host de destino al que se conectará Zabbix.
Comandos	Introduzca los comandos. La compatibilidad con macros depende del ámbito del script (consulte *Ámbito* más arriba). Se admiten macros de usuario personalizadas.
Tipo de script: **IPMI**	
Comando	Introduzca el comando IPMI. La compatibilidad con macros depende del ámbito del script (consulte *Ámbito* más arriba). Se admiten macros de usuario personalizadas.
Descripción	Introduzca una descripción para el script.
Grupo de hosts	Seleccione el grupo de hosts para el que estará disponible el script (o *Todos* para todos los grupos de hosts).
Grupo de usuarios	Seleccione el grupo de usuarios para el que estará disponible el script (o *Todos* para todos los grupos de usuarios). Este campo solo se muestra si se selecciona «Acción manual del host» o «Acción manual del evento» como *Ámbito*.
Permisos requeridos para el host	Seleccione el nivel de permisos para el grupo de hosts: *Lectura* o *Escritura*. Solo los usuarios con el nivel de permisos requerido podrán ejecutar el script. Este campo solo se muestra si se selecciona «Acción manual del host» o «Acción manual del evento» como *Ámbito*.
Configuración avanzada	Haga clic en el encabezado *Configuración avanzada* para mostrar las opciones de configuración avanzada. Este campo solo se muestra si se selecciona «Acción manual del host» o «Acción manual del evento» como *Ámbito*.

Configuración avanzada

Las opciones de configuración avanzada están disponibles en una sección contraíble de *Configuración avanzada*:



Parámetro	Descripción
Habilitar entrada del usuario	Marque la casilla para habilitar la entrada manual del usuario antes de ejecutar el script. La entrada manual del usuario reemplazará el valor de la macro {MANUALINPUT} en el script. Consulte también: Entrada manual del usuario.

Parámetro	Descripción
Mensaje de entrada	Introduzca el texto personalizado que solicitará la entrada personalizada del usuario. Este texto se mostrará encima del campo de entrada en la ventana emergente *Entrada manual*. Para ver una vista previa de la ventana emergente *Entrada manual*, haga clic en *Probar entrada del usuario*. La vista previa también permite comprobar si la cadena de entrada cumple la regla de validación de entrada (consulte los parámetros siguientes). La compatibilidad con macros y macros de usuario depende del ámbito del script (consulte *Ámbito* en los parámetros generales de configuración del script).
Tipo de entrada	Seleccione el tipo de entrada manual: **Cadena**: una sola cadena; **Lista desplegable**: el valor se selecciona entre varias opciones de la lista desplegable.
Opciones de la lista desplegable	Introduzca valores únicos para la lista desplegable de entrada del usuario en una lista delimitada por comas. Para incluir una opción vacía en la lista desplegable, añada una coma adicional al principio, en medio o al final de la lista. Este campo se muestra solo si se selecciona 'Lista desplegable' como *Tipo de entrada*.
Cadena de entrada predeterminada	Introduzca la cadena predeterminada para la entrada del usuario (o ninguna). Este campo se validará con la expresión regular proporcionada en el campo *Regla de validación de entrada*. El valor introducido aquí se mostrará de forma predeterminada en la ventana emergente *Entrada manual*. Este campo se muestra solo si se selecciona 'Cadena' como *Tipo de entrada*.
Regla de validación de entrada	Introduzca una expresión regular para validar la cadena de entrada del usuario. Se admiten expresiones regulares globales. Este campo se muestra solo si se selecciona 'Cadena' como *Tipo de entrada*.
Habilitar confirmación	Marque la casilla para mostrar un mensaje de confirmación antes de ejecutar el script. Esta función puede resultar especialmente útil para operaciones potencialmente peligrosas (como un script de reinicio) o para aquellas que puedan tardar mucho tiempo.
Texto de confirmación	Introduzca el texto de confirmación personalizado para la ventana emergente de confirmación habilitada con la casilla anterior (por ejemplo, *El sistema remoto se reiniciará. ¿Está seguro?*). Para ver cómo se mostrará el texto, haga clic en *Probar confirmación* junto al campo. Se admiten macros y macros de usuario personalizadas. *Nota:* las macros no se expandirán al probar el mensaje de confirmación.

Si se configuran tanto la entrada manual del usuario como un mensaje de confirmación, se mostrarán en ventanas emergentes consecutivas.

Entrada manual del usuario

La entrada manual del usuario permite proporcionar un parámetro personalizado en cada ejecución del script. Esto evita la necesidad de crear varios scripts de usuario similares que solo se diferencien en un parámetro.

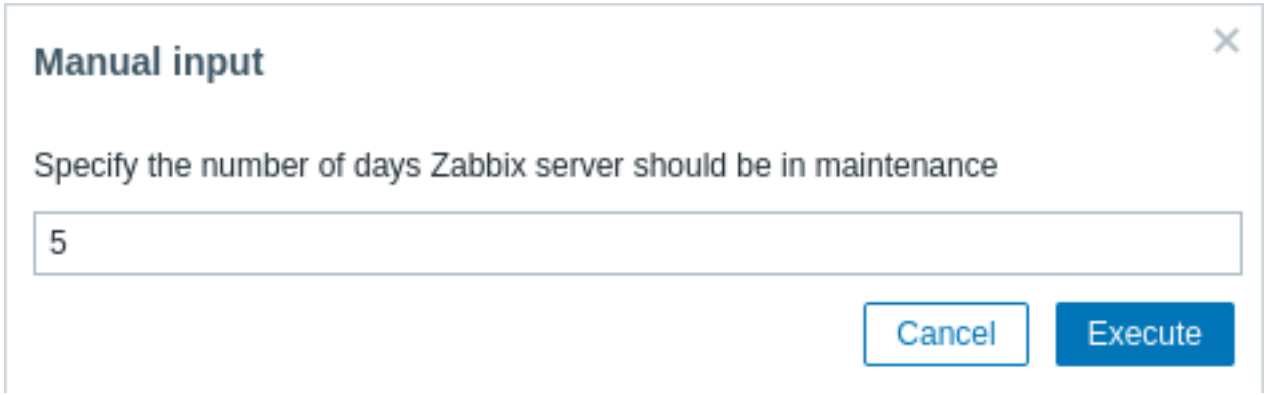
Por ejemplo, puede que desee proporcionar un número entero diferente o una dirección URL diferente al script durante la ejecución.

Para habilitar la entrada manual del usuario:

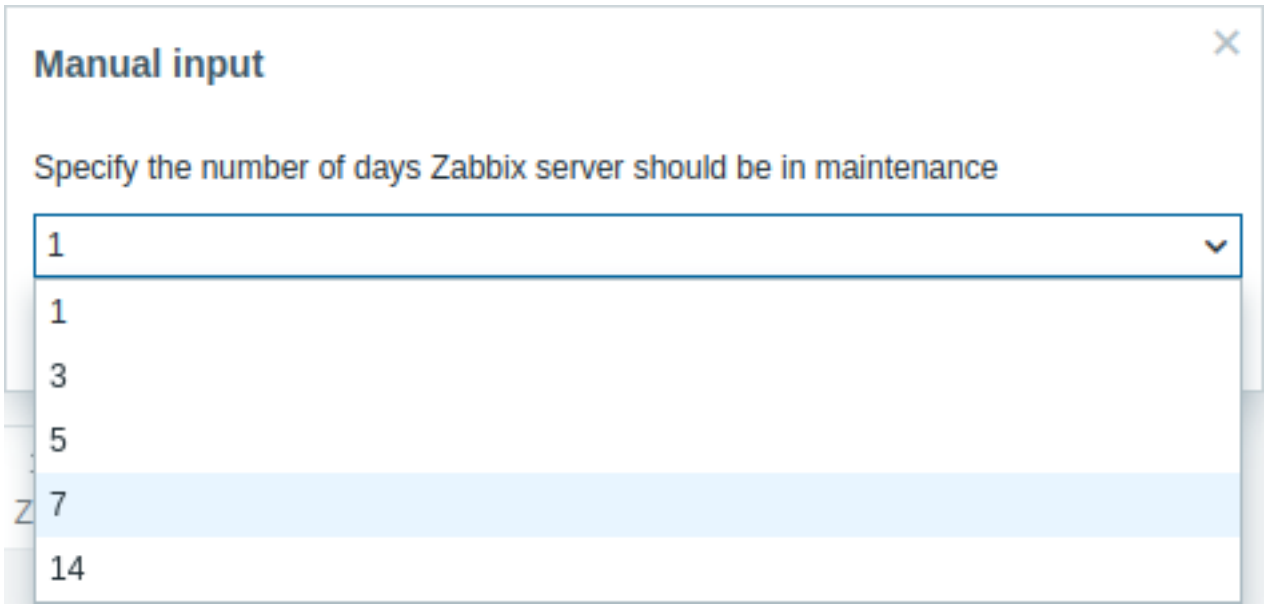
- use la macro {MANUALINPUT} en el script (comandos, script, parámetro del script) donde sea necesario; o en el campo URL de los scripts URL;
- en la configuración avanzada del script, habilite la entrada manual del usuario y configure las opciones de entrada.

Con la entrada del usuario habilitada, antes de ejecutar el script, aparecerá al usuario una ventana emergente de *Entrada manual* en la que se le solicitará proporcionar un valor personalizado. El valor proporcionado reemplazará a {MANUALINPUT} en el script.

Según la configuración, se solicitará al usuario que introduzca un valor de cadena:



O que seleccione el valor en una lista desplegable de opciones predeterminadas:



La entrada manual del usuario solo está disponible para scripts cuyo ámbito sea 'Acción manual del host' o 'Acción manual del evento'.

Ejecución del script y resultado

Los scripts ejecutados por el servidor Zabbix se ejecutan en el orden descrito en la página de ejecución de comandos.

El resultado del script se muestra en una ventana emergente que aparece después de ejecutar el script. El valor de retorno del script es una salida estándar:

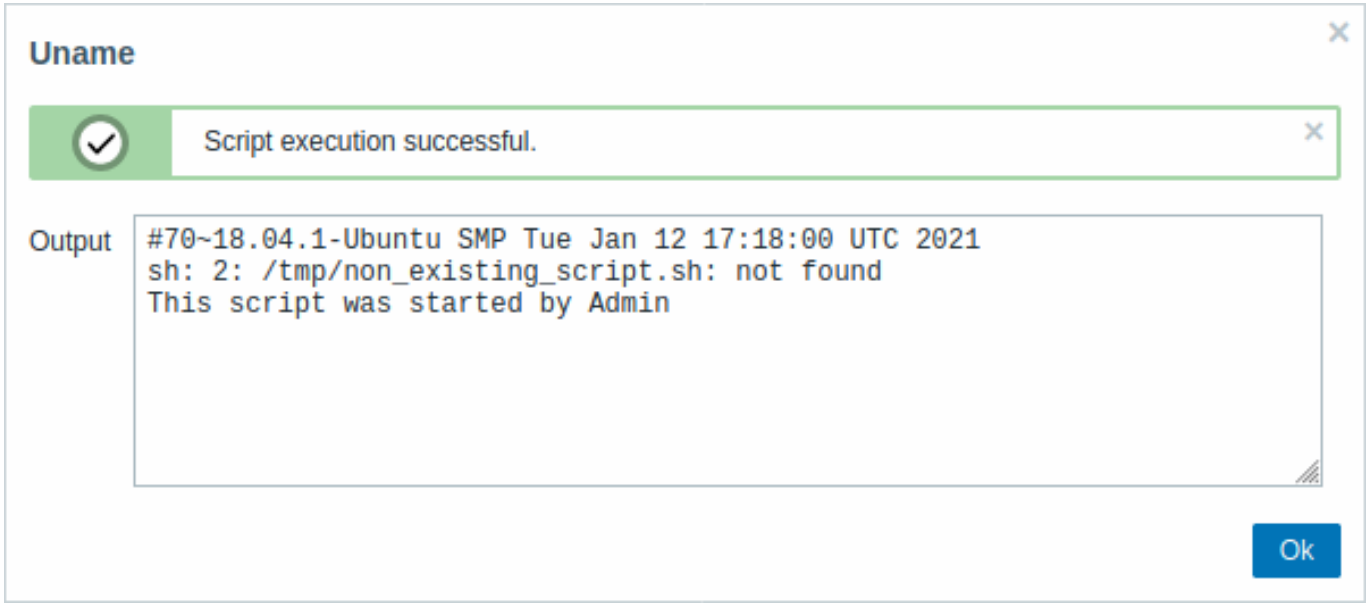
- Si el script finaliza correctamente (código de salida 0), el valor de retorno está limitado a 16MB (incluyendo los espacios en blanco finales que se truncan).
- Si el script finaliza con un error (código de salida distinto de cero), el valor de retorno es un error estándar limitado a 2KB.

Zabbix no almacena la salida extendida del script por defecto. Para conservar los detalles completos de la salida, puede implementar el registro dentro del propio script (por ejemplo, redirigir la salida a un archivo de registro local).

Tenga en cuenta que para los scripts ejecutados tanto en el servidor Zabbix como en el proxy Zabbix, también se aplican los límites de la base de datos.

A continuación se muestra un ejemplo de un script y la ventana de resultados:

```
uname -v
/tmp/non_existing_script.sh
echo "Este script fue iniciado por {USER.USERNAME}"
```



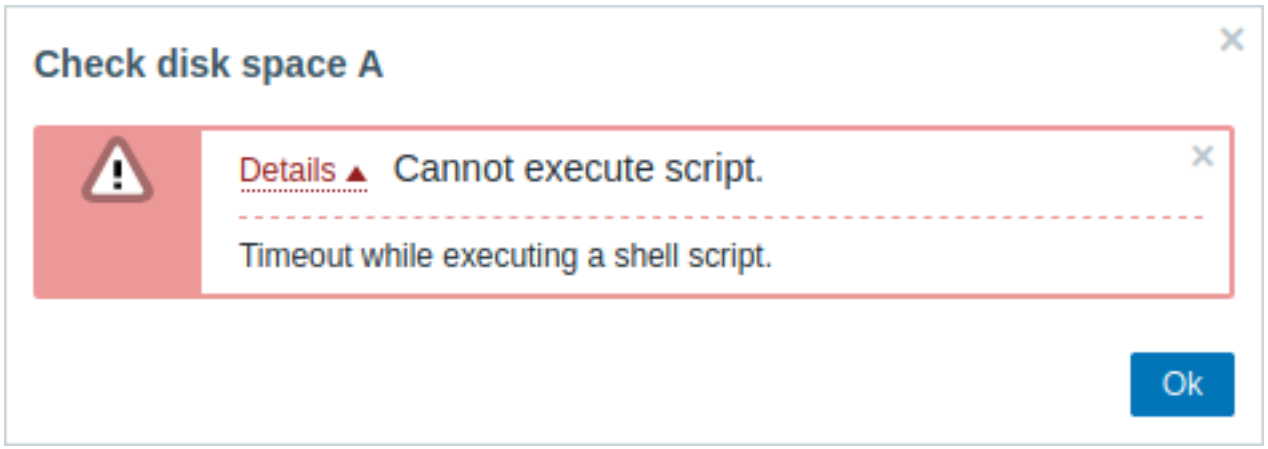
El resultado del script no muestra el propio script.

Tiempo de espera agotado del script en el Zabbix agent

Puede encontrarse con una situación en la que se agote el tiempo de espera mientras se ejecuta un script.

Vea a continuación un ejemplo de un script ejecutado en el Zabbix agent y la ventana de resultados:

```
sleep 5
df -h
```



En este caso, el mensaje de error es el siguiente:

```
Timeout while executing a shell script.
```

Para evitar estas situaciones, se recomienda optimizar el propio script (en el ejemplo anterior, ”5”) en lugar de ajustar el parámetro `Timeout` en la configuración del agent de Zabbix y en la configuración del server de Zabbix.

Si se ha cambiado el parámetro `Timeout` en la configuración del agent de Zabbix, aparecerá el siguiente mensaje de error:

```
Get value from agent failed: ZBX_TCP_READ() timed out.
```

Esto significa que la modificación se ha realizado en la configuración del agent de Zabbix, pero también es necesario modificar el parámetro `Timeout` en la configuración del server de Zabbix.

Tiempo de espera de script en agents activos

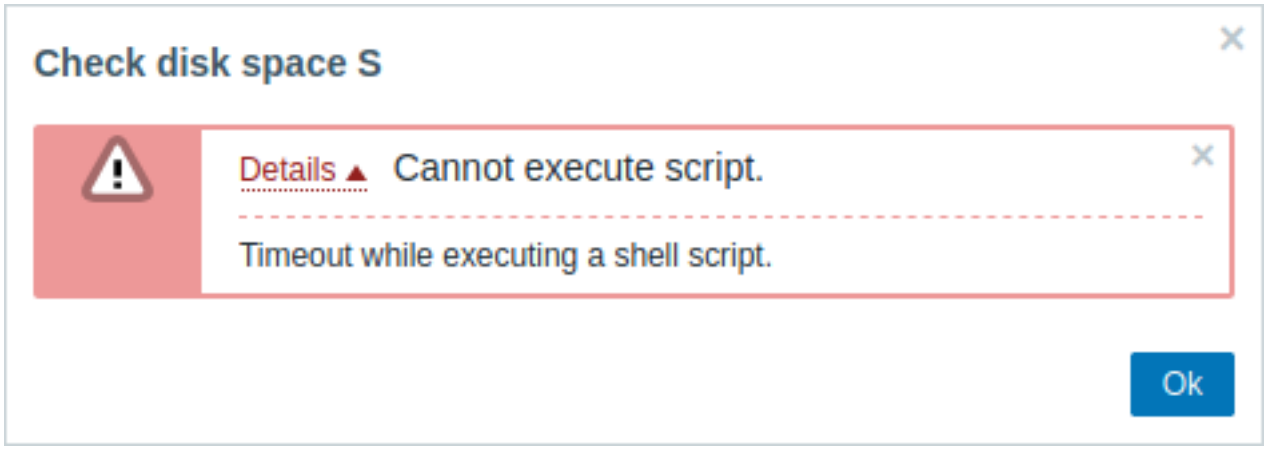
Es posible ejecutar scripts en agents que operan solo en modo activo. Una vez que la ejecución del script se activa mediante una operation de acción o la ejecución manual del script, el comando se incluye en la configuración de comprobación activa y se ejecuta cuando el agent activo lo recibe.

Los scripts manuales se envían al agent activo junto con el tiempo de espera de server/proxy para la ejecución del script. Aumente el tiempo de espera predeterminado de server/proxy para la ejecución del script. El tiempo de espera debe ser mayor que la frecuencia de actualización de la comprobación activa; de lo contrario, el tiempo de espera se excede antes de que el agent activo reciba el script y pueda devolver el resultado.

Tiempo de espera agotado del script en Zabbix server/proxy

A continuación se muestra un ejemplo de un script ejecutado en Zabbix server y la ventana de resultados:

```
sleep 11
df -h
```



También se recomienda optimizar el propio script (en lugar de ajustar el parámetro `TrapperTimeout` a un valor correspondiente (en nuestro caso, > 11) modificando la configuración del server de Zabbix).

Usuarios

Descripción general

Este menú contiene secciones relacionadas con la configuración de usuarios en Zabbix. Este menú está disponible solo para usuarios del tipo SuperAdmin.

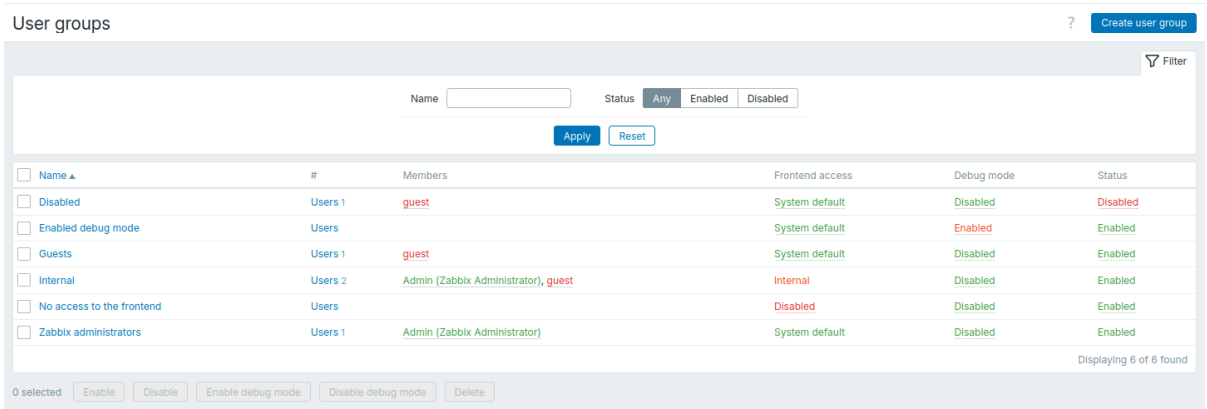
Grupos de usuarios

Resumen

En la sección *Users* > *User groups* se administran los grupos de usuarios del sistema.

Grupos de usuarios

Se muestra una lista de los grupos de usuarios existentes con sus detalles.



Datos mostrados:

Columna	Descripción
Nombre	Nombre del grupo de usuarios. Al hacer clic en el nombre del grupo de usuarios, se abre el formulario de configuración del grupo de usuarios.
#	Número de usuarios del grupo. Al hacer clic en *Usuarios*, se mostrarán los usuarios correspondientes filtrados en la lista de usuarios.
Miembros	Nombres de usuario de los usuarios individuales del grupo de usuarios (con el nombre y los apellidos entre paréntesis). Al hacer clic en el nombre de usuario, se abre el formulario de configuración del usuario. Los usuarios de grupos deshabilitados se muestran en rojo.

Columna	Descripción
Acceso al frontend	Se muestra el nivel de acceso al frontend: **Predeterminado del sistema**: los usuarios se autentican mediante Zabbix, LDAP o HTTP (según el método de autenticación establecido globalmente); **Interno**: los usuarios se autentican mediante Zabbix; se ignora si la autenticación HTTP es la predeterminada global; **LDAP**: los usuarios se autentican mediante LDAP; se ignora si la autenticación HTTP es la predeterminada global; **Deshabilitado**: se prohíbe el acceso al frontend de Zabbix para este grupo. Al hacer clic en el nivel actual, puede cambiarlo.
Modo de depuración	Se muestra el estado del modo de depuración: *Habilitado* o *Deshabilitado*. Al hacer clic en el estado, puede cambiarlo.
Estado	Se muestra el estado del grupo de usuarios: *Habilitado* o *Deshabilitado*. Al hacer clic en el estado, puede cambiarlo.

Para configurar un nuevo grupo de usuarios, haga clic en el botón *Crear grupo de usuarios* en la esquina superior derecha.

Opciones de edición masiva

Los botones debajo de la lista ofrecen algunas opciones de edición masiva:

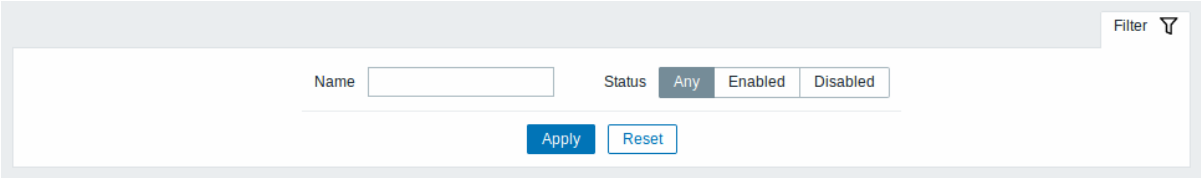
- *Enable* - cambia el estado del grupo de usuarios a *Enabled*
- *Disable* - cambia el estado del grupo de usuarios a *Disabled*
- *Enable debug mode* - habilita el modo de depuración para los grupos de usuarios
- *Disable debug mode* - deshabilita el modo de depuración para los grupos de usuarios
- *Delete* - elimina los grupos de usuarios

Para usar estas opciones, marque las casillas de verificación delante de los grupos de usuarios correspondientes y, a continuación, haga clic en el botón requerido.

Uso del filtro

Puede usar el filtro para mostrar solo los grupos de usuarios que le interesan. Para un mejor rendimiento de búsqueda, los datos se buscan con los macros sin resolver.

El enlace *Filter* está disponible encima de la lista de grupos de usuarios. Si hace clic en él, se mostrará un filtro que le permitirá filtrar los grupos de usuarios por nombre y estado.



Roles de usuario

Descripción general

En la sección *Usuarios* > *Roles de usuario* puede crear roles de usuario.

Los roles de usuario permiten crear permisos detallados basados en el tipo de usuario seleccionado inicialmente (*Usuario*, *Administrador*, *Superadministrador*).

Al seleccionar un tipo de usuario, se conceden todos los permisos disponibles para ese tipo de usuario (seleccionados de forma predeterminada).

Los permisos solo pueden revocarse del subconjunto disponible para el tipo de usuario; no pueden ampliarse más allá de lo disponible para dicho tipo de usuario.

Las casillas de verificación de los permisos no disponibles aparecen atenuadas; los usuarios no podrán acceder al elemento ni siquiera introduciendo directamente en el navegador la URL de dicho elemento.

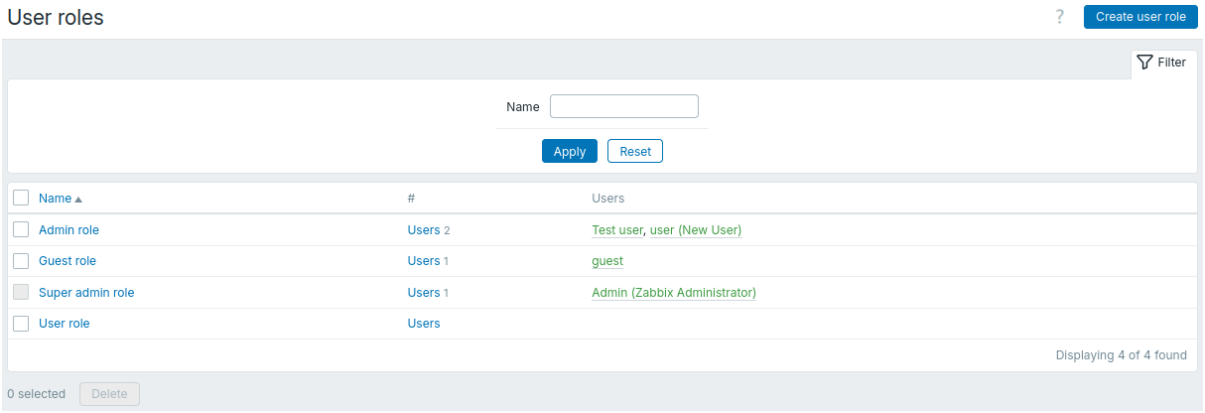
Note:
Restringir el acceso a algunos elementos de la interfaz de usuario solo impide abrir esa página; no elimina la posibilidad de acceder a los datos subyacentes desde otras partes de la interfaz.

Los roles de usuario pueden asignarse a los usuarios del sistema. Cada usuario solo puede tener un rol asignado.

Roles de usuario predeterminados

De forma predeterminada, Zabbix está configurado con cuatro roles de usuario, que tienen un conjunto predefinido de permisos:

- Rol Guest
- Rol User
- Rol Admin
- Rol Super admin



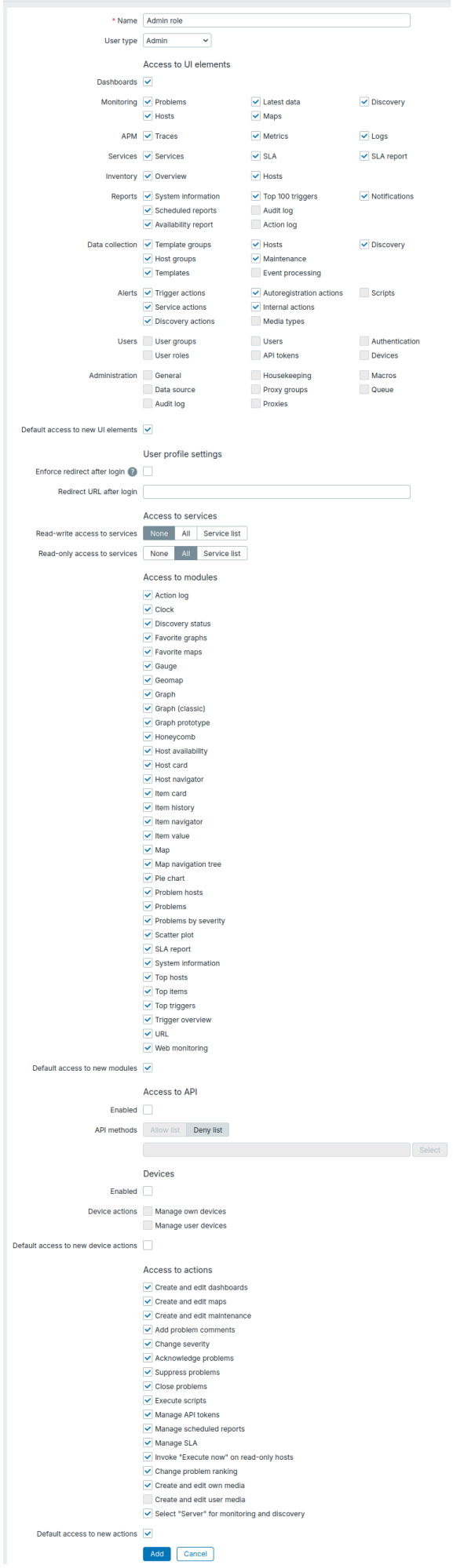
Estos se basan en los principales tipos de usuario de Zabbix. Se muestra la lista de todos los usuarios asignados al rol correspondiente. Los usuarios incluidos en grupos deshabilitados se muestran en rojo. El *rol Guest* es un rol de tipo de usuario que solo tiene permisos para ver algunas secciones del frontend.

Note:
El *rol Super admin* predeterminado no se puede modificar ni eliminar, porque en Zabbix debe existir al menos un usuario Super admin con privilegios ilimitados. Los usuarios de tipo *Super admin* pueden modificar la configuración de su propio rol, pero no el tipo de usuario.

Configuración

Para crear un nuevo rol, haga clic en el botón *Create user role* en la esquina superior derecha.
Para actualizar un rol existente, haga clic en el nombre del rol para abrir el formulario de configuración.

User roles


* Name
Admin role
User type
Admin
Access to UI elements
Dashboards
Monitoring
Problems
Latest data
Discovery
Hosts
Maps
APM
Traces
Metrics
Logs
Services
Services
SLA
SLA report
Inventory
Overview
Hosts
Reports
System information
Top 100 triggers
Notifications
Scheduled reports
Audit log
Availability report
Action log
Data collection
Template groups
Hosts
Discovery
Host groups
Maintenance
Templates
Event processing
Alerts
Trigger actions
Autoregistration actions
Scripts
Service actions
Internal actions
Discovery actions
Media types
Users
User groups
Users
Authentication
User roles
API tokens
Devices
Administration
General
Housekeeping
Macros
Data source
Proxy groups
Queue
Audit log
Proxies
Default access to new UI elements
User profile settings
Enforce redirect after login
Redirect URL after login
Access to services
Read-write access to services
None
All
Service list
Read-only access to services
None
All
Service list
Access to modules
Action log
Clock
Discovery status
Favorite graphs
Favorite maps
Gauge
Geomap
Graph
Graph (classic)
Graph prototype
Honeycomb
Host availability
Host card
Host navigator
Item card
Item history
Item navigator
Item value
Map
Map navigation tree
Pie chart
Problem hosts
Problems
Problems by severity
Scatter plot
SLA report
System information
Top hosts
Top items
Top triggers
Trigger overview
URL
Web monitoring
Default access to new modules
Access to API
Enabled
API methods
Allow list
Deny list
Select
Devices
Enabled
Device actions
Manage own devices
Manage user devices
Default access to new device actions
Access to actions
Create and edit dashboards
Create and edit maps
Create and edit maintenance
Add problem comments
Change severity
Acknowledge problems
Suppress problems
Close problems
Execute scripts
Manage API tokens
Manage scheduled reports
Manage SLA
Invoke "Execute now" on read-only hosts
Change problem ranking
Create and edit own media
Create and edit user media
Select "Server" for monitoring and discovery
Default access to new actions
Add
Cancel

Se muestran los permisos disponibles.
Para revocar un permiso determinado, desmarque su casilla.

A continuación se describen los permisos disponibles junto con los valores predeterminados para cada rol de usuario preconfigurado en Zabbix.

Permisos predeterminados

Acceso a elementos de la interfaz de usuario

El acceso predeterminado a las secciones del menú depende del tipo de usuario.
Consulte la página Permisos para obtener más información.

Acceso a otras opciones

Algunas de las secciones descritas en la tabla siguiente pueden estar ocultas:

- *Acceso a módulos* está oculto si `$ZBX_FEATURE_FLAGS['modules_config_enabled']` está establecido en `false` en el archivo de configuración del frontend.
- *Dispositivos* está oculto si `EnableMobileDevices` está deshabilitado.

Parámetro	Descripción	Roles de usuario predeterminados			
		Rol de superadministrador	*Rol de administrador*	*Rol de usuario*	*Rol de invitado*
Acceso predeterminado a nuevos elementos de la interfaz de usuario	Esta opción especifica cómo estarán disponibles las nuevas secciones del menú después de una actualización de Zabbix. Las secciones existentes del menú de los módulos no se ven afectadas.	Sí	Sí	Sí	Sí
Configuración del perfil de usuario					
Forzar redirección después del inicio de sesión	Marque la casilla para restringir a los usuarios a la URL de redirección después del inicio de sesión. Si no está marcada, los usuarios pueden establecer su URL personal después del inicio de sesión en su perfil; la URL de redirección después del inicio de sesión se utiliza únicamente como alternativa.	Sí	No	No	No
URL de redirección después del inicio de sesión	Especifique la URL de redirección después del inicio de sesión.				

Acceso a servicios					
Acceso de lectura y escritura a servicios	Seleccione el acceso de lectura y escritura a los servicios: **Ninguno**: ningún acceso **Todos**: acceso de lectura y escritura a todos los servicios **Lista de servicios**: seleccione los servicios para el acceso de lectura y escritura El acceso de lectura y escritura, si se concede, tiene prioridad sobre la configuración de acceso de solo lectura y se hereda dinámicamente en los servicios secundarios.	Todos	Todos	Ninguno	Ninguno
Acceso de lectura y escritura a servicios con etiqueta	Especifique el nombre de la etiqueta y, opcionalmente, su valor para conceder adicionalmente acceso de lectura y escritura a los servicios que coincidan con la etiqueta. Esta opción está disponible si se selecciona *Lista de servicios* en el parámetro *Acceso de lectura y escritura a servicios*. El acceso de lectura y escritura, si se concede, tiene prioridad sobre la configuración de acceso de solo lectura y se hereda dinámicamente en los servicios secundarios.				
Acceso de solo lectura a servicios	Seleccione el acceso de solo lectura a los servicios: **Ninguno**: ningún acceso **Todos**: acceso de solo lectura a todos los servicios **Lista de servicios**: seleccione los servicios para el acceso de solo lectura El acceso de solo lectura no tiene prioridad sobre el acceso de lectura y escritura y se hereda dinámicamente en los servicios secundarios.			Todos	Todos
Acceso de solo lectura a servicios con etiqueta	Especifique el nombre de la etiqueta y, opcionalmente, su valor para conceder adicionalmente acceso de solo lectura a los servicios que coincidan con la etiqueta. Esta opción está disponible si se selecciona *Lista de servicios* en el parámetro *Acceso de solo lectura a servicios*. El acceso de solo lectura no tiene prioridad sobre el acceso de lectura y escritura y se hereda dinámicamente en los servicios secundarios.				
Acceso a módulos					
<Nombre del módulo>	Permitir o denegar el acceso a un módulo específico. En esta sección solo se muestran los módulos habilitados. No es posible conceder ni restringir el acceso a un módulo que esté deshabilitado actualmente.	Sí	Sí	Sí	Sí
Acceso predeterminado a nuevos módulos	Esta opción especifica cómo estarán disponibles los nuevos módulos y widgets después de una actualización de Zabbix. También se aplica a los módulos y widgets añadidos en la sección *Administración > General > Módulos*.				

Acceso a la API					
Habilitado	Habilitar o deshabilitar el acceso a la API.	Sí	No	No	No
Métodos de la API	Seleccione *Lista de permitidos* para permitir, o *Lista de denegados* para denegar, los métodos de la API especificados en el campo de búsqueda. Tenga en cuenta que no es posible permitir algunos métodos de la API y denegar otros. En el campo de búsqueda, comience a escribir el nombre del método y, a continuación, seleccione el método en la lista de autocompletado. También puede hacer clic en el botón *Seleccionar* y seleccionar métodos de la lista completa disponible para este tipo de usuario. Tenga en cuenta que, si se desmarca alguna acción del bloque Acceso a acciones, los usuarios no podrán utilizar los métodos de la API relacionados con dicha acción. Se admiten comodines. Ejemplos: `dashboard.*` (todos los métodos del servicio de la API «dashboard.»), `*` (cualquier método), `*.export` (métodos con el nombre «.export» de todos los servicios de la API). Si no se especifica ningún método, *Lista de permitidos* deniega el acceso a todos los métodos de la API, mientras que *Lista de denegados* permite el acceso a todos los métodos de la API disponibles para el tipo de usuario. Para permitir todos los métodos con *Lista de permitidos*, especifique `*`. Para bloquear completamente el acceso a la API, desmarque la casilla *Habilitado* en lugar de dejar vacía la *Lista de permitidos*. La lista de métodos de la API no se aplica a los métodos que no requieren autenticación, como `apiinfo.version` y `user.login`.				
Acceso a acciones					
Crear y editar dashboards	Al desmarcar esta casilla también se revocarán los permisos para utilizar los métodos de la API `.create`, `.update` y `.delete` correspondientes a estos elementos.	Sí	Sí	Sí	No
Crear y editar mapas					
Crear y editar mantenimientos				No	
Añadir comentarios a problemas	Al desmarcar esta casilla también se revocarán los permisos para realizar la acción correspondiente mediante el método de la API `event.acknowledge`.			Sí	
Cambiar la gravedad					
Reconocer problemas					
Suprimir problemas					
Cerrar problemas					

Ejecutar scripts	Al desmarcar esta casilla también se revocarán los permisos para utilizar el método de la API `script.execute`.				
Administrar tokens de la API	Al desmarcar esta casilla también se revocarán los permisos para utilizar todos los métodos de la API `token.*`.				
Administrar informes programados	Al desmarcar esta casilla también se revocarán los permisos para utilizar todos los métodos de la API `report.*`.			No	
Administrar SLA	Habilitar o deshabilitar los permisos para administrar SLA.				
Invocar «Ejecutar ahora» en hosts de solo lectura	Permitir a los usuarios utilizar la opción «Ejecutar ahora» en Datos más recientes para los items de hosts de solo lectura.			Sí	
Cambiar la clasificación de los problemas	Permitir a los usuarios cambiar la clasificación del problema de causa a síntoma y viceversa.				
Crear y editar medios propios	Permitir a los usuarios crear y editar sus propios medios.				
Crear y editar medios de usuario	Permitir a los usuarios crear y editar medios para otros usuarios. Esta opción solo está disponible para usuarios superadministradores.		No	No	
Seleccionar «Server» para la monitorización y el descubrimiento	Al desmarcar esta casilla se restringe la creación de hosts monitorizados por el server y de reglas de descubrimiento de red ejecutadas por el server, así como las pruebas de items con el server. La opción *Server* estará deshabilitada en la configuración del host para el parámetro *Monitorizado por*, en la configuración de la regla de descubrimiento de red para el parámetro *Descubrimiento por* y en las pruebas de items para el parámetro *Probar con*.		Sí	No	
Acceso predeterminado a nuevas acciones	Esta opción especifica cómo estarán disponibles las nuevas acciones después de una actualización de Zabbix.		Sí	Sí	
Dispositivos					
Habilitado	Habilitar o deshabilitar el acceso a *Configuración del usuario* > *Dispositivos* (accesible para todos los usuarios) y *Usuarios* > *Dispositivos* (accesible únicamente para superadministradores).	Sí	No	No	No
Administrar dispositivos propios	Permitir a los usuarios administrar sus propios dispositivos en *Configuración del usuario* > *Dispositivos*.				
Administrar dispositivos de usuarios	Permitir a los usuarios administrar dispositivos pertenecientes a otros usuarios en *Usuarios* > *Dispositivos*.				
Acceso predeterminado a nuevas acciones	Esta opción especifica cómo estarán disponibles las nuevas acciones después de una actualización de Zabbix.				

Véase también:

- Configurar un usuario

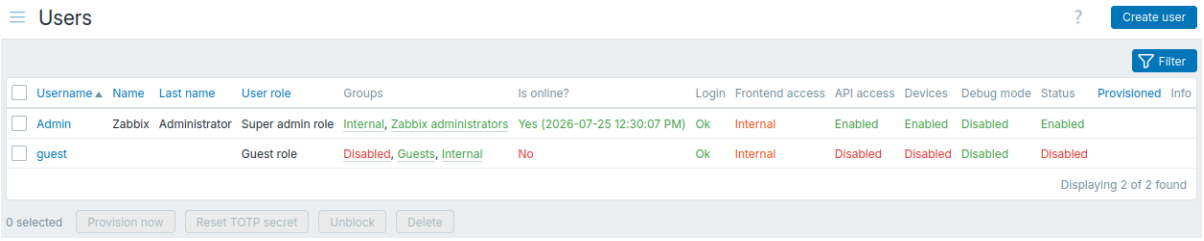
Usuarios

Descripción general

En la sección *Usuarios* > *Usuarios* se gestionan los usuarios del sistema.

Usuarios

Se muestra una lista de los usuarios existentes con sus detalles.



Datos mostrados:

Columna	Descripción
Nombre de usuario	Nombre de usuario para iniciar sesión en Zabbix. Al hacer clic en el nombre de usuario, se abre el formulario de configuración del usuario.
Nombre	Nombre del usuario.
Apellido	Segundo nombre del usuario.
Rol de usuario	Se muestra el rol de usuario.
Grupos	Se muestran los grupos de los que el usuario es miembro. Al hacer clic en el nombre del grupo de usuarios, se abre el formulario de configuración del grupo de usuarios. Los grupos deshabilitados se muestran en rojo.
¿Está en línea?	Se muestra el estado en línea del usuario: *Sí* o *No*. La hora de la última actividad del usuario se muestra entre paréntesis.
Inicio de sesión	Se muestra el estado de inicio de sesión del usuario: *Correcto* o *Bloqueado*. Un usuario puede bloquearse temporalmente al superar el número de intentos de inicio de sesión fallidos establecido en la sección *Administración > General > Otros* (cinco de forma predeterminada). Al hacer clic en *Bloqueado*, puede desbloquear el usuario.
Acceso al frontend	Se muestra el nivel de acceso al frontend: *Predeterminado del sistema*, *Interno*, *LDAP* o *Deshabilitado*, según el establecido para todo el grupo de usuarios.
Acceso a la API	Se muestra el estado de acceso a la API: *Habilitado* o *Deshabilitado*, según el establecido para el rol de usuario. Si el acceso a la API está habilitado, al pasar el cursor sobre *Habilitado* se muestran los métodos de API del rol de usuario en *Métodos permitidos* o *Métodos denegados*, según el modo de métodos de API. Se muestra *Ninguno* si no se especifica ningún método.
Dispositivos	Se muestra el estado de acceso a los dispositivos: *Habilitado* o *Deshabilitado*, según el establecido para el rol de usuario. Esta columna se muestra si `EnableMobileDevices` está habilitado.
Modo de depuración	Se muestra el estado del modo de depuración: *Habilitado* o *Deshabilitado*, según el establecido para todo el grupo de usuarios.
Estado	Se muestra el estado del usuario: *Habilitado* o *Deshabilitado*, según el establecido para todo el grupo de usuarios.
Provisionado	Se muestra la fecha en la que el usuario se provisionó por última vez. Se utiliza para los usuarios creados mediante el aprovisionamiento JIT desde LDAP/SAML.
Información	Se muestra información sobre los errores. Se muestra una advertencia amarilla para los usuarios sin grupos de usuarios. Se muestra una advertencia roja para los usuarios sin roles y para los usuarios sin roles ni grupos de usuarios.

Para configurar un usuario nuevo, haga clic en el botón *Crear usuario* en la esquina superior derecha.

Opciones de edición masiva

Los botones situados debajo de la lista ofrecen varias opciones de edición masiva:

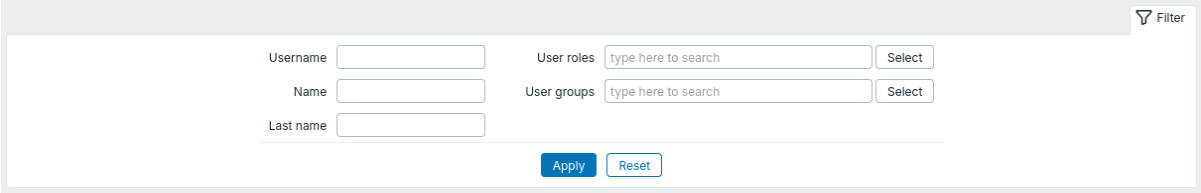
- *Aprovisionar ahora* - actualizar la información del usuario desde LDAP (esta opción solo está habilitada si se selecciona un usuario de LDAP)
- *Restablecer el secreto TOTP* - restablecer los secretos TOTP del usuario para todos los métodos TOTP y eliminar la sesión del usuario (esta opción solo está habilitada si MFA está habilitada; en el caso de los usuarios sin secretos TOTP, su sesión no se eliminará)
- *Desbloquear* - volver a habilitar el acceso al sistema para los usuarios bloqueados
- *Eliminar* - eliminar los usuarios

Para utilizar estas opciones, marque las casillas situadas delante de los usuarios correspondientes y, a continuación, haga clic en el botón requerido.

Uso del filtro

Puede utilizar el filtro para mostrar únicamente los usuarios que le interesan. Para mejorar el rendimiento de la búsqueda, los datos se buscan con las macros sin resolver.

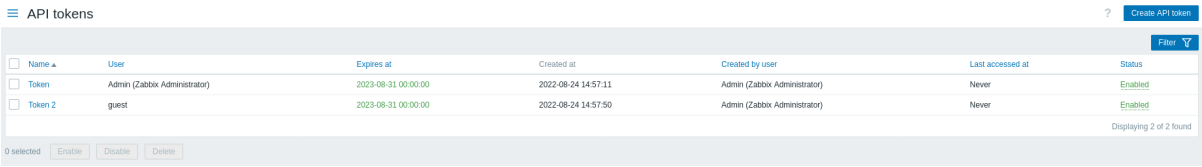
El enlace *Filtro* está disponible encima de la lista de usuarios. Al hacer clic en él, estará disponible un filtro que permite filtrar los usuarios por nombre de usuario, nombre, apellido, rol de usuario y grupo de usuarios.



Tokens de API

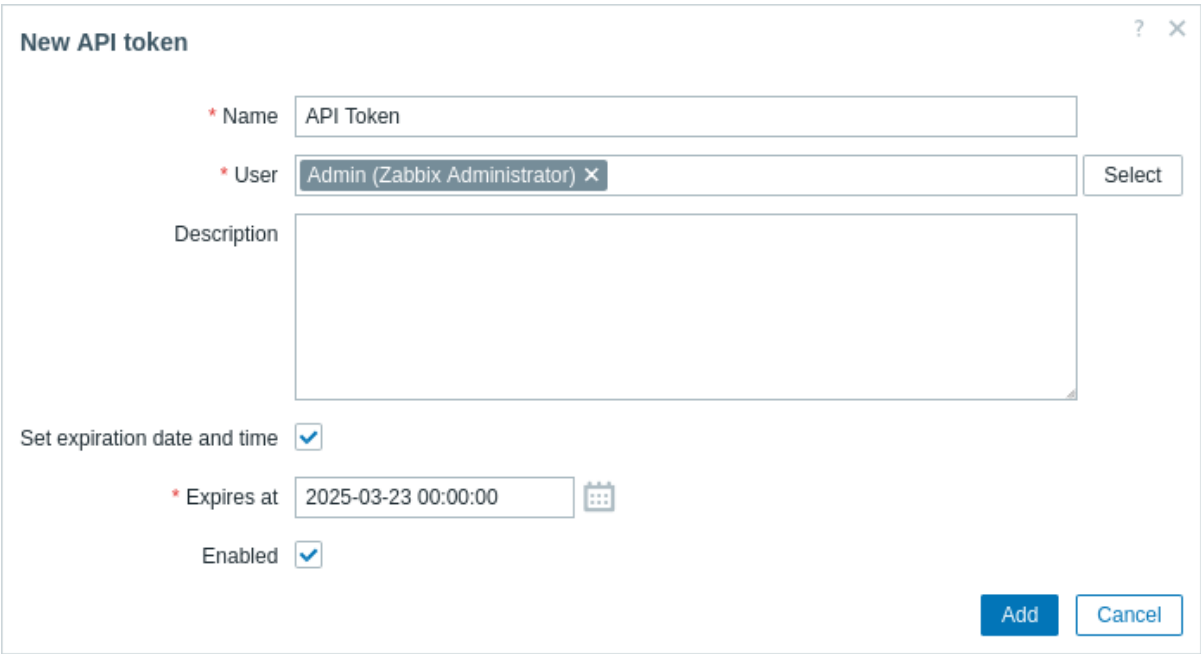
Descripción general

Esta sección permite crear y administrar tokens de API.
 Los tokens de API se utilizan para la autenticación en la API de Zabbix.
 Cada token de API se asigna a un único usuario y hereda los permisos de dicho usuario. Las solicitudes autenticadas con un token de API se autorizan según el rol y los permisos del usuario asociado.



Puede filtrar los tokens de API por nombre, usuarios a los que están asignados los tokens, fecha de caducidad, usuarios que crearon los tokens o estado (habilitado/deshabilitado). Haga clic en el estado del token en la lista para habilitarlo o deshabilitarlo rápidamente. También puede habilitar o deshabilitar varios tokens a la vez: selecciónelos en la lista y haga clic en los botones *Enable/Disable* situados debajo de la lista.

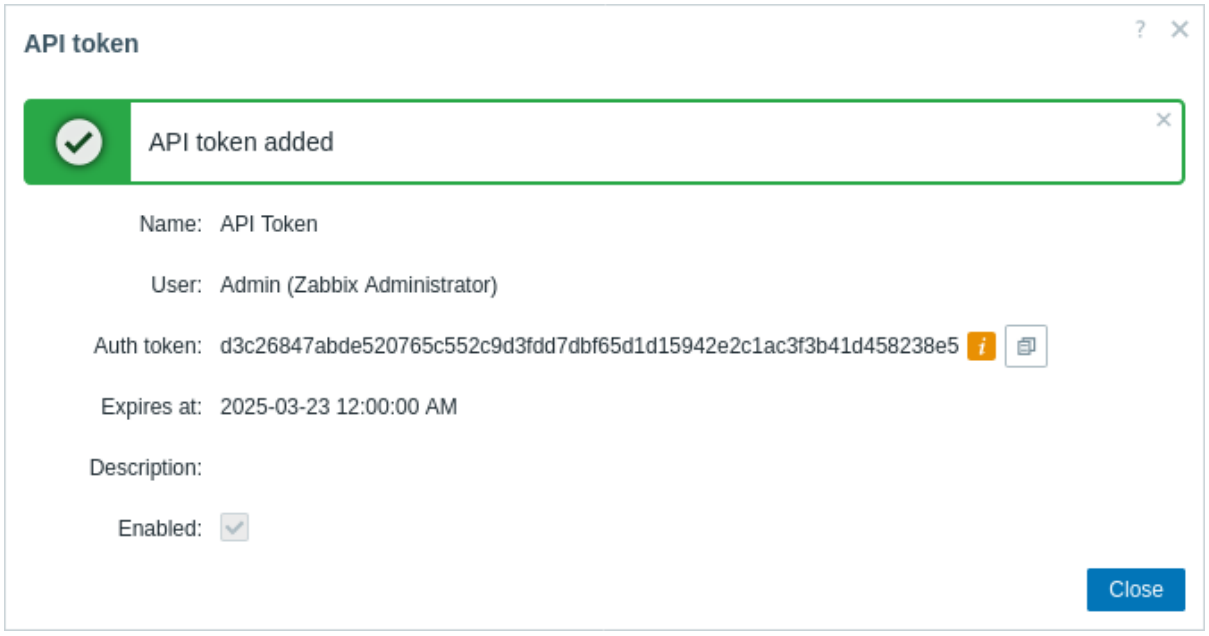
Para crear un token nuevo, haga clic en el botón *Create API token* situado en la esquina superior derecha y, a continuación, complete los campos obligatorios en la pantalla de configuración del token:



Parámetro	Descripción
Name	Nombre visible del token.
User	Usuario al que se debe asignar el token. Para seleccionar rápidamente un usuario, comience a escribir el nombre de usuario, el nombre o los apellidos y, a continuación, seleccione el usuario requerido en la lista de autocompletado. También puede hacer clic en el botón *Select* y seleccionar un usuario de la lista completa de usuarios. Un token solo puede asignarse a un usuario.
Description	Descripción opcional del token.

Parámetro	Descripción
Set expiration date and time	Desmarque esta casilla si el token no debe tener una fecha de caducidad.
Expiry date	Haga clic en el icono del calendario para seleccionar la fecha de caducidad del token o introduzca la fecha manualmente con el formato YYYY-MM-DD hh:mm:ss
Enabled	Desmarque esta casilla si necesita crear el token en estado deshabilitado.

Haga clic en *Add* para crear un token.



Copie el valor de *Auth token* y guárdelo en un lugar seguro **antes de cerrar la página**; después, haga clic en *Close*. El token aparecerá en la lista.

Warning:
El valor de *Auth token* no se puede volver a consultar posteriormente. Solo está disponible inmediatamente después de crear un token. Si pierde un token guardado, tendrá que regenerarlo, lo que creará una nueva cadena de autorización.

Haga clic en el nombre del token para editar el nombre, la descripción, la configuración de la fecha de caducidad o el estado del token. Tenga en cuenta que no es posible cambiar el usuario al que está asignado el token. Haga clic en el botón *Update* para guardar los cambios. Si un token se ha perdido o ha quedado expuesto, puede hacer clic en el botón *Regenerate* para generar un nuevo valor de token. Aparecerá un cuadro de diálogo de confirmación en el que se le pedirá confirmar esta operación, ya que, después de continuar, el token generado anteriormente dejará de ser válido.

Los usuarios sin acceso a la sección del menú *Administration* solo pueden ver y modificar los detalles de los tokens que tienen asignados en la sección *User profile* > *API tokens* si la opción *Manage API tokens* está permitida en los permisos de su rol de usuario.

Autenticación

Descripción general

La sección *Usuarios* > *Autenticación* permite especificar el método de autenticación de los usuarios para Zabbix y los requisitos de las contraseñas internas.

Los métodos de autenticación disponibles son la autenticación interna, HTTP, LDAP, SAML y MFA.

Autenticación predeterminada

De forma predeterminada, Zabbix utiliza la autenticación **interna** de Zabbix para todos los usuarios.

Es posible cambiar el método de autenticación predeterminado al sistema LDAP para todo el sistema. Para ello, vaya a la pestaña *LDAP* y configure los parámetros de LDAP; después, vuelva a la pestaña *Autenticación* y cambie el selector de *Autenticación predeterminada* a LDAP.

Tenga en cuenta que el método de autenticación se puede ajustar en el nivel de grupo de usuarios. Aunque la autenticación LDAP esté configurada globalmente, algunos grupos de usuarios todavía pueden autenticarse mediante Zabbix. Estos grupos deben tener el acceso al frontend establecido en Interno.

También es posible habilitar la autenticación LDAP solo para grupos de usuarios específicos si la autenticación interna se utiliza globalmente. En este caso, los detalles de la autenticación LDAP se pueden especificar y utilizar para grupos de usuarios específicos, cuyo acceso al frontend debe establecerse en LDAP. Si un usuario pertenece al menos a un grupo de usuarios con autenticación LDAP, no podrá utilizar el método de autenticación interna.

Los métodos de autenticación HTTP, SAML 2.0 y MFA se pueden utilizar además del método de autenticación predeterminado.

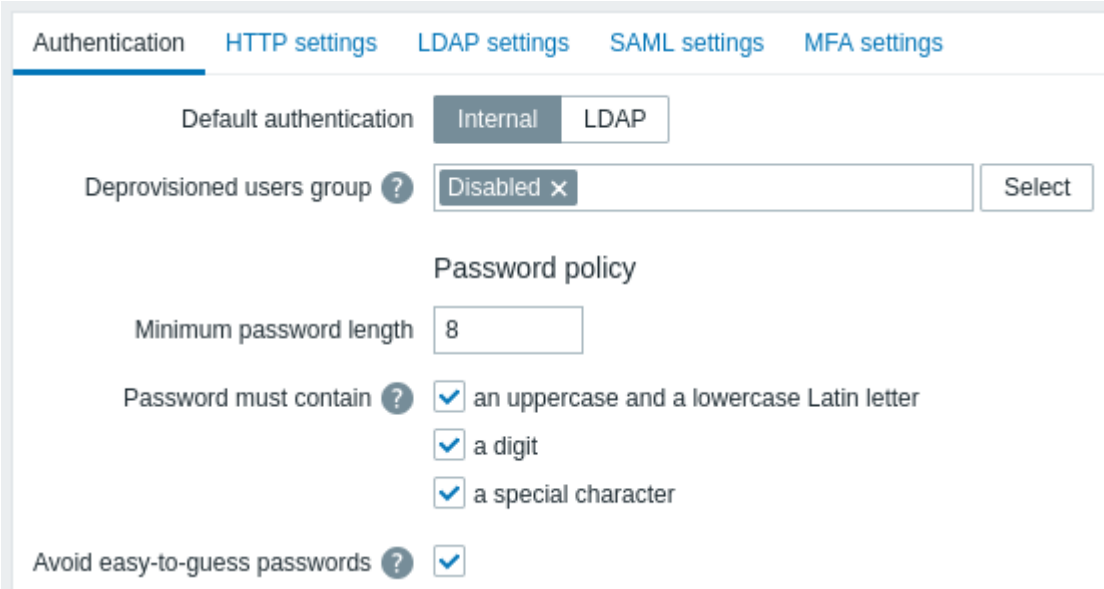
Zabbix admite el aprovisionamiento just-in-time (JIT), que permite crear cuentas de usuario en Zabbix la primera vez que un usuario externo se autentica y aprovisionar dichas cuentas de usuario. El aprovisionamiento JIT es compatible con LDAP y SAML.

Véase también:

- Autenticación HTTP
- Autenticación LDAP
- Autenticación SAML
- Autenticación MFA

Configuración

La pestaña *Autenticación* permite establecer el método de autenticación predeterminado, especificar un grupo para los usuarios desprovisionados y definir los requisitos de complejidad de las contraseñas de los usuarios de Zabbix.



Parámetros de configuración:

Parámetro	Descripción
Autenticación predeterminada	Seleccione el método de autenticación predeterminado para Zabbix: *Interna* o *LDAP*.
Grupo de usuarios desprovisionados	Especifique un grupo de usuarios para los usuarios desprovisionados. Esta configuración solo es necesaria para el aprovisionamiento JIT, en relación con los usuarios que se crearon en Zabbix desde sistemas LDAP o SAML, pero que ya no necesitan aprovisionarse. Debe especificarse un grupo de usuarios deshabilitado.
Longitud mínima de la contraseña	De forma predeterminada, la longitud mínima de la contraseña está establecida en 8. Rango admitido: 1-70. Tenga en cuenta que las contraseñas de más de 72 caracteres se truncarán.
La contraseña debe contener	Marque una o varias casillas para exigir el uso de los caracteres especificados en una contraseña: - una letra latina mayúscula y una minúscula - un dígito - un carácter especial Pase el cursor sobre el signo de interrogación para ver una sugerencia con la lista de caracteres de cada opción.

Parámetro	Descripción
Evitar contraseñas fáciles de adivinar	Si se marca, la contraseña se comprobará según los siguientes requisitos: - no debe contener el nombre, apellido ni nombre de usuario del usuario - no debe ser una de las contraseñas comunes o específicas del contexto. La lista de contraseñas comunes y específicas del contexto se genera automáticamente a partir de la lista "Top 100k passwords" de NCSC, la lista "Top 1M passwords" de SecLists y la lista de contraseñas específicas del contexto de Zabbix. No se permitirá a los usuarios internos establecer contraseñas incluidas en esta lista, ya que dichas contraseñas se consideran débiles debido a su uso común.

Los cambios en los requisitos de complejidad de las contraseñas no afectarán a las contraseñas existentes de los usuarios, pero si un usuario existente decide cambiar su contraseña, la nueva contraseña deberá cumplir los requisitos actuales. Junto al campo *Contraseña* se mostrará una sugerencia con la lista de requisitos en el perfil de usuario y en el formulario de configuración de usuarios, accesible desde el menú *Usuarios* > *Usuarios*.

HTTP

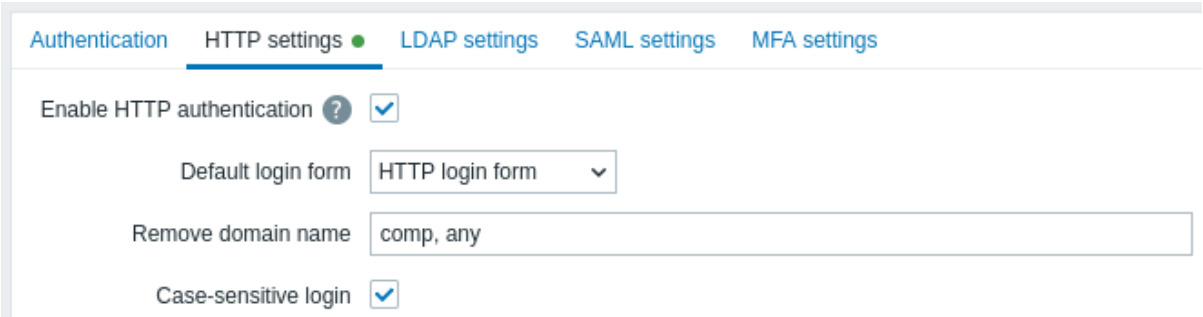
Descripción general

La autenticación basada en HTTP o en un servidor web (por ejemplo: BasicAuthentication, NTLM/Kerberos) se puede utilizar para comprobar los nombres de usuario y las contraseñas. Sin embargo, tenga en cuenta que el usuario también debe existir en Zabbix; no se utilizará su contraseña de Zabbix.

Attention:
¡Tenga cuidado! Asegúrese de que la autenticación del servidor web esté configurada y funcione correctamente antes de activarla.

Note:
La configuración de la autenticación HTTP se puede desactivar estableciendo `$ZBX_FEATURE_FLAGS['http_auth_enabled'] = false` en el archivo de configuración del frontend (`zabbix.conf.php`). En este caso, la pestaña *Configuración HTTP*, que contiene las opciones de autenticación HTTP, estará oculta.

Configuración



Parámetros de configuración:

Parámetro	Descripción
Habilitar la autenticación HTTP	Marque la casilla para habilitar la autenticación HTTP. Al pasar el ratón sobre (?), aparecerá un cuadro de información que advierte que, en el caso de la autenticación del servidor web, todos los usuarios (incluso aquellos cuyo acceso al frontend esté configurado como LDAP/Interno) serán autenticados por el servidor web y no por Zabbix.

Parámetro	Descripción
Formulario de inicio de sesión predeterminado	Especifique si se debe dirigir a los usuarios no autenticados a: **Formulario de inicio de sesión de Zabbix**: página de inicio de sesión estándar de Zabbix. **Formulario de inicio de sesión HTTP**: página de inicio de sesión HTTP. Se recomienda habilitar la autenticación basada en el servidor web únicamente para la página `index_http.php`. Si *Formulario de inicio de sesión predeterminado* está configurado como «Página de inicio de sesión HTTP», el usuario iniciará sesión automáticamente si el módulo de autenticación del servidor web establece un nombre de usuario válido en la variable `$_SERVER`. Las claves `$_SERVER` compatibles son `PHP_AUTH_USER`, `REMOTE_USER`, `AUTH_USER`.
Eliminar el nombre de dominio	Lista de nombres de dominio, separados por comas, que deben eliminarse del nombre de usuario. Por ejemplo, `comp,any`: si el nombre de usuario es «Admin@any» o «comp\Admin», el usuario iniciará sesión como «Admin»; si el nombre de usuario es «notacompany\Admin», se denegará el inicio de sesión.
Inicio de sesión sensible a mayúsculas y minúsculas	Desmarque la casilla para deshabilitar la distinción entre mayúsculas y minúsculas en los nombres de usuario (habilitada de forma predeterminada). Deshabilitar esta distinción permite, por ejemplo, iniciar sesión como «admin» aunque el usuario de Zabbix sea «Admin» o «ADMIN». Tenga en cuenta que, si la distinción entre mayúsculas y minúsculas está deshabilitada y existen varios usuarios de Zabbix con nombres de usuario similares (por ejemplo, Admin y admin), el inicio de sesión de esos usuarios siempre se denegará con el siguiente mensaje de error: «Authentication failed: supplied credentials are not unique.»

Note:
Para los usuarios internos que no puedan iniciar sesión mediante credenciales HTTP (con el formulario de inicio de sesión HTTP configurado como predeterminado) y reciban el error 401, puede añadir una línea `ErrorDocument 401 /index.php?form=default` a las directivas de autenticación básica. Esto redirigirá al formulario de inicio de sesión habitual de Zabbix.

LDAP

Descripción general

Se puede utilizar la autenticación externa LDAP para comprobar los nombres de usuario y las contraseñas.

La autenticación LDAP de Zabbix funciona al menos con Microsoft Active Directory y OpenLDAP.

Si sólo se configura el inicio de sesión LDAP, el usuario también debe existir en Zabbix, sin embargo, su contraseña de Zabbix no se utilizará. Si la autenticación es exitosa, Zabbix emparejará un nombre de usuario local con el atributo de nombre de usuario devuelto por LDAP.

Aprovisionamiento de usuarios

Es posible configurar el **aprovisionamiento de usuarios** JIT (just-in-time) para usuarios LDAP. En este caso, no es necesario que el usuario ya exista en Zabbix. La cuenta de usuario puede crearse cuando el usuario inicia sesión en Zabbix por primera vez.

Cuando un usuario LDAP introduce su nombre de usuario y contraseña de LDAP, Zabbix comprueba en el server LDAP *predeterminado* si existe este usuario. Si el usuario existe y todavía no tiene una cuenta en Zabbix, se crea un nuevo usuario en Zabbix y el usuario puede iniciar sesión.

Un usuario creado mediante el aprovisionamiento JIT se asocia con el server LDAP (directorio) establecido como predeterminado en el momento de la creación. Cambiar posteriormente el server LDAP predeterminado no cambia ni actualiza el server LDAP vinculado a los usuarios que ya se aprovisionaron.

Attention:
Si el aprovisionamiento JIT está habilitado, se debe especificar un grupo de usuarios para los usuarios desaprovisionados en la pestaña *Authentication*.

El aprovisionamiento JIT también permite actualizar las cuentas de usuario aprovisionadas según los cambios realizados en LDAP. Por ejemplo, si un usuario se mueve de un grupo LDAP a otro, también se moverá de un grupo a otro en Zabbix; si se elimina a un usuario de un grupo LDAP, también se eliminará del grupo en Zabbix y, si no pertenece a ningún otro grupo, se añadirá

al grupo de usuarios para usuarios desaprovisionados. Las cuentas de usuario aprovisionadas se actualizan según el período de aprovisionamiento configurado o cuando el usuario inicia sesión en Zabbix.

Tenga en cuenta que el aprovisionamiento en segundo plano lo realiza el frontend de Zabbix cuando el usuario interactúa con él o, al menos, tiene una página del frontend abierta en el navegador. No existen procesos en segundo plano dedicados al aprovisionamiento de usuarios.

LDAP admite tres formas de vincularse al directorio para la autenticación y las búsquedas:

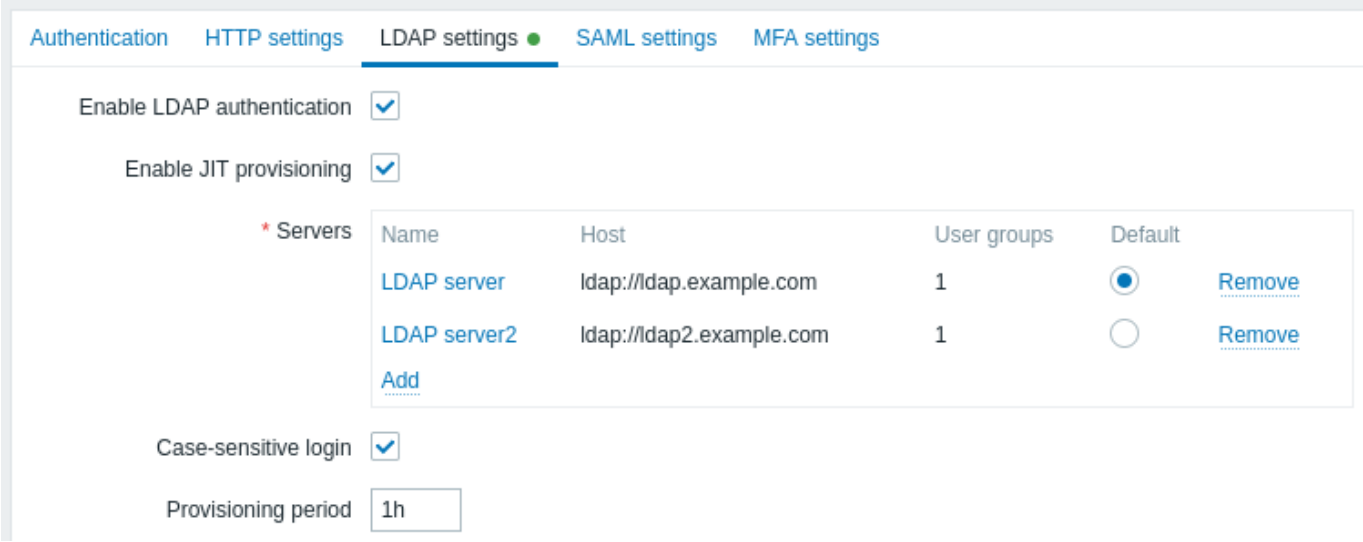
- Vinculación anónima: no se proporciona *Bind DN* / *Bind password* y el server LDAP permite consultas anónimas.
- Usuario de vinculación dedicado (cuenta de servicio): se configura una cuenta LDAP específica en *Bind DN* / *Bind password* y Zabbix la utiliza para realizar búsquedas y aprovisionamiento. Esta es la opción recomendada y más flexible, ya que Zabbix puede realizar búsquedas y aprovisionamiento en segundo plano sin las credenciales del usuario final.
- Vinculación directa del usuario: Zabbix realiza la vinculación utilizando las credenciales que el usuario introduce al iniciar sesión (no se configura *Bind DN* / *Bind password*); esto se configura incluyendo un marcador de posición como `uid=%{user}` en el Base DN. En este modo, Zabbix solo tiene acceso a la contraseña del usuario durante el inicio de sesión interactivo. Como resultado, las acciones de aprovisionamiento que requieren autenticarse en LDAP fuera de la sesión de inicio de sesión del usuario (por ejemplo, mediante el botón *Provision now* o al ejecutar el aprovisionamiento en segundo plano cuando el usuario no ha iniciado sesión activamente) no pueden autenticarse y, por lo tanto, no funcionarán. El aprovisionamiento y las actualizaciones para la vinculación directa del usuario solo se realizan cuando el usuario inicia sesión.

Varios servidores

Se pueden definir varios servidores LDAP, si es necesario. Por ejemplo, se puede utilizar un servidor diferente para autenticar a un grupo de usuarios diferente. Una vez configurados los servidores LDAP, en la configuración del grupo de usuarios es posible seleccionar el servidor LDAP requerido para el grupo de usuarios respectivo.

Si un usuario pertenece a varios grupos de usuarios y a varios servidores LDAP, se utilizará para la autenticación el primer servidor de la lista de servidores LDAP ordenados por nombre en orden ascendente.

Configuración

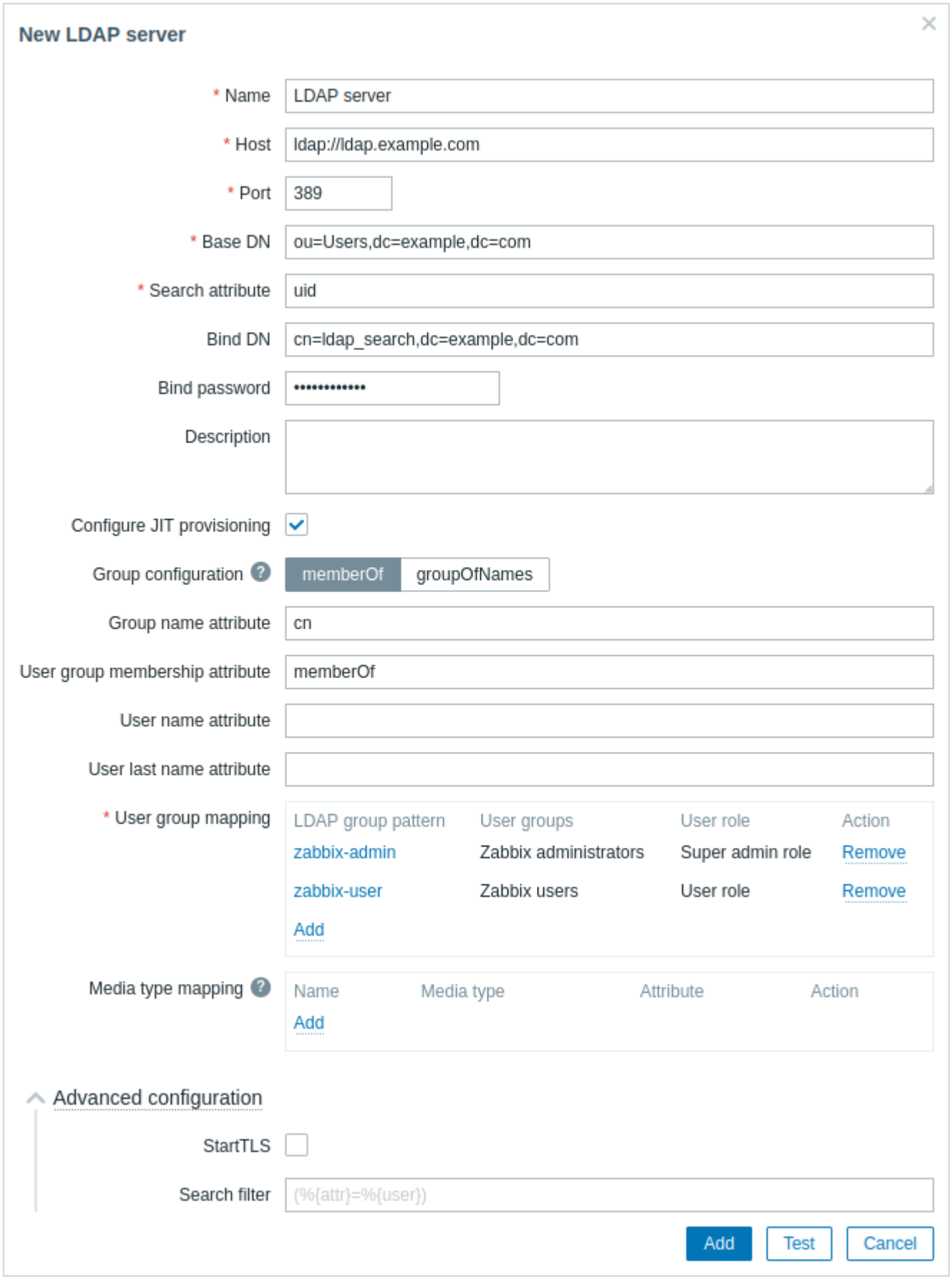


Parámetros de configuración:

Parámetro	Descripción
Habilitar la autenticación LDAP	Marque la casilla para habilitar la autenticación LDAP.
Habilitar el aprovisionamiento JIT	Marque la casilla para habilitar el aprovisionamiento JIT.
Servidores	Haga clic en *Añadir* para configurar un servidor LDAP (consulte Configuración del servidor LDAP a continuación).
Inicio de sesión sensible a mayúsculas y minúsculas	Desmarque la casilla para deshabilitar el inicio de sesión sensible a mayúsculas y minúsculas para los nombres de usuario (habilitado de forma predeterminada). Al deshabilitar el inicio de sesión sensible a mayúsculas y minúsculas, es posible, por ejemplo, iniciar sesión como ”admin” aunque el usuario de Zabbix sea ”Admin” o ”ADMIN”. Tenga en cuenta que, si el inicio de sesión sensible a mayúsculas y minúsculas está deshabilitado y existen varios usuarios de Zabbix con nombres de usuario similares (por ejemplo, Admin y admin), el inicio de sesión de esos usuarios siempre se rechazará con el siguiente mensaje de error: ”Authentication failed: supplied credentials are not unique.”

Parámetro	Descripción
Período de aprovisionamiento	Establezca el período de aprovisionamiento, es decir, con qué frecuencia se aprovisionará al usuario que ha iniciado sesión mientras trabaja con el frontend.

Configuración del servidor LDAP



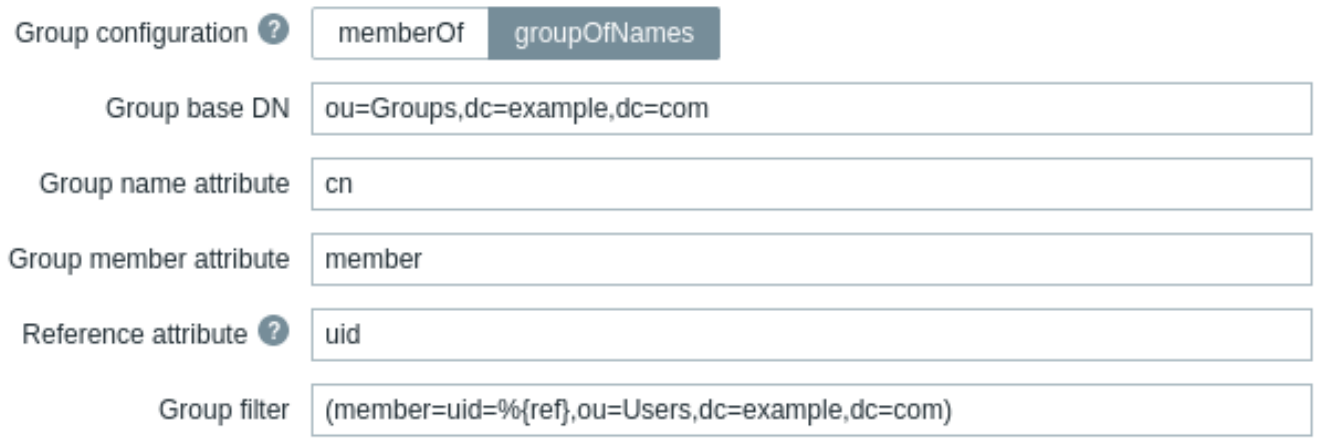
Parámetros de configuración del servidor LDAP:

Parameter	Description
Name	Nombre del servidor LDAP en la configuración de Zabbix.
Host	Nombre de host, IP o URI del servidor LDAP. Ejemplos: ldap.example.com, 127.0.0.1, ldap://ldap.example.com Para un servidor LDAP seguro, use el protocolo *ldaps* y el nombre de host. Ejemplo: ldaps://ldap.example.com Con OpenLDAP 2.x.x y versiones posteriores, se puede usar una URI LDAP completa del tipo ldap://hostname:port o ldaps://hostname:port.
Port	Puerto del servidor LDAP. El valor predeterminado es 389. Para una conexión LDAP segura, el número de puerto normalmente es 636. No se usa cuando se emplean URI LDAP completas.
Base DN	Ruta base de las cuentas de usuario en el servidor LDAP: ou=Users,ou=system (para OpenLDAP), DC=company,DC=com (para Microsoft Active Directory) uid=%{user},dc=example,dc=com (para enlace directo de usuario, consulte una nota más abajo)
Search attribute	Atributo de la cuenta LDAP usado para la búsqueda: uid (para OpenLDAP), sAMAccountName (para Microsoft Active Directory)
Bind DN	Cuenta LDAP para enlazar y realizar búsquedas en el servidor LDAP, ejemplos: uid=ldap_search,ou=system (para OpenLDAP), CN=ldap_search,OU=user_group,DC=company,DC=com (para Microsoft Active Directory) También se admite el enlace anónimo. Tenga en cuenta que el enlace anónimo puede exponer la configuración del dominio a usuarios no autorizados (información sobre usuarios, equipos, servidores, grupos, servicios, etc.). Por motivos de seguridad, deshabilite los enlaces anónimos en los hosts LDAP y use acceso autenticado en su lugar.
Bind password	Contraseña LDAP de la cuenta para enlazar y realizar búsquedas en el servidor LDAP.
Description	Descripción del servidor LDAP.
Configure JIT provisioning	Marque esta casilla para mostrar las opciones relacionadas con el aprovisionamiento JIT.
Group configuration	Seleccione el método de configuración de grupos: **memberOf** - mediante la búsqueda de usuarios y su atributo de pertenencia a grupos **groupOfNames** - mediante la búsqueda de grupos a través del atributo member Tenga en cuenta que memberOf es preferible porque es más rápido; use groupOfNames si su servidor LDAP no admite `memberOf` o si se requiere filtrado de grupos.
Group name attribute	Especifique el atributo del que obtener el nombre del grupo a partir de todos los objetos del atributo `memberOf` (consulte el campo *User group membership attribute*) El nombre del grupo es necesario para la asignación de grupos de usuarios.
User group membership attribute	Especifique el atributo que contiene información sobre los grupos a los que pertenece el usuario (por ejemplo, `memberOf`). Por ejemplo, el atributo memberOf puede contener información como esta: `memberOf=cn=zabbix-admin,ou=Groups,dc=example,dc=com` Este campo solo está disponible para el método memberOf.
User name attribute	Especifique el atributo que contiene el nombre del usuario.
User last name attribute	Especifique el atributo que contiene el apellido del usuario.
User group mapping	Asigne un patrón de grupo de usuarios LDAP a un grupo de usuarios y un rol de usuario de Zabbix. Esto es necesario para determinar qué grupo/rol de usuario obtendrá el usuario aprovisionado en Zabbix. Haga clic en *Add* para añadir una asignación. El campo *LDAP group pattern* admite comodines. El nombre del grupo debe coincidir con un grupo existente. Si un usuario LDAP coincide con varios grupos de usuarios de Zabbix, el usuario pasa a ser miembro de todos ellos. Si un usuario coincide con varios roles de usuario de Zabbix, obtendrá el que tenga el nivel de permisos más alto entre ellos.
Media type mapping	Asigne los atributos LDAP media del usuario (por ejemplo, correo electrónico) a los medios de usuario de Zabbix para enviar notificaciones (el valor del atributo se usa como campo *Send to* del medio).
Advanced configuration	Haga clic en el encabezado *Advanced configuration* para mostrar las opciones de configuración avanzada (consulte más abajo).

Parameter	Description
StartTLS	Marque la casilla para usar la operación StartTLS al conectarse al servidor LDAP. La conexión fallará si el servidor no admite StartTLS. StartTLS no se puede usar con servidores que utilizan el protocolo *ldaps*.
Search filter	Defina una cadena personalizada al autenticar un usuario en LDAP. Se admiten los siguientes marcadores de posición: `%{attr}` - nombre del atributo de búsqueda (uid, sAMAccountName) `%{user}` - valor del nombre de usuario a autenticar Por ejemplo, para realizar una búsqueda sensible a mayúsculas y minúsculas dentro de un entorno LDAP o Microsoft Active Directory que no distingue mayúsculas y minúsculas, la cadena se puede definir de la siguiente manera: `(%{attr}:caseExactMatch:=%{user})`. Si el filtro no se personaliza, LDAP usará el valor predeterminado: `(%{attr}=%{user})`.

Note:
Para configurar un servidor LDAP para **enlace directo de usuario**, añada un atributo uid=%{user} al parámetro *Base DN* (por ejemplo, *uid=%{user},dc=example,dc=com*) y deje vacíos los parámetros *BindDN* y *Bind password*. Al autenticar, el marcador de posición %{user} se reemplazará por el nombre de usuario introducido durante el inicio de sesión.
 Con el enlace directo de usuario, Zabbix solo tiene acceso a las credenciales del usuario durante el inicio de sesión interactivo. Por lo tanto, las tareas de aprovisionamiento que funcionan sin un inicio de sesión interactivo (por ejemplo, el botón *Provision now* o el método de API `user.provision`) ignorarán a los usuarios que se autentiquen mediante enlace directo de usuario porque Zabbix no puede enlazar en su nombre. Use enlace anónimo o un usuario de enlace dedicado (cuenta de servicio) si necesita que el aprovisionamiento y las búsquedas LDAP funcionen desde el frontend sin requerir que el usuario final haya iniciado sesión.

Los siguientes campos son específicos de "groupOfNames" como método de *Group configuration*:



Parameter	Description
Group base DN	Ruta base de los grupos en el servidor LDAP.
Group name attribute	Especifique el atributo para obtener el nombre del grupo en la ruta base de grupos especificada. El nombre del grupo es necesario para la asignación de grupos de usuarios.
Group member attribute	Especifique el atributo que contiene información sobre los miembros del grupo en LDAP (por ejemplo, `member`).
Reference attribute	Especifique el atributo de referencia para el filtro de grupos (consulte el campo *Group filter*). Luego use `%{ref}` en el filtro de grupos para obtener valores del atributo especificado aquí.
Group filter	Especifique el filtro para recuperar el grupo del que el usuario es miembro. Por ejemplo, `(member=uid=%{ref},ou=Users,dc=example,dc=com)` coincidirá con "User1" si el atributo member del grupo es `uid=User1,ou=Users,dc=example,dc=com` y devolverá el grupo del que "User1" es miembro.

Warning:
En caso de problemas con los certificados, para que funcione una conexión LDAP segura (ldaps) puede ser necesario añadir una línea `TLS_REQCERT allow` al archivo de configuración `/etc/openldap/ldap.conf`. Esto puede reducir la seguridad de la conexión al catálogo LDAP.

Note:
Se recomienda crear una cuenta LDAP separada (*Bind DN*) para realizar el enlace y las búsquedas en el servidor LDAP con privilegios mínimos en LDAP, en lugar de usar cuentas de usuario reales (utilizadas para iniciar sesión en el frontend de Zabbix).
 Este enfoque proporciona más seguridad y no requiere cambiar la *Bind password* cuando el usuario cambia su propia contraseña en el servidor LDAP.
 En la tabla anterior, es la cuenta *ldap_search*.

Prueba de acceso

El botón *Probar* permite probar el acceso del usuario:

Parámetro	Descripción
Inicio de sesión	Nombre de usuario LDAP que se probará (precompletado con el nombre del usuario actual del frontend de Zabbix). Este nombre de usuario debe existir en el server LDAP. Zabbix no activará la autenticación LDAP si no puede autenticar al usuario de prueba.
Contraseña del usuario	Contraseña del usuario LDAP que se probará.

SAML

Resumen

SAML 2.0 authentication se puede usar para iniciar sesión en Zabbix.

Si el inicio de sesión SAML se configura sin aprovisionamiento de usuarios JIT, el usuario debe existir previamente en Zabbix; la contraseña de Zabbix del usuario no se usa para la autenticación. Después de una autenticación SAML correcta, Zabbix compara el atributo de nombre de usuario recibido desde SAML con un nombre de usuario local existente en Zabbix.

Aprovisionamiento de usuarios

Es posible configurar el **aprovisionamiento de usuarios** JIT (just-in-time) para usuarios SAML. En este caso, no se requiere que un usuario ya exista en Zabbix. La cuenta de usuario puede crearse cuando el usuario inicia sesión en Zabbix por primera vez.

Attention:
Si el aprovisionamiento JIT está habilitado, se debe especificar un grupo de usuarios para los usuarios desaprovisionados en la pestaña *Autenticación*.

Además del aprovisionamiento JIT, también es posible habilitar y configurar el aprovisionamiento SCIM (System for Cross-domain Identity Management): la gestión *continua* de cuentas de usuario para aquellos usuarios que han sido creados mediante aprovisionamiento de usuarios. El aprovisionamiento SCIM requiere un token de API de Zabbix (con permisos de Superadministrador) para la autenticación en Zabbix.

Por ejemplo, si un usuario se mueve de un grupo SAML a otro, el usuario también se moverá de un grupo a otro en Zabbix; si un usuario se elimina de un grupo SAML, el usuario también se eliminará del grupo en Zabbix y, si no pertenece a ningún otro grupo, se añadirá al grupo de usuarios para usuarios desaprovisionados.

Si SCIM está habilitado y configurado, un usuario SAML será aprovisionado en el momento en que inicie sesión en Zabbix y se actualizará continuamente en función de los cambios en SAML. Los usuarios SAML ya existentes no serán aprovisionados, y solo los usuarios aprovisionados serán actualizados. Tenga en cuenta que solo los medios válidos se añadirán a un usuario cuando este sea aprovisionado o actualizado.

Si SCIM no está habilitado, un usuario SAML será aprovisionado (y posteriormente actualizado) en el momento en que inicie sesión en Zabbix.

Note:
Si la autenticación SAML está habilitada, los usuarios podrán elegir entre iniciar sesión localmente o mediante inicio de sesión único SAML. Si se utiliza el aprovisionamiento JIT, entonces solo es posible el inicio de sesión único.

Configuración del proveedor de identidad

Para trabajar con Zabbix, un proveedor de identidad SAML (onelogin.com, auth0.com, okta.com, etc.) debe configurarse de la siguiente manera:

- La *URL de consumidor de aserciones* debe establecerse en `<ruta_a_la_ui_de_zabbix>/index_sso.php?acs`
- La *URL de cierre de sesión único* debe establecerse en `<ruta_a_la_ui_de_zabbix>/index_sso.php?sls`

Ejemplos de <ruta_a_la_ui_de_zabbix>: https://example.com/zabbix/ui, http://another.example.com/zabbix, http://<cualquier_dirección_ip_pública>/zabbix

Configuración de Zabbix

> **Attention:**
> Es necesario instalar php-openssl si desea utilizar la autenticación SAML en el frontend.

Para utilizar la autenticación SAML, Zabbix debe configurarse de la siguiente manera:

1. Proporcione la clave privada y los certificados del SP. Debe proporcionarse un certificado del IdP (pegando el texto o seleccionando un archivo) para activar SAML mediante el frontend. Deben proporcionarse un certificado del SP y una clave privada del SP si se seleccionan las opciones *Firmar* o *Cifrar*.

Si se establece `$SSO['CERT_STORAGE'] = 'database'` en el archivo de configuración del frontend (`zabbix.conf.php`), puede pegar el texto del certificado o cargar el archivo del certificado en el frontend durante la configuración de SAML; no se requieren archivos en el sistema de archivos.

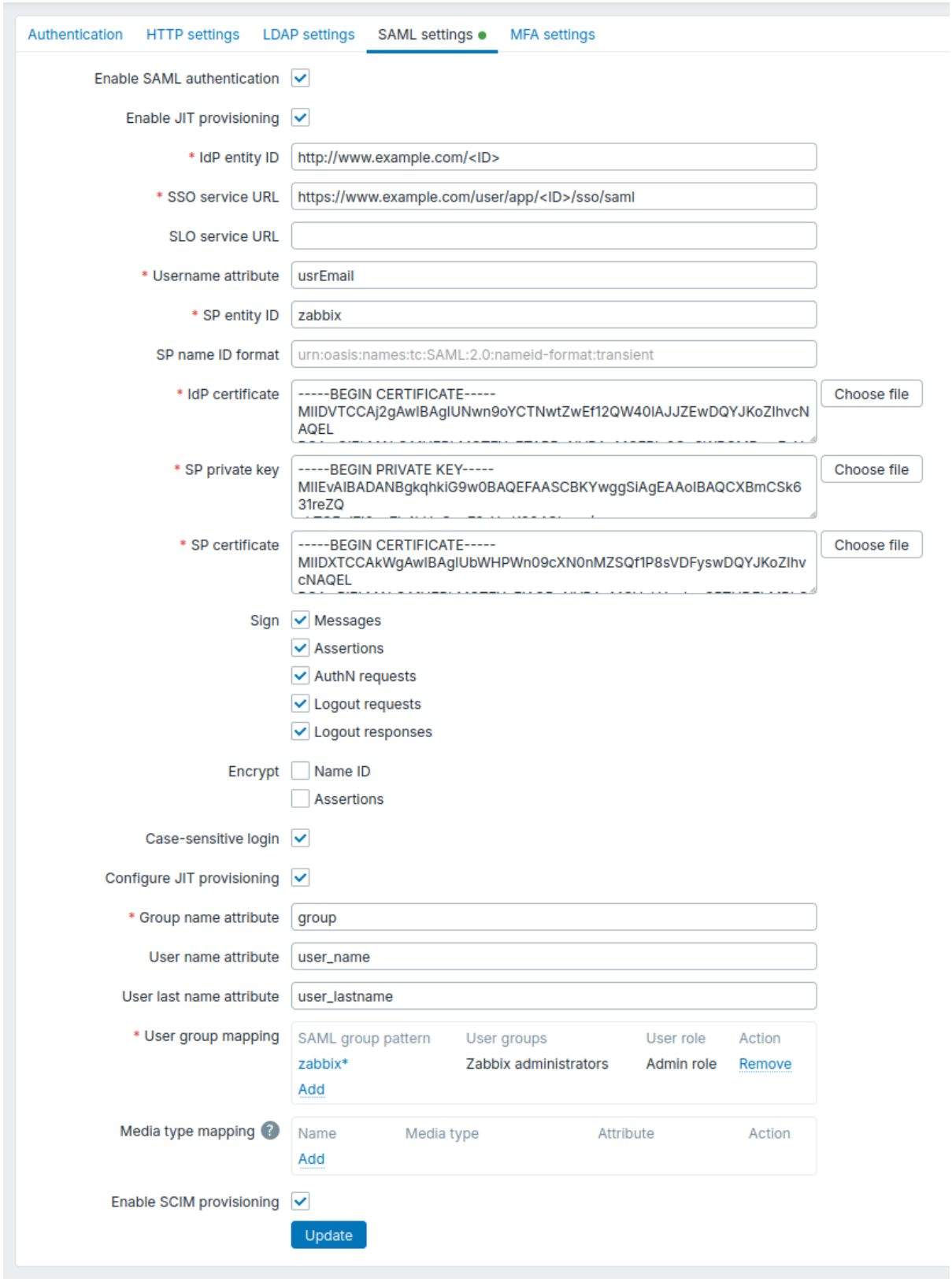
Si se establece `$SSO['CERT_STORAGE'] = 'file'` en el archivo de configuración, el certificado debe estar disponible en el sistema de archivos (de forma predeterminada, en `ui/conf/certs` o en la ruta configurada en el archivo de configuración) y el frontend no almacenará los certificados en la base de datos. Tenga en cuenta que, si `$SSO['CERT_STORAGE']` no está establecido o está comentado, se asumirá el almacenamiento en archivos.

De forma predeterminada, Zabbix buscará en las siguientes ubicaciones:

- ui/conf/certs/sp.key - archivo de clave privada del SP
- ui/conf/certs/sp.crt - archivo de certificado del SP
- ui/conf/certs/idp.crt - archivo de certificado del IdP

> **Attention:**
> El material de los certificados puede importarse mediante el frontend pegando el texto o cargando archivos. Aunque en algunos entornos se permiten importaciones sin cifrar por comodidad, se recomienda encarecidamente utilizar transporte cifrado (por ejemplo, habilitar HTTPS/TLS para el frontend de Zabbix). Cuando los certificados o las claves SAML se almacenan en la base de datos, habilite TLS/SSL para las conexiones de la base de datos con el fin de proteger el tráfico entre la aplicación y la base de datos.
 Cifre las copias de seguridad de la base de datos que puedan contener certificados SAML o claves privadas y limite el acceso a los archivos de copia de seguridad y a las tablas de la base de datos que almacenan material SAML. Restrinja los privilegios del usuario de la base de datos al mínimo necesario.

2. Todos los ajustes más importantes pueden configurarse en el frontend de Zabbix. Sin embargo, es posible especificar ajustes adicionales en el archivo de configuración.



Parámetros de configuración disponibles en el frontend de Zabbix:

Parámetro	Descripción
Habilitar la autenticación SAML	Marque la casilla para habilitar la autenticación SAML.
Habilitar el aprovisionamiento JIT	Marque la casilla para habilitar el aprovisionamiento de usuarios JIT.
ID de entidad del IDP	El identificador único de entidad dentro del proveedor de identidad SAML.
URL del servicio SSO	La URL a la que se redirigirá a los usuarios al iniciar sesión.
URL del servicio SLO	La URL a la que se redirigirá a los usuarios al cerrar sesión. Si se deja vacía, no se utilizará el servicio SLO.

Parámetro	Descripción
Atributo de nombre de usuario	Atributo SAML que se utilizará como nombre de usuario al iniciar sesión en Zabbix. La lista de valores compatibles está determinada por el proveedor de identidad. Ejemplos: uid userprincipalname samaccountname username userusername urn:oid:0.9.2342.19200300.100.1.1 urn:oid:1.3.6.1.4.1.5923.1.1.1.13 urn:oid:0.9.2342.19200300.100.1.44
ID de entidad del SP	El identificador único del proveedor de servicios (si no coincide, se rechazará la operación). Es posible especificar una URL o cualquier cadena de datos.
Formato de ID de nombre del SP	Solicita un formato de ID de nombre específico en la respuesta. Ejemplos: urn:oasis:names:tc:SAML:2.0:nameid-format:persistent urn:oasis:names:tc:SAML:2.0:nameid-format:unspecified urn:oasis:names:tc:SAML:2.0:nameid-format:transient
Certificado del IdP	Contenido del certificado del proveedor de servicios (SP) para la configuración del servicio de inicio de sesión único (SSO) SAML.
Clave privada del SP	Contenido de la clave privada del proveedor de servicios (SP) para la configuración del servicio de inicio de sesión único (SSO) SAML. Proporciona autenticación segura e intercambio de datos con el proveedor de identidad (IdP).
Certificado del SP	Contenido del certificado del proveedor de servicios (SP) para la configuración del servicio de inicio de sesión único (SSO) SAML.
Firmar	Desmarque las casillas para deshabilitar las firmas SAML para las entidades correspondientes: *Mensajes* *Aserciones* *Solicitudes AuthN* *Solicitudes de cierre de sesión* *Respuestas de cierre de sesión*
Cifrar	Marque las casillas para seleccionar las entidades para las que se debe habilitar el cifrado SAML: *ID de nombre* *Aserciones*
Inicio de sesión sensible a mayúsculas y minúsculas	Desmarque la casilla para deshabilitar el inicio de sesión sensible a mayúsculas y minúsculas para los nombres de usuario (habilitado de forma predeterminada). Deshabilitar el inicio de sesión sensible a mayúsculas y minúsculas permite, por ejemplo, iniciar sesión como ”admin” aunque el usuario de Zabbix sea ”Admin” o ”ADMIN”. Tenga en cuenta que, si se deshabilita el inicio de sesión sensible a mayúsculas y minúsculas y existen varios usuarios de Zabbix con nombres de usuario similares (por ejemplo, Admin y admin), el inicio de sesión de esos usuarios siempre se denegará con el siguiente mensaje de error: ”Authentication failed: supplied credentials are not unique.”
Configurar el aprovisionamiento JIT	Marque esta casilla para mostrar las opciones relacionadas con el aprovisionamiento de usuarios JIT.
Atributo del nombre del grupo	Especifique el atributo del nombre del grupo para el aprovisionamiento de usuarios JIT.
Atributo del nombre del usuario	Especifique el atributo del nombre del usuario para el aprovisionamiento de usuarios JIT.
Atributo del apellido del usuario	Especifique el atributo del apellido del usuario para el aprovisionamiento de usuarios JIT.

Parámetro	Descripción
Asignación de grupos de usuarios	Asigne un patrón de grupo de usuarios SAML a un grupo de usuarios y un rol de usuario de Zabbix. Esto es necesario para determinar qué grupo o rol de usuario recibirá el usuario aprovisionado en Zabbix. Haga clic en *Añadir* para añadir una asignación. El campo *Patrón de grupo SAML* admite comodines. El nombre del grupo debe coincidir con un grupo existente. Si un usuario SAML coincide con varios grupos de usuarios de Zabbix, pasará a ser miembro de todos ellos. Si un usuario coincide con varios roles de usuario de Zabbix, obtendrá el nivel de permisos más alto entre ellos.
Asignación de tipos de medios	Asigne los atributos de medios SAML del usuario (por ejemplo, el correo electrónico) a los medios del usuario de Zabbix para enviar notificaciones.
Habilitar el aprovisionamiento SCIM	Marque esta casilla para habilitar el aprovisionamiento SCIM 2.0.

Consulte ejemplos de configuración de proveedores de identidad SAML para iniciar sesión y aprovisionar usuarios en Zabbix con:

- Microsoft Azure AD
- Okta
- Onelogin

Notas sobre el aprovisionamiento SCIM

Para el aprovisionamiento SCIM, especifique la ruta al frontend de Zabbix y añada `api_scim.php` al final, en el lado del proveedor de identidad, es decir:

`https://<path-to-zabbix-ui>/api_scim.php`

Los atributos de usuario que se utilizan en Zabbix (nombre de usuario, nombre, apellido y atributos de medios) deben añadirse como atributos personalizados y, si es necesario, el espacio de nombres externo debe ser el mismo que el esquema de usuario: `urn:ietf:params:scim:schemas:core:2.0:User`.

Configuración avanzada

Se pueden configurar parámetros SAML adicionales en el archivo de configuración del frontend de Zabbix (`zabbix.conf.php`):

- `$SSO['SP_KEY']`
- `$SSO['SP_CERT']`
- `$SSO['IDP_CERT']`
- `$SSO['SETTINGS']`

Ejemplos de configuración

Ejemplo 1: Conexión a una instancia de Zabbix con un balanceador de carga mediante TLS/SSL

Si se utiliza un balanceador de carga para conectarse a una instancia de Zabbix y el balanceador de carga utiliza TLS/SSL mientras que Zabbix no lo utiliza, especifique los parámetros `baseurl`, `strict` y `use_proxy_headers` de la siguiente manera:

```
$SSO['SETTINGS'] = [
    'strict' => false,
    'baseurl' => 'https://zabbix.example.com/zabbix/',
    'use_proxy_headers' => true
];
```

Ejemplo 2: Configuración de algoritmos personalizados de firma y resumen SAML

```
$SSO['SETTINGS'] = [
    'security' => [
        'signatureAlgorithm' => 'http://www.w3.org/2001/04/xmldsig-more#rsa-sha384'
        'digestAlgorithm' => 'http://www.w3.org/2001/04/xmldsig-more#sha384',
        // ...
    ],
    // ...
];
```

Ejemplo 3: Configuración de SSO con autenticación Kerberos mediante ADFS

```
$SSO['SETTINGS'] = [
    'security' => [
        'requestedAuthnContext' => [
            'urn:oasis:names:tc:SAML:2.0:ac:classes:Kerberos',
        ],
        'requestedAuthnContextComparison' => 'exact'
    ]
];
```

En este caso, en el campo *SP name ID* de la configuración SAML, establezca:

```
urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified
```

MFA

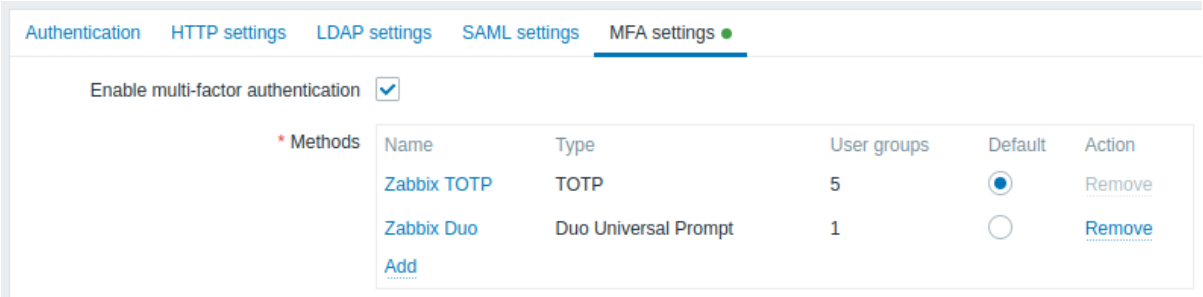
Descripción general

La autenticación multifactor (MFA) se puede utilizar para iniciar sesión en Zabbix, proporcionando una capa adicional de seguridad más allá de solo un nombre de usuario y contraseña.

Con MFA, el usuario debe existir en Zabbix, debe proporcionar las credenciales de Zabbix al iniciar sesión y también debe demostrar su identidad por otros medios, generalmente, un código generado por una aplicación autenticadora en el teléfono del usuario.

Hay disponibles varios métodos de MFA, lo que permite a los usuarios elegir la opción que mejor se adapte a sus requisitos de seguridad y preferencias. Estos métodos son Contraseña de un solo uso basada en tiempo (TOTP) y Duo Universal Prompt.

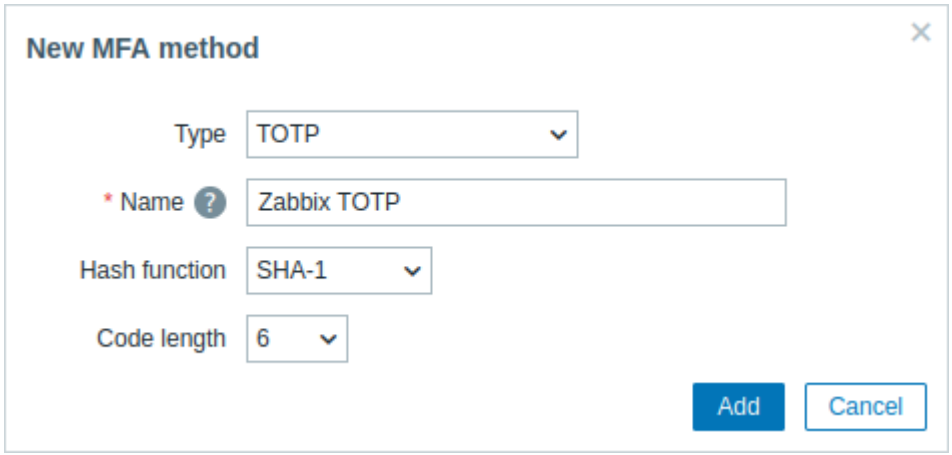
Configuración



Parámetros de configuración:

Parámetro	Descripción
Habilitar autenticación multifactor	Marque la casilla para habilitar la autenticación multifactor.
Métodos	Haga clic en *Añadir* para configurar un método MFA (consulte la configuración del método a continuación).

Configuración del método



Parámetros de configuración del método:

Parámetro	Descripción
Tipo	Seleccione el tipo de método MFA: **TOTP**: utilice una aplicación de autenticación para generar contraseñas de un solo uso basadas en el tiempo; **Duo Universal Prompt**: utilice el servicio de autenticación Duo para proporcionar autenticación multifactor.
Nombre	Introduzca un nombre que se mostrará como el nombre de la cuenta a todos los usuarios MFA en las aplicaciones de autenticación (por ejemplo, ”Zabbix”).
Función hash	Seleccione la función hash (SHA-1, SHA-256 o SHA-512) para generar los códigos TOTP. Este parámetro está disponible si el tipo de método MFA está configurado en ”TOTP”. Tenga en cuenta que elegir SHA-256 o SHA-512 puede limitar significativamente la compatibilidad, ya que muchas aplicaciones actualmente no admiten estas funciones.
Longitud del código	Seleccione la longitud del código de verificación (6 u 8). Este parámetro está disponible si el tipo de método MFA está configurado en ”TOTP”.
API hostname	Introduzca el nombre de equipo de la API proporcionado por el servicio de autenticación Duo. Este parámetro está disponible si el tipo de método MFA está configurado en ”Duo Universal Prompt”.
Client ID	Introduzca el ID de cliente proporcionado por el servicio de autenticación Duo. Este parámetro está disponible si el tipo de método MFA está configurado en ”Duo Universal Prompt”.
Client secret	Introduzca el secreto de cliente proporcionado por el servicio de autenticación Duo. Este parámetro está disponible si el tipo de método MFA está configurado en ”Duo Universal Prompt”.

Ejemplos de configuración

Esta sección proporciona ejemplos de configuración de MFA utilizando Contraseña de un solo uso basada en tiempo (TOTP) y Duo Universal Prompt.

TOTP

Para TOTP, los usuarios deben verificar su identidad mediante una aplicación autenticadora (por ejemplo, la aplicación Google Authenticator).

1. Vaya a la configuración de MFA en Zabbix, en *Users* > *Authentication*, y habilite la autenticación multifactor.

2. Añada un nuevo método de MFA con la siguiente configuración:

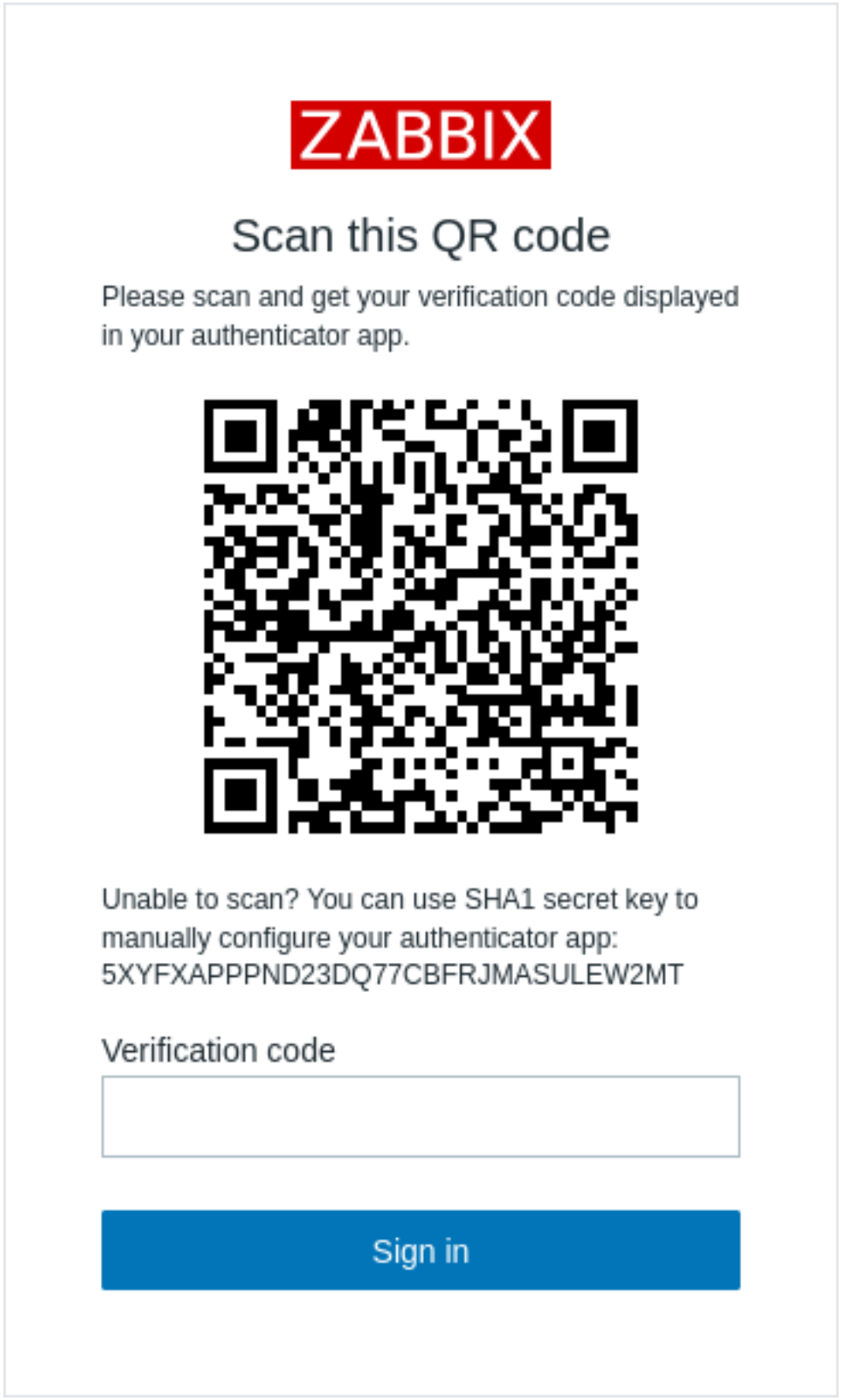
- Type: TOTP
- Name: Zabbix TOTP
- Hash function: SHA-1
- Code length: 6

3. Haga clic en el botón *Add* y luego en el botón *Update*.

4. Vaya a *Users* > *User groups* y cree un nuevo grupo de usuarios con la siguiente configuración:

- Group name: TOTP group
- Users: Admin
- Multi-factor authentication: Default (o ”Zabbix TOTP” si no está establecido como predeterminado)

5. Cierre sesión en Zabbix y vuelva a iniciarla con sus credenciales. Tras iniciar sesión correctamente, se le pedirá que se inscriba en MFA, mostrando un código QR y una clave secreta.



6. Escanee el código QR o introduzca la clave secreta en la aplicación Google Authenticator. La aplicación generará un código de verificación que deberá introducir para completar el proceso de inicio de sesión.

7. En los inicios de sesión posteriores, obtenga el código de verificación de la aplicación Google Authenticator e introdúzcalo durante el inicio de sesión.

Duo Universal Prompt

Para Duo Universal Prompt, los usuarios deben verificar su identidad usando la aplicación autenticadora Duo Mobile.

Attention:
El método MFA Duo Universal Prompt requiere la instalación de la extensión php-curl, acceso a Zabbix mediante HTTPS y permiso para conexiones salientes a los servidores de Duo. Además, si ha habilitado Content Security Policy (CSP) en el servidor web, asegúrese de agregar "duo.com" a la directiva CSP en el archivo de configuración de su virtual host.

1. Regístrese para obtener una cuenta gratuita de administrador de Duo en Duo Signup.

2. Abra el panel de administración de Duo, vaya a *Applications* > *Protect an Application*, busque la aplicación *Web SDK* y haga clic en *Protect*.

3. Anote las credenciales (Client ID, Client secret, API hostname) necesarias para configurar el método MFA en Zabbix.

4. Vaya a la configuración de MFA en Zabbix en *Users* > *Authentication* y habilite la autenticación multifactor.

5. Añada un nuevo método MFA con la siguiente configuración:

- Type: Duo Universal Prompt
- Name: Zabbix Duo
- API hostname: (use API hostname from Duo)
- Client ID: (use Client ID from Duo)
- Client secret: (use Client secret from Duo)

6. Haga clic en el botón *Add* y luego en el botón *Update*.

7. Vaya a *Users* > *User groups* y cree un nuevo grupo de usuarios con la siguiente configuración:

- Group name: Duo group
- Users: Admin
- Multi-factor authentication: Default (or "Zabbix Duo" if it is not set as default)

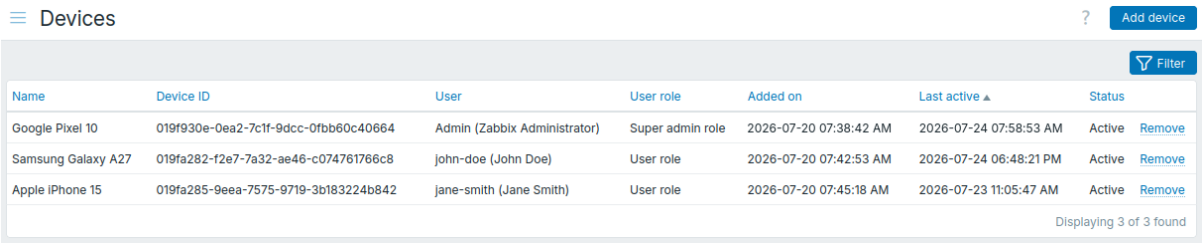
8. Cierre sesión en Zabbix y vuelva a iniciarla con sus credenciales. Tras iniciar sesión correctamente, se le pedirá que se inscriba en MFA y se le redirigirá a Duo. Complete la configuración de Duo y verifique su usuario con la aplicación Duo de su teléfono para iniciar sesión.

9. Para los inicios de sesión posteriores, use el método MFA adecuado proporcionado por la aplicación Duo (por ejemplo, obtener un código de verificación, responder a notificaciones push o usar llaves de seguridad) e introduzca la información requerida durante el inicio de sesión.

Dispositivos

Esta sección permite a los superadministradores gestionar los dispositivos móviles que se utilizarán con la aplicación móvil de Zabbix.

Esta sección se muestra si `EnableMobileDevices` está habilitado.



Para añadir un dispositivo:

1. Haga clic en *Añadir dispositivo*.
2. Seleccione un usuario y, a continuación, haga clic en *Añadir*. El usuario debe tener habilitado el acceso a los dispositivos.
3. Aparecerá un código QR. El usuario debe escanear este código en su aplicación móvil de Zabbix.

Note:
Al añadir un dispositivo para un usuario, Zabbix también añade el tipo de medio de notificaciones push a los medios del usuario. Al eliminar el dispositivo, también se eliminan esos medios del usuario. Si tiene configurados varios tipos de medios de notificaciones push, Zabbix añade el que está habilitado y aparece primero alfabéticamente.

Los dispositivos añadidos aparecen en una lista que puede filtrar.

También puede eliminar dispositivos. Al eliminar un dispositivo, sus credenciales de acceso se revocan y eliminan de Zabbix, tras lo cual ya no podrá acceder a Zabbix.

La lista muestra estos detalles:

Column	Description
Name	Nombre del dispositivo.
Device ID	ID del dispositivo.
User	Usuario para el que se añadió el dispositivo, o *Usuario inaccesible* si este usuario ha sido eliminado.
User role	Rol de usuario.
Added on	Fecha y hora en que se añadió el dispositivo.
Last active	Fecha y hora en que el dispositivo estuvo activo por última vez.
Status	Estado del dispositivo: **Nuevo**: el dispositivo se añadió sin escanear el código QR y deberá añadirse de nuevo. **Nuevo (token caducado)**: el dispositivo se añadió sin escanear el código QR y el token del código QR ha caducado; el dispositivo deberá añadirse de nuevo. **Activo**: el dispositivo se añadió, se escaneó el código QR y puede utilizar la aplicación móvil de Zabbix. **Sin asignar**: el dispositivo se añadió, pero su usuario fue eliminado y no puede utilizar la aplicación móvil de Zabbix.

Los usuarios con permisos suficientes también pueden gestionar sus propios dispositivos en *Configuración del usuario* > *Dispositivos*.

Administración

Descripción general

El menú Administración es para funciones administrativas de Zabbix. Este menú está disponible solo para usuarios del tipo de usuario SuperAdmin.

Fuente de datos

Descripción general

La sección *Administración* > *Fuente de datos* permite especificar la fuente de datos de telemetría, es decir, la base de datos que contiene los datos de telemetría (trazas, métricas y registros). Los datos de telemetría se utilizan para la supervisión del rendimiento de las aplicaciones (APM), lo que permite ver cómo fluyen las solicitudes a través de sus aplicaciones y servicios, medir su rendimiento y encontrar la causa de las operaciones lentas o fallidas.

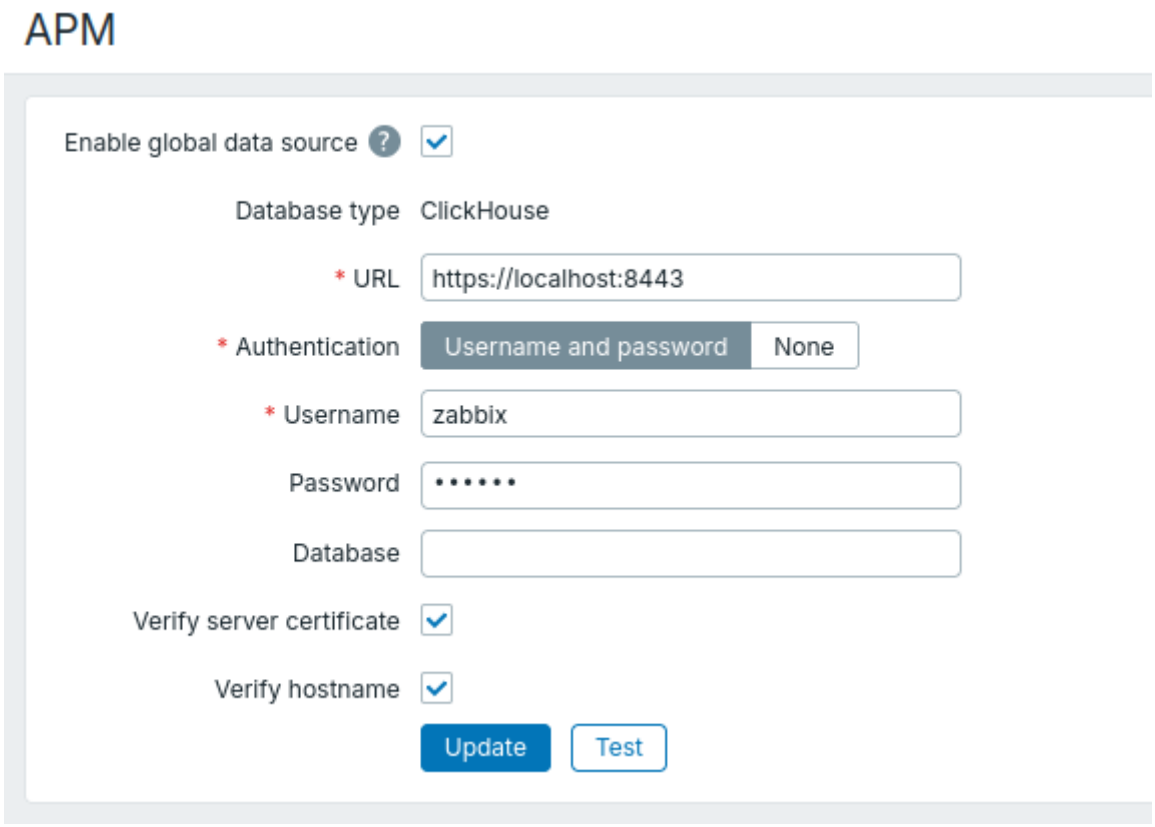
La fuente de datos de telemetría debe ser compatible con OpenTelemetry.

La fuente de datos de telemetría especificada en esta sección se denomina fuente de datos global, porque es compartida por Zabbix server, proxy y frontend. Sin embargo, puede anularla para cualquiera de estos componentes especificando una fuente de datos de telemetría diferente en la configuración de server, proxy o frontend.

El item de consulta de telemetría consulta esta fuente de datos, a menos que se especifique otra en la configuración del componente.

APM

Para especificar la fuente de datos global, marque la casilla *Enable global data source* y complete los parámetros descritos a continuación. Antes de guardar, puede hacer clic en *Test* para comprobar la conexión con esta fuente de datos (requiere la extensión php-curl).



Parameter	Description
Database type	El tipo de base de datos está establecido de forma fija en *ClickHouse* y no se puede cambiar. Consulte también: Requirements y ClickHouse setup.
URL	Introduzca la URL HTTP o HTTPS utilizada para conectarse a la base de datos.
Authentication	Seleccione el método de autenticación utilizado para iniciar sesión en la base de datos: *Username and password* o *None*.
Username	Especifique el nombre de usuario utilizado para iniciar sesión en la base de datos. Este parámetro está disponible si *Authentication* está establecido en *Username and password*.
Password	Especifique la contraseña utilizada para iniciar sesión en la base de datos o déjelo vacío si el usuario de la base de datos no tiene contraseña. La contraseña se restablece al cambiar la URL. Este parámetro está disponible si *Authentication* está establecido en *Username and password*.
Database	Introduzca el nombre de la base de datos que contiene los datos de telemetría. Si se deja vacío, Zabbix utiliza la base de datos predeterminada configurada en ClickHouse para el usuario.
Verify server certificate	Marque la casilla para verificar el certificado del server. Si está marcada, las opciones TLS relacionadas deben configurarse en la configuración del server, del proxy o del frontend. Este parámetro está disponible si *URL* contiene una URL HTTPS.
Verify hostname	Marque la casilla para verificar que el nombre de host de *URL* coincida con el nombre de host del certificado del server. Si está marcada, las opciones TLS relacionadas deben configurarse en la configuración del server, del proxy o del frontend. Este parámetro está disponible si *URL* contiene una URL HTTPS.

General

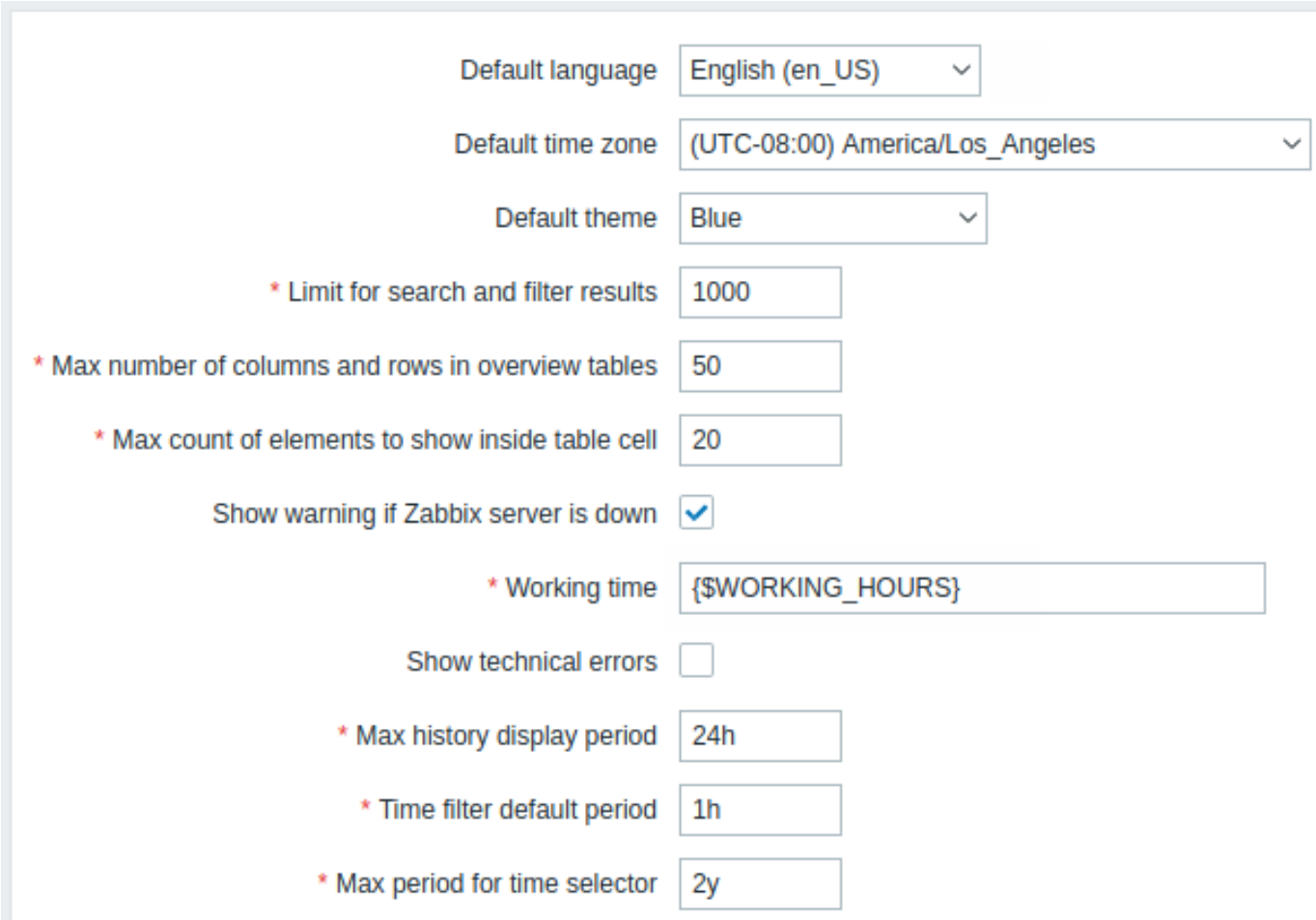
Descripción general

La sección *Administration* > *General* contiene varias subsecciones para configurar los valores predeterminados relacionados con el frontend y personalizar Zabbix.

La lista de subsecciones disponibles aparece al hacer clic en *General* en la sección del menú *Administration*. También es posible cambiar entre subsecciones usando el menú desplegable del título en la esquina superior izquierda.

GUI

Esta sección proporciona la personalización de varios valores predeterminados relacionados con el frontend.



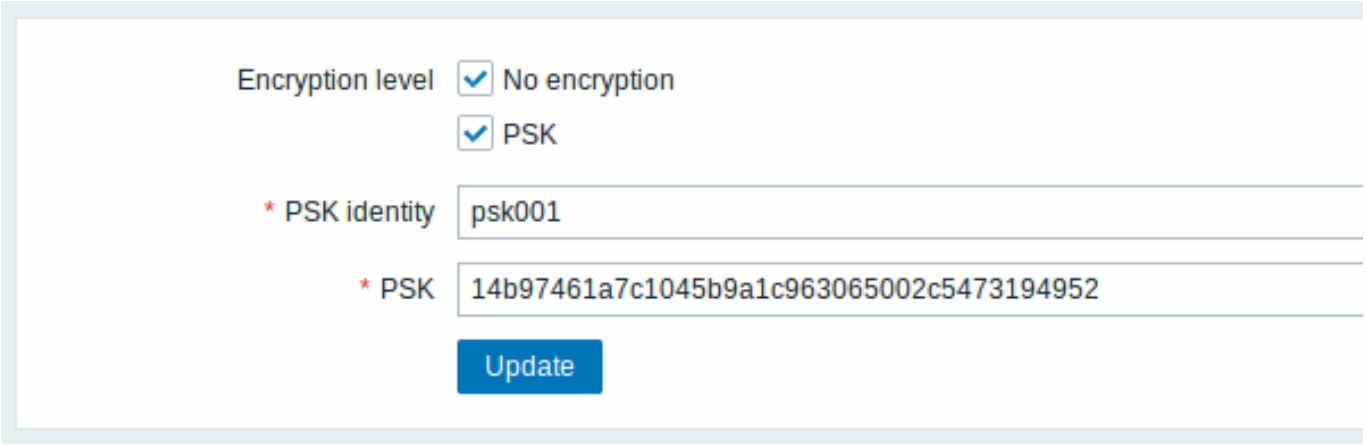
Parámetros de configuración:

Parameter	Description
Default language	Idioma predeterminado para los usuarios que no han especificado un idioma en sus perfiles y para los usuarios invitados. Para más información, consulte Instalación de idiomas adicionales del frontend.
Default time zone	Zona horaria predeterminada para los usuarios que no han especificado una zona horaria en sus perfiles y para los usuarios invitados.
Default theme	Tema predeterminado para los usuarios que no han especificado un tema en sus perfiles y para los usuarios invitados.
Limit for search and filter results	Cantidad máxima de elementos (filas) que se mostrarán en una lista de la interfaz web, por ejemplo, en *Data collection* > *Hosts*. *Nota*: Si se establece, por ejemplo, en '50', solo se mostrarán los primeros 50 elementos en todas las listas del frontend afectadas. Si alguna lista contiene más de cincuenta elementos, la indicación de ello será el signo '+' en *"Displaying 1 to 50 of **50+** found"*. Además, si se usa el filtrado y aun así hay más de 50 coincidencias, solo se mostrarán las primeras 50. Tenga en cuenta que aumentar el valor de este parámetro puede provocar una disminución del rendimiento y un mayor consumo de memoria en el lado del servidor web.
Max number of columns and rows in overview tables	Número máximo de columnas y filas que se mostrarán en el widget de panel *Trigger overview*. El mismo límite se aplica tanto a las columnas como a las filas. Si existen más filas y/o columnas de las que se muestran, el sistema mostrará una advertencia en la parte inferior de la tabla: "Not all results are displayed. Please provide more specific search criteria."
Max count of elements to show inside table cell	Para las entradas que se muestran en una sola celda de tabla, no se mostrará más de lo configurado aquí.
Show warning if Zabbix server is down	Este parámetro permite mostrar un mensaje de advertencia en una ventana del navegador si no se puede acceder al server de Zabbix (posiblemente esté caído). El mensaje permanece visible incluso si el usuario se desplaza hacia abajo en la página. Al pasar el cursor sobre él, el mensaje se oculta temporalmente para revelar el contenido que hay debajo.

Parameter	Description
Working time	Este parámetro global define el horario laboral. En los gráficos, el tiempo de trabajo se muestra con un fondo blanco y el tiempo no laborable con un fondo gris. Consulte la página Time period specification para obtener una descripción del formato de tiempo. Se admiten macros de usuario.
Show technical errors	Si está marcada, todos los usuarios registrados podrán ver errores técnicos (PHP/SQL). Si no está marcada, la información solo está disponible para los Zabbix Super Admins y para los usuarios que pertenecen a grupos de usuarios con el modo de depuración habilitado.
Max history display period	Período máximo durante el cual se mostrarán los datos históricos en *Monitoring* > *Latest data*, la información de web scenario del host en *Monitoring* > *Hosts* y en el widget de panel *Top items*. Rango permitido: 24 horas (predeterminado) - 1 semana. Se admiten sufijos de tiempo, por ejemplo, 1w (una semana), 36h (36 horas).
Time filter default period	Período de tiempo que se usará de forma predeterminada en gráficos y paneles. Rango permitido: 1 minuto - 10 años (predeterminado: 1 hora). Se admiten sufijos de tiempo, por ejemplo, 10m (diez minutos), 5w (cinco semanas). Nota: cuando un usuario cambia el período de tiempo mientras visualiza un gráfico, este período se guarda como preferencia del usuario, sustituyendo el valor predeterminado global o una selección anterior del usuario.
Max period for time selector	Período máximo disponible para gráficos y paneles. Los usuarios no podrán visualizar datos más antiguos. Rango permitido: 1 año - 10 años (predeterminado: 2 años). Se admiten sufijos de tiempo, por ejemplo, 1y (un año), 365w (365 semanas).

Autoregistro

En esta sección, puede configurar el nivel de cifrado para el autoregistro del agent activo.



Los parámetros marcados con un asterisco son obligatorios.

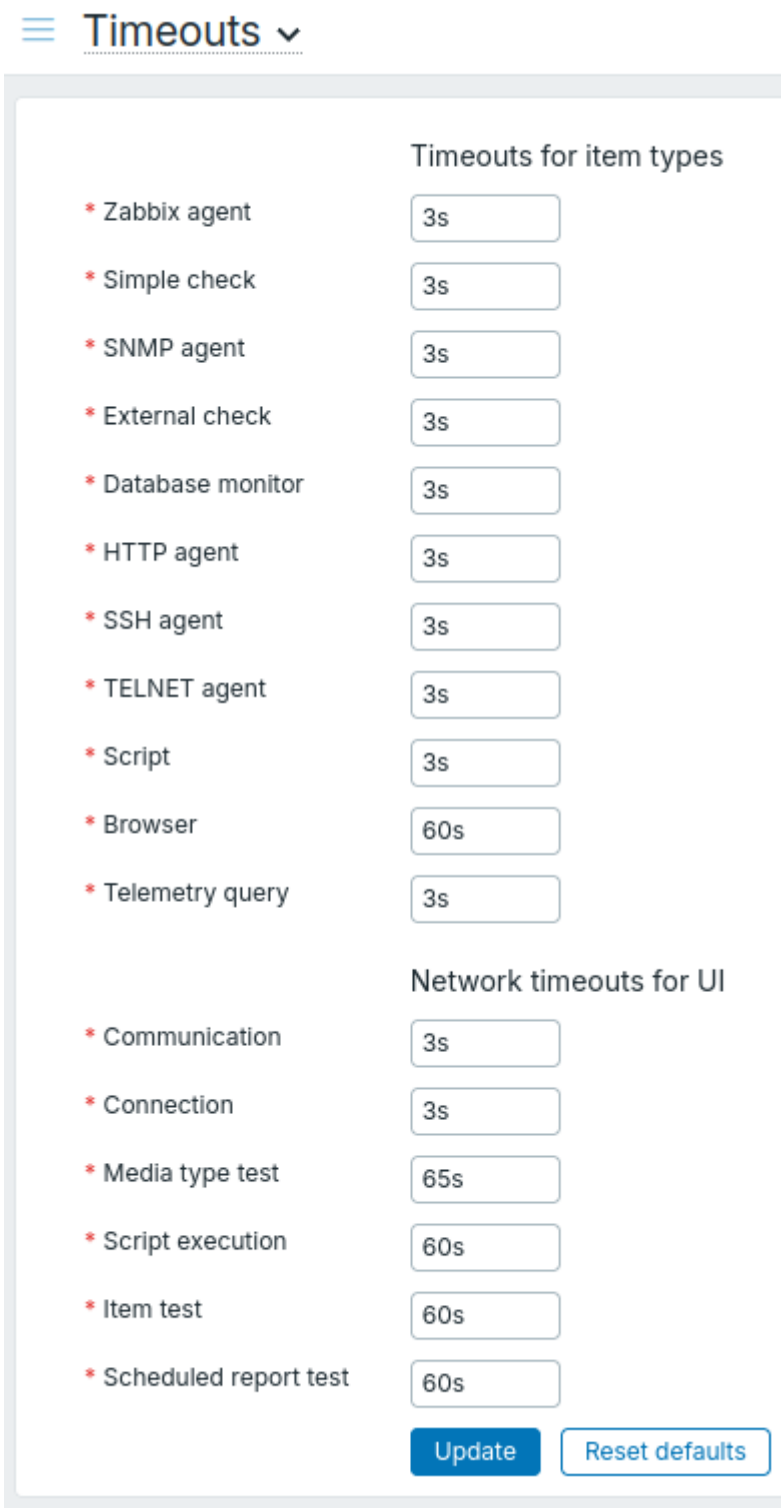
Parámetros de configuración:

Parameter	Description
Nivel de cifrado	Seleccione una o ambas opciones para el nivel de cifrado: **Sin cifrado** - se permiten conexiones sin cifrar **PSK** - se permiten conexiones cifradas con TLS mediante una clave precompartida
Identidad PSK	Introduzca la cadena de identidad de la clave precompartida. Este campo solo está disponible si se selecciona 'PSK' como *Nivel de cifrado*. No coloque información sensible en la identidad PSK; se transmite sin cifrar por la red para informar al receptor qué PSK debe usar.
PSK	Introduzca la clave precompartida (un número par de caracteres hexadecimales). Longitud máxima: 512 dígitos hexadecimales (PSK de 256 bytes) si Zabbix usa la biblioteca GnuTLS u OpenSSL, 64 dígitos hexadecimales (PSK de 32 bytes) si Zabbix usa la biblioteca mbed TLS (PolarSSL). Ejemplo: 1f87b595725ac58dd977beef14b97461a7c1045b9a1c963065002c5473194952 Este campo solo está disponible si se selecciona 'PSK' como *Nivel de cifrado*.

Véase también: Autoregistro seguro

Timeouts

En esta sección, es posible establecer timeouts globales para tipos de item y timeouts de red. Todos los campos de este formulario son obligatorios.

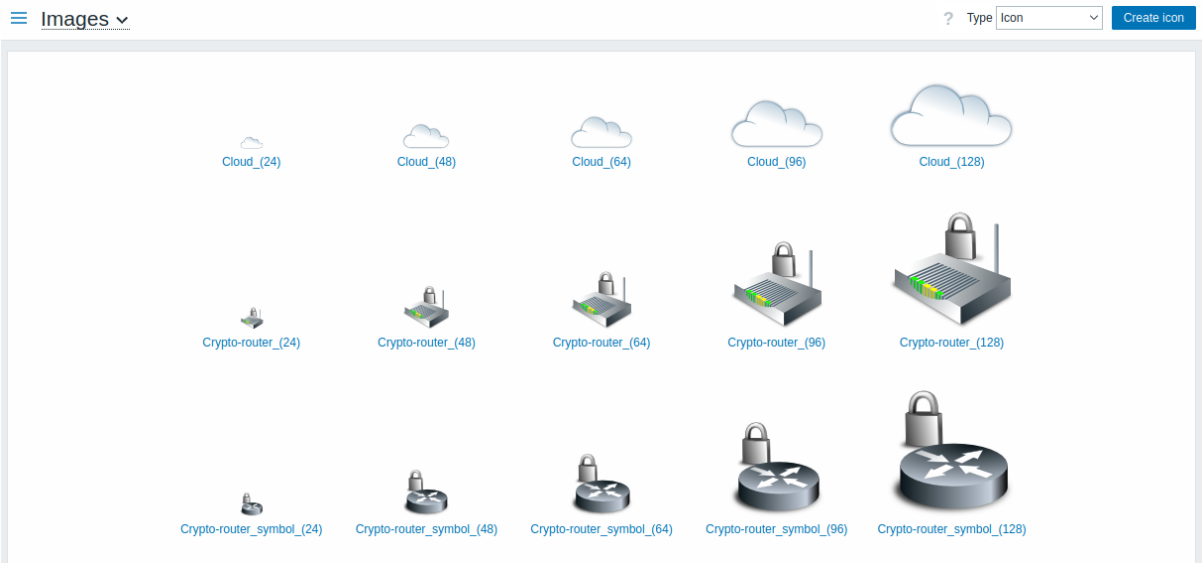


Parameter	Description
Timeouts for item types	Cuántos segundos debe esperar Zabbix antes de abortar la comprobación (según su tipo). Para obtener detalles adicionales y limitaciones, consulte item timeout. Rango permitido: 1 - 600s (predeterminado: 3s; predeterminado para items de browser: 60s). Se admiten sufijos de tiempo, por ejemplo, 30s, 1m, y macros de usuario. Tenga en cuenta que los timeouts a nivel de proxy, si están configurados, anulan estos ajustes globales. Además, si se configuran timeouts para items específicos, estos anulan tanto los ajustes globales como los del proxy.
Network timeouts for UI	

Parameter		Description
	Communication	Cuántos segundos esperar antes de cerrar un socket inactivo (si se estableció previamente una conexión con Zabbix server, pero frontend no puede completar la operación de lectura/envío de datos durante este tiempo, la conexión se cerrará). Rango permitido: 1 - 300s (predeterminado: 3s).
	Connection	Cuántos segundos esperar antes de detener un intento de conexión con Zabbix server. Rango permitido: 1 - 30s (predeterminado: 3s).
	Media type test	Cuántos segundos esperar una respuesta al probar un tipo de medio. Rango permitido: 1 - 300s (predeterminado: 65s).
	Script execution	Cuántos segundos esperar una respuesta al ejecutar un script. Rango permitido: 1 - 300s (predeterminado: 60s). Este timeout se aplica a toda la cadena de ejecución del script, que puede tener distinta duración. Por ejemplo, si el script se ejecuta en el agent, es un recorrido de ida y vuelta a través de server (posiblemente también proxy) hasta el agent y de regreso. Para los scripts que se ejecutan en el agent, se aplica el timeout de server o proxy. Para los scripts que se ejecutan solo en un agent activo, es probable que sea necesario aumentar el timeout predeterminado de server/proxy. El timeout de server/proxy debe ser mayor que la frecuencia de actualización de la comprobación activa; de lo contrario, el timeout de server/proxy puede superarse antes de que el agent activo incluso reciba el script. Consulte también: Script timeout.
	Item test	Cuántos segundos esperar los datos devueltos al probar un item. Rango permitido: 1 - 600s (predeterminado: 60s).
	Scheduled report test	Cuántos segundos esperar los datos devueltos al probar un informe programado. Rango permitido: 1 - 300s (predeterminado: 60s).

Imágenes

La sección Imágenes muestra todas las imágenes disponibles en Zabbix.
Las imágenes se almacenan en la base de datos.

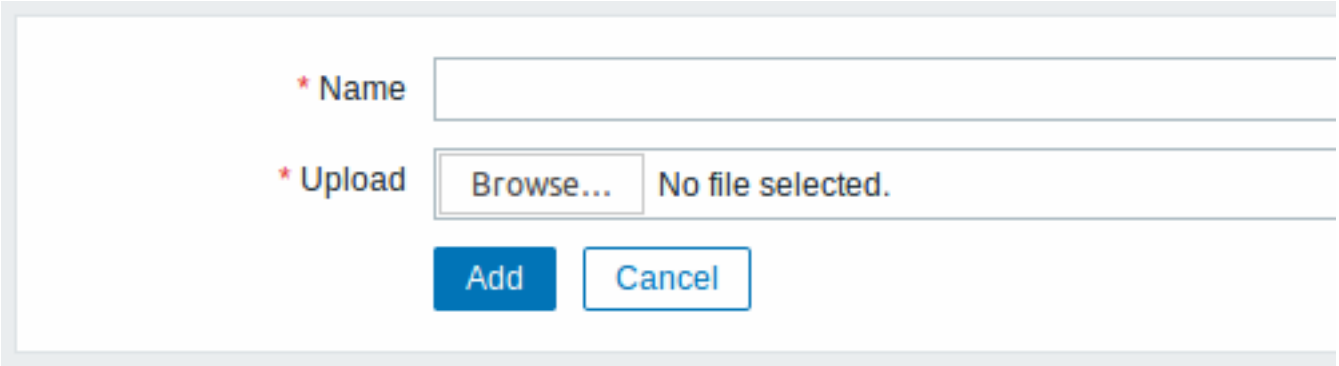


La lista desplegable *Tipo* permite cambiar entre imágenes de iconos e imágenes de fondo:

- Los iconos se utilizan para mostrar elementos de mapas de red.
- Los fondos se utilizan como imágenes de fondo de los mapas de red.

Adición de una imagen

Puede añadir su propia imagen haciendo clic en el botón *Crear icono* o *Crear fondo* en la esquina superior derecha.



Atributos de la imagen:

Parámetro	Descripción
Nombre	Nombre único de una imagen.
Carga	Seleccione el archivo (PNG, JPEG, GIF o WebP) de un sistema local para cargarlo en Zabbix. *Nota*: es posible que se puedan cargar otros formatos que se convertirán a PNG durante la carga. La biblioteca GD se utiliza para procesar imágenes; por lo tanto, los formatos compatibles dependen de la versión de la biblioteca utilizada (Zabbix requiere la versión 2.0.28 o posterior).

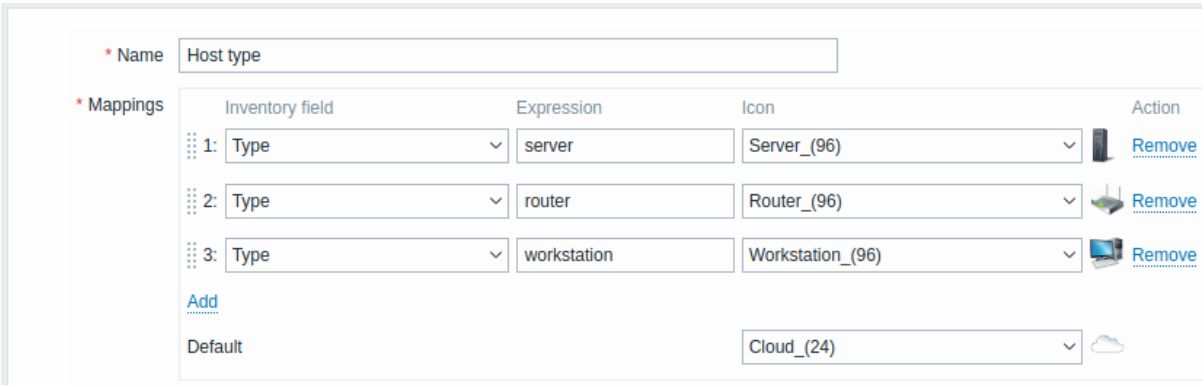
Note:
El tamaño máximo del archivo de carga está limitado por el valor de `ZBX_MAX_IMAGE_SIZE`, que es de 1024x1024 bytes o 1 MB.
La carga de una imagen puede fallar si su tamaño se aproxima a 1 MB y el parámetro de configuración de MySQL `max_allowed_packet` tiene el valor predeterminado de 1 MB. En este caso, aumente el valor del parámetro `max_allowed_packet`.

Mapeo de iconos

Esta sección permite crear el mapeo de ciertos hosts con ciertos iconos. La información del campo de inventario del host se utiliza para crear el mapeo.

Luego, los mapeos se pueden usar en network map configuration para asignar automáticamente los iconos adecuados a los hosts coincidentes.

Para crear un nuevo mapa de iconos, haga clic en *Create icon map* en la esquina superior derecha.



Parámetros de configuración:

Parameter	Description
Name	Nombre único del mapa de iconos.
Mappings	Lista de mapeos. El orden de los mapeos determina cuál tendrá prioridad. Puede mover los mapeos hacia arriba y hacia abajo en la lista mediante arrastrar y soltar.
Inventory field	Campo de inventario del host que se examinará para buscar una coincidencia.
Expression	Expresión regular que describe la coincidencia.
Icon	Icono que se usará si se encuentra una coincidencia para la expresión.
Default	Icono predeterminado que se usará.

Expresiones regulares

Esta sección permite crear expresiones regulares personalizadas que se pueden usar en varios lugares del frontend. Consulte la sección Expresiones regulares para obtener más detalles.

Opciones de visualización de trigger

Esta sección permite personalizar cómo se muestra el estado de un trigger en el frontend y los nombres y colores de severidad de trigger.

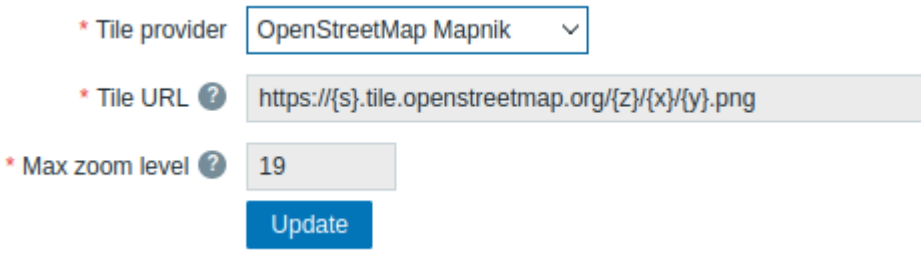


Parameter	Description
Use custom event status colors *Unacknowledged PROBLEM events*, *Acknowledged PROBLEM events*, *Unacknowledged RESOLVED events*, *Acknowledged RESOLVED events*	Al marcar este parámetro, se activa la personalización de colores para problemas reconocidos/no reconocidos. Introduzca un nuevo código de color o haga clic en el color para seleccionar uno nuevo de la paleta proporcionada. Si la casilla *blinking* está marcada, los triggers parpadearán durante un tiempo cuando cambie su estado para ser más visibles.
Display OK triggers for	Período de tiempo para mostrar triggers OK. Rango permitido: 0 - 24 horas. Se admiten sufijos de tiempo, por ejemplo 5m, 2h, 1d.
On status change triggers blink for	Duración del parpadeo del trigger. Rango permitido: 0 - 24 horas. Se admiten sufijos de tiempo, por ejemplo 5m, 2h, 1d.
Not classified, *Information*, *Warning*, *Average*, *High*, *Disaster*	Nombres y/o colores personalizados de severidad para mostrar en lugar de los valores predeterminados del sistema. Introduzca un nuevo código de color o haga clic en el color para seleccionar uno nuevo de la paleta proporcionada. Tenga en cuenta que los nombres personalizados de severidad introducidos aquí se usarán en todos los idiomas. Si necesita traducirlos a otros idiomas para determinados usuarios, consulte la página Personalización de severidades de trigger.

Mapas geográficos

Esta sección permite seleccionar el proveedor del servicio de teselas del mapa geográfico y configurar los ajustes del proveedor de servicios para el widget de dashboard Geomap dashboard widget. Para proporcionar visualización mediante los mapas geográficos, Zabbix utiliza la biblioteca JavaScript de código abierto Leaflet para mapas interactivos. Tenga en cuenta que Zabbix no tiene

control sobre la calidad de las imágenes proporcionadas por proveedores de teselas de terceros, incluidos los proveedores de teselas predefinidos.



Parameter	Description
Proveedor de teselas	Seleccione uno de los proveedores de servicio de teselas disponibles o seleccione *Other* para añadir otro proveedor de teselas o teselas alojadas por usted mismo (consulte Using a custom tile service provider).
URL de teselas	La plantilla de URL (hasta 2048 caracteres) para cargar y mostrar la capa de teselas en los mapas geográficos. Este campo solo es editable si *Tile provider* está configurado como *Other*. Se admiten los siguientes marcadores de posición: *{s}* representa uno de los subdominios disponibles; *{z}* representa el parámetro de nivel de zoom en la URL; *{x}* y *{y}* representan las coordenadas de la tesela; *{r}* se puede usar para añadir "@2x" a la URL y cargar teselas retina. Ejemplo: `https://{s}.example.com/{z}/{x}/{y}{r}.png`
Texto de atribución	Texto de atribución del proveedor de teselas que se mostrará en un pequeño cuadro de texto en el mapa. Este campo solo es visible si *Tile provider* está configurado como *Other*.
Nivel máximo de zoom	Nivel máximo de zoom del mapa. Este campo solo es editable si *Tile provider* está configurado como *Other*.

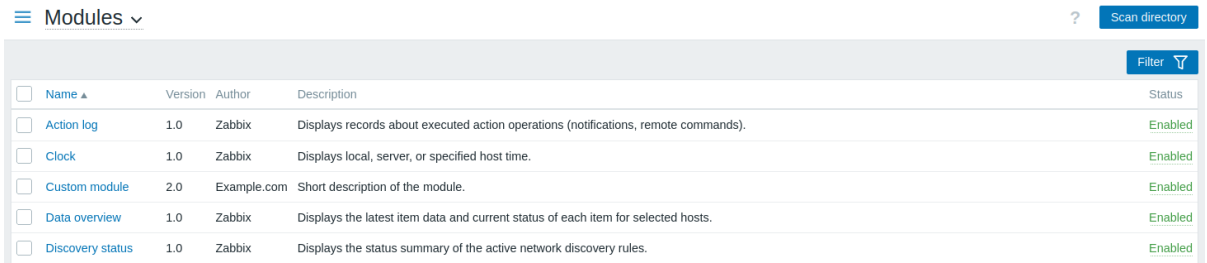
Uso de un proveedor de servicio de mosaicos personalizado

El widget Geomap es capaz de cargar imágenes de mosaicos raster desde un servicio de proveedor de mosaicos personalizado alojado por usted mismo o de terceros. Para usar un servicio de proveedor de mosaicos de terceros personalizado o una carpeta o server de mosaicos alojada por usted mismo, seleccione *Other* en el campo *Tile provider* y especifique la URL personalizada en el campo *Tile URL* usando los marcadores de posición adecuados.

Módulos

Esta sección permite administrar módulos del frontend personalizados, así como integrados.

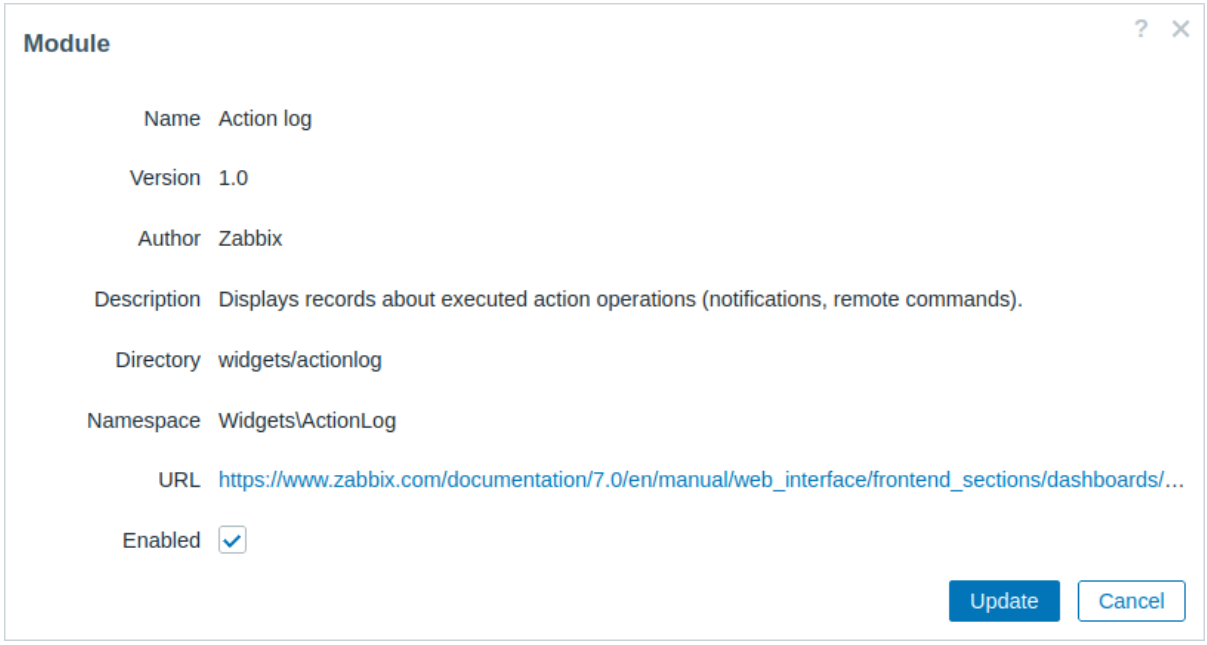
Note:
Si el indicador de funcionalidad `$ZBX_FEATURE_FLAGS['modules_config_enabled']` está establecido en `false` en el archivo de configuración del frontend, el acceso a esta sección está deshabilitado.



Haga clic en *Escanear directorio* para registrar/anular el registro de cualquier módulo personalizado. Los módulos registrados aparecerán en la lista; los módulos cuyo registro se haya anulado se eliminarán de la lista.

Haga clic en el estado del módulo en la lista para habilitarlo/deshabilitarlo. También puede habilitar/deshabilitar varios módulos a la vez seleccionándolos en la lista y haciendo clic en los botones *Habilitar/Deshabilitar* situados debajo de la lista.

Haga clic en el nombre del módulo en la lista para ver sus detalles en una ventana emergente.

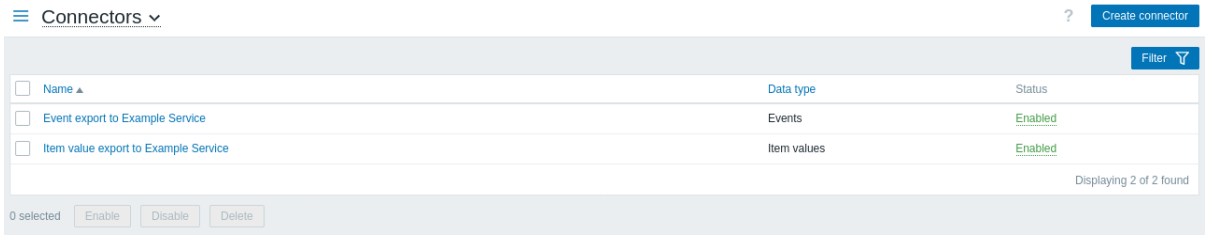


El estado del módulo también se puede actualizar en la ventana emergente de detalles del módulo; para ello, marque/desmarque la casilla *Habilitado* y haga clic en *Actualizar*.

Puede filtrar los módulos por nombre o estado (habilitado/deshabilitado).

Conectores

Esta sección permite configurar conectores para la transmisión de datos de Zabbix a sistemas externos mediante HTTP.



Haga clic en *Crear conector* para configurar un nuevo conector.

Puede filtrar los conectores por nombre o estado (habilitado/deshabilitado). Haga clic en el estado del conector en la lista para habilitarlo o deshabilitarlo. También puede habilitar o deshabilitar varios conectores a la vez seleccionándolos en la lista y haciendo clic en los botones *Habilitar/Deshabilitar* situados debajo de la lista.

Otros

Esta sección permite configurar otros parámetros diversos del frontend.



Parámetro	Descripción
URL del frontend	URL (de hasta 2048 caracteres) de la interfaz web de Zabbix. Este parámetro es utilizado por el servicio web de Zabbix para la comunicación con el frontend y debe especificarse para habilitar los informes programados.
Grupo para hosts descubiertos	Los hosts descubiertos por descubrimiento de red y autorregistro de agent se colocarán automáticamente en el grupo de hosts seleccionado aquí.
Modo de inventario de host predeterminado	Modo predeterminado para el inventario de hosts. Se aplicará siempre que se cree un nuevo host o prototipo de host mediante el server o el frontend, a menos que se sobrescriba durante el descubrimiento/autorregistro del host mediante la operación *Establecer modo de inventario de host*.

Parámetro	Descripción
Grupo de usuarios para mensaje de base de datos caída	Grupo de usuarios para enviar el mensaje de alarma de caída de la base de datos. Zabbix server depende de la disponibilidad de la base de datos backend. No puede funcionar sin una base de datos. Si la base de datos no está disponible, Zabbix puede notificar a los usuarios seleccionados. Las notificaciones se enviarán al grupo de usuarios establecido aquí utilizando las entradas de medios de usuario habilitadas. Las notificaciones se transmiten utilizando los siguientes tipos de medio (cuando están habilitados): correo electrónico; SMS; scripts de alerta personalizados. Incluso si una entrada de medio webhook está configurada y habilitada, no activará notificaciones. Zabbix server no se detendrá; esperará hasta que la base de datos vuelva a estar disponible para continuar el procesamiento. La notificación consta del siguiente contenido: `[MySQL|PostgreSQL] database <DB Name> [on <DB Host>:<DB Port>] is not available: <error message depending on the type of DBMS (database)>` <DB Host> no se añade al mensaje si se define como un valor vacío y <DB Port> no se añade si es el valor predeterminado (”0”). El gestor de alertas (un proceso especial de Zabbix server) intenta establecer una nueva conexión con la base de datos cada 10 segundos. Si la base de datos sigue sin estar disponible, el gestor de alertas repite el envío de alertas, pero no con una frecuencia mayor a una vez cada 15 minutos.
Registrar traps SNMP no coincidentes	Registrar el trap SNMP si no se han encontrado interfaces SNMP correspondientes.

Autorización

Parámetro	Descripción
Intentos de inicio de sesión	Número de intentos fallidos de inicio de sesión antes de que se bloquee la posibilidad de iniciar sesión.
Intervalo de bloqueo de inicio de sesión	Período de tiempo durante el cual se prohibirá el inicio de sesión cuando se supere el límite de *Intentos de inicio de sesión*. Rango permitido: 30 - 3600 segundos. Se admiten sufijos de tiempo, por ejemplo, 90s, 5m, 1h.

Almacenamiento de secretos

Parameter	Description
Vault provider	Seleccione el software de gestión de secretos para almacenar valores de user macro - *HashiCorp Vault* (predeterminado) o *CyberArk Vault*.
Resolve secret vault macros by	Resolver macros de secret vault mediante: **Zabbix server** - los secretos son recuperados por Zabbix server y reenviados a los proxies cuando es necesario (predeterminado); **Zabbix server and proxy** - los secretos son recuperados tanto por Zabbix server como por los proxies, lo que les permite resolver macros de forma independiente.

Véase también: Storage of secrets.

Seguridad

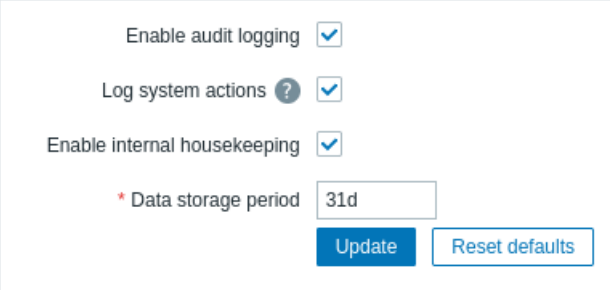
Parámetro	Descripción
Validar esquemas URI	Desmarque esta casilla para deshabilitar la validación de esquemas URI (habilitado por defecto). Si está marcada, puede especificar una lista separada por comas de esquemas URI permitidos (por defecto: http,https,ftp,file,mailto,tel,ssh). Se aplica a todos los campos en la interfaz donde se utilizan URIs (por ejemplo, URLs de elementos de mapas).

Parámetro	Descripción
Usar la cabecera HTTP X-Frame-Options	Desmarque esta casilla para deshabilitar la cabecera HTTP X-Frame-options (no recomendado). Si está marcada, puede especificar el valor de la cabecera HTTP X-Frame-options. Valores soportados: **SAMEORIGIN** (por defecto) o **'self'** (debe ir entre comillas simples): la página solo puede mostrarse en un marco en el mismo origen que la propia página; **DENY** o **'none'** (debe ir entre comillas simples): la página no puede mostrarse en un marco, independientemente del sitio que intente hacerlo; **una cadena de nombres de equipos separados por espacios**; agregar **'self'** (debe ir entre comillas simples) a la lista permite que la página se muestre en un marco en el mismo origen que la propia página. Tenga en cuenta que usar **'self'** o **'none'** sin comillas simples hará que se consideren como nombres de equipos.
Usar sandboxing en iframe	Desmarque esta casilla para deshabilitar la inclusión del contenido de la URL recuperada en un sandbox (no recomendado). Si está marcada, puede especificar las excepciones de sandboxing para el iframe; las restricciones no especificadas seguirán aplicándose. Si este campo está vacío, se aplican todas las restricciones del atributo sandbox. Para más información, consulte la descripción del atributo `sandbox`.

Registro de auditoría

Descripción general

Esta sección permite configurar los ajustes del registro de auditoría.



Están disponibles los siguientes parámetros:

Parámetro	Descripción
Habilitar registro de auditoría	Habilita (por defecto) o deshabilita el registro de auditoría.
Registrar acciones del sistema	Habilita (por defecto) o deshabilita el registro de auditoría de actividades de bajo nivel, descubrimiento de red y autorregistro realizadas por el servidor (usuario del sistema).
Habilitar limpieza interna	Habilita (por defecto) o deshabilita la limpieza interna de los registros de auditoría.
Período de almacenamiento de datos	Cantidad de días que los registros de auditoría deben conservarse antes de ser eliminados por el limpiador. Obligatorio si la limpieza está habilitada. Por defecto: 31 días.

Los parámetros de compresión del registro de auditoría están disponibles en la sección *Administración* > *Limpieza*, dentro del bloque *Compresión de histórico, tendencias y registro de auditoría*, que se vuelve visible cuando se utiliza TimescaleDB.

Mantenimiento de datos

Resumen

La sección *Administration* > *Housekeeping* permite configurar el housekeeping interno de Zabbix, que elimina automáticamente de la base de datos los datos obsoletos o eliminados por el usuario. Esto evita que la base de datos crezca indefinidamente, lo que consumiría espacio en disco innecesario y degradaría el rendimiento de la base de datos.

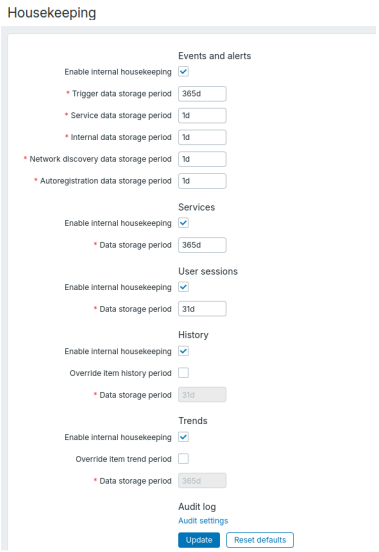
El housekeeping se puede habilitar y configurar para los siguientes tipos de datos:

- Eventos y alertas de triggers, servicios, datos internos, descubrimiento de red y autoregistro
- Servicios
- Sesiones de usuario
- Historial y tendencias para items

Note:
El housekeeping del registro de auditoría se configura por separado.

Configuración

Los siguientes parámetros están disponibles:



Parameter	Description
Enable internal housekeeping	Habilitar o deshabilitar el housekeeping interno (habilitado de forma predeterminada). Si está habilitado, el procedimiento de housekeeping elimina automáticamente de la base de datos los datos que superan el *Data storage period*.
Data storage period	Especifique durante cuánto tiempo conservar los datos antes de que el housekeeper los elimine. Obligatorio si el housekeeping interno está habilitado. Rango: 1 day (1 hour for history) - 25 years; o ”0”. Se admiten sufijos de tiempo (por ejemplo, 1d, 1w). Para *Events and alerts*, el período de almacenamiento de datos se establece por separado para triggers, services, datos internos, descubrimiento de red y autoregistro. El housekeeper elimina solo aquellos eventos que no están asociados con problemas. Por ejemplo, un evento de problema/recuperación anterior al *Data storage period* no se eliminará si está asociado con un problema sin resolver. Cuando el housekeeper elimina entidades obsoletas, primero elimina los problemas y luego los eventos. Para *History* y *Trends*, los períodos de almacenamiento de datos también determinan cuánto tiempo permanecen visibles los datos en *Monitoring > Latest data*, incluso si el housekeeping interno está deshabilitado. Tenga en cuenta que cuando se elimina un item o trigger, también se eliminan sus problemas y los eventos asociados.
Override item history period	Si está habilitado, el período de almacenamiento de history/trend especificado en la configuración del item se reemplaza por el ajuste *Data storage period* (excepto para los items con la opción *Do not store* habilitada). Esta opción puede utilizarse incluso cuando el housekeeping interno está deshabilitado y se usa un housekeeper externo.
Override item trend period	

Si usa **ClickHouse**, la sección *ClickHouse data storage period* estará disponible. Muestra el período de almacenamiento de datos

para cada tipo de valor de item. Para cambiar el período, consulte Configuring ClickHouse data storage period. Tenga en cuenta que el housekeeper no elimina datos de ClickHouse.

Si usa **Elasticsearch**, la sección *Elasticsearch data storage period* estará disponible. Muestra los tipos de valor de item guardados en Elasticsearch, con el período de almacenamiento de datos mostrado como *Unknown*. Para controlar durante cuánto tiempo se conservan los datos, consulte Index Lifecycle Management. Tenga en cuenta que el housekeeper no elimina datos de Elasticsearch.

Si usa **TimescaleDB**, la sección *History, trends and audit log compression* estará disponible.

Note:
Para TimescaleDB, habilite *Override item history period*, *Override item trend period* y *Enable internal housekeeping* para history y trends para aprovechar al máximo la partición automática. Si estas opciones están deshabilitadas, los datos almacenados en las tablas de history y trends seguirán particionados, pero el housekeeper no eliminará las particiones obsoletas y aparecerán advertencias de configuración. Cuando se habilita la eliminación de particiones obsoletas, Zabbix server y frontend dejarán de seguir los items eliminados, y el history de esos items se borrará cuando se elimine una partición obsoleta.

El housekeeper se ejecuta automáticamente cada hora de forma predeterminada. Puede cambiar la frecuencia de ejecución configurando el parámetro de configuración de Zabbix server `HousekeepingFrequency`.

El botón *Reset defaults* le permite revertir cualquier cambio realizado.

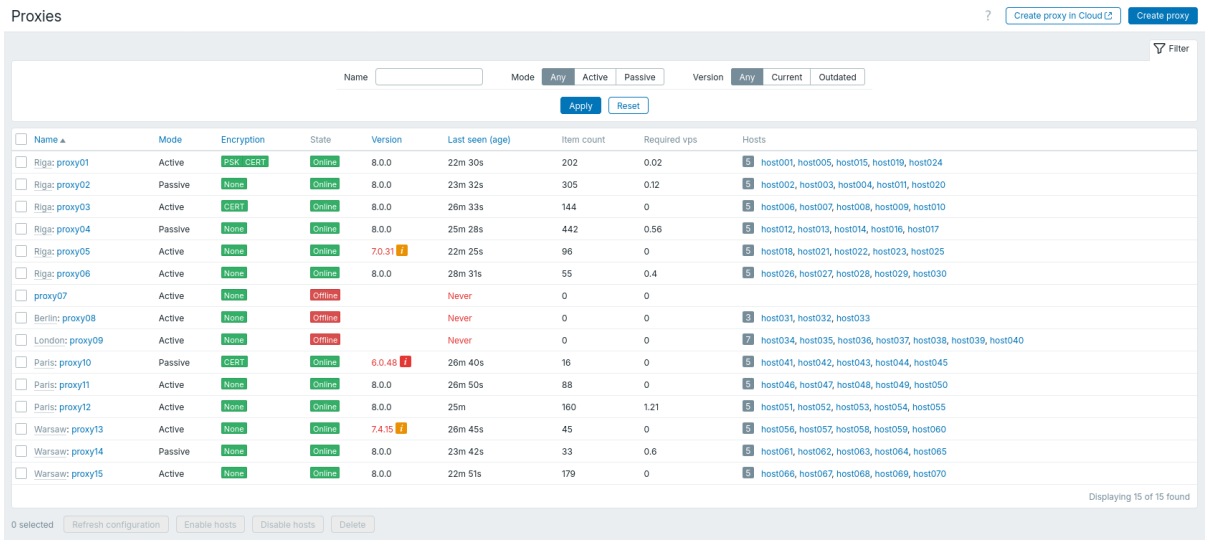
Proxies

Descripción general

En la sección *Administración* > *Proxies* del frontend de Zabbix se pueden configurar los proxies para la monitorización distribuida.

Proxies

Se muestra una lista de los proxies existentes con sus detalles.



Datos mostrados:

Columna	Descripción
Nombre	Nombre del proxy. Al hacer clic en el nombre del proxy, se abre el formulario de configuración del proxy. Si el proxy pertenece a un grupo de proxies, el nombre del grupo se muestra antes del nombre del proxy como un enlace gris. Al hacer clic en el nombre del grupo, se abre el formulario de configuración del grupo de proxies.
Modo	Modo del proxy: *Activo* o *Pasivo*.
Cifrado	Estado del cifrado para las conexiones del proxy: **Ninguno**: sin cifrado; **PSK**: uso de una clave precompartida; **Cert**: uso de un certificado.

Columna	Descripción
Estado	Estado del proxy: **Desconocido**: el proxy se creó mientras el Zabbix server estaba detenido o el server aún no ha actualizado el estado; **En línea**: el proxy se ha comunicado con el Zabbix server (el proxy pasivo respondió a una solicitud del server; el proxy activo envió una solicitud) durante el período de conmutación por error; **Fuera de línea**: el proxy no se ha comunicado con el Zabbix server durante el período de conmutación por error.
Versión	Versión del proxy (número de versión de tres dígitos). Si el proxy está desactualizado o no es compatible, el número de versión se resalta (en rojo) y se muestra un icono de estado informativo (amarillo o rojo). Pase el cursor sobre el icono para ver los detalles.
Última detección (antigüedad)	Momento en que el server detectó por última vez el proxy.
Cantidad de items	Número de items habilitados en los hosts habilitados asignados al proxy.
VPS requeridos	Rendimiento requerido del proxy (número de valores que deben recopilarse por segundo).
Hosts	Cantidad de hosts habilitados asignados al proxy y lista de hosts supervisados por el proxy. Al hacer clic en el nombre del host, se abre el formulario de configuración del host.

Para configurar un proxy nuevo, haga clic en el botón *Crear proxy* situado en la esquina superior derecha. Para implementar un proxy nuevo en Zabbix Cloud, haga clic en el botón *Crear proxy en Cloud* situado junto a él. Esto abre Zabbix Cloud en una nueva pestaña del navegador, donde puede crear un nodo proxy. Una vez creado el nodo proxy, configure el proxy mediante el botón *Crear proxy*. Para obtener más información, consulte Configurar un proxy de Zabbix Cloud.

Opciones de edición masiva

Los botones debajo de la lista ofrecen algunas opciones de edición masiva:

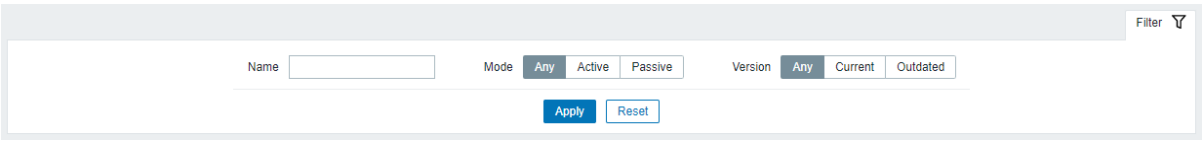
- *Actualizar configuración* - actualiza la configuración de los proxies;
- *Habilitar hosts* - cambia el estado de los hosts monitorizados por el proxy a *Monitored*;
- *Deshabilitar hosts* - cambia el estado de los hosts monitorizados por el proxy a *Not monitored*;
- *Eliminar* - elimina los proxies.

Para usar estas opciones, marque las casillas de verificación delante de los proxies correspondientes y luego haga clic en el botón requerido.

Uso del filtro

Puede utilizar el filtro para mostrar únicamente los proxy que le interesan. Para mejorar el rendimiento de la búsqueda, los datos se buscan con las macros sin resolver.

El enlace *Filtro* está disponible encima de la lista de proxy. Al hacer clic en él, aparece un filtro que permite filtrar los proxy por nombre, modo y versión. Tenga en cuenta que la opción de filtro *Obsoletos* muestra tanto los proxy obsoletos (con compatibilidad parcial) como los no compatibles.



Grupos de proxy

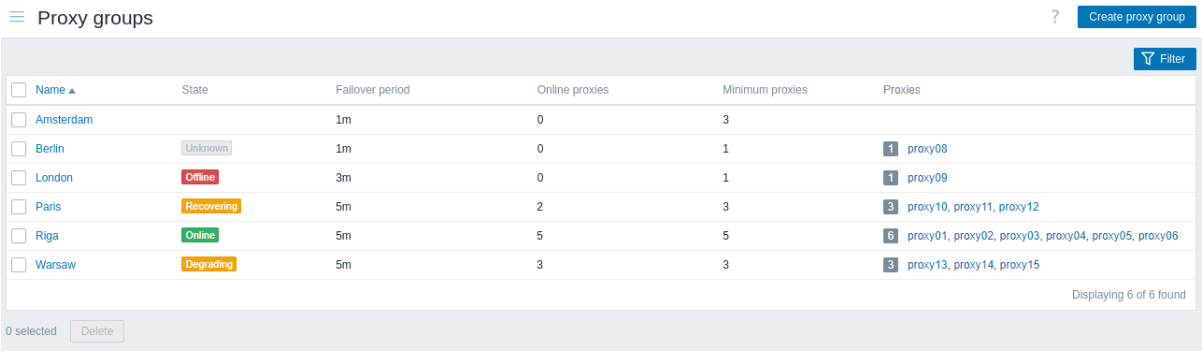
Resumen

En *Administration* > *Proxy groups* es posible configurar grupos de proxy.

Los grupos de proxy se utilizan en proxy load balancing con distribución automatizada de hosts entre proxies y alta disponibilidad entre proxies.

Grupos de proxy

Se muestra una lista de los grupos de proxy existentes con sus detalles.



Datos mostrados:

Columna	Descripción
Nombre	Nombre del grupo de proxy. Al hacer clic en el nombre del grupo de proxy, se abre el formulario de configuración del grupo de proxy.
Estado	Estado del grupo de proxy (se muestra si el grupo contiene al menos un proxy): **Desconocido**: el grupo se creó mientras el Zabbix server estaba detenido o el server aún no ha actualizado el estado; **En línea**: el número mínimo de proxies se ha comunicado con el Zabbix server durante el período de conmutación por error; **Degradándose**: el grupo pasará a estar fuera de línea en breve porque el número de proxies en línea ha caído por debajo del umbral *Proxies mínimos*; **Fuera de línea**: menos del número mínimo de proxies se ha comunicado con el Zabbix server durante el período de conmutación por error; **Recuperándose**: el grupo volverá a estar en línea en breve porque el número de proxies en línea ha alcanzado el umbral *Proxies mínimos*. El estado del grupo de proxy se puede supervisar con los items zabbix[proxy group,<name>,state] y zabbix[proxy group,discovery].
Período de conmutación por error	Período durante el cual un proxy del grupo de proxy debe comunicarse con el Zabbix server para considerarse en línea.
Proxies en línea	Número de proxies en línea (se muestra en rojo si es inferior al mínimo del grupo).
Proxies mínimos	Número mínimo de proxies en línea necesarios para mantener el grupo de proxy en línea.
Proxies	Número de proxies del grupo y una lista de los proxies del grupo, con enlaces al formulario de configuración del proxy. El número máximo de proxies mostrados está limitado por el valor de *Número máximo de elementos que se mostrarán dentro de una celda de tabla*.

Para configurar un nuevo grupo de proxy, haga clic en el botón *Crear grupos de proxy* situado en la esquina superior derecha.

Opciones de edición masiva

Los botones debajo de la lista ofrecen algunas opciones de edición masiva:

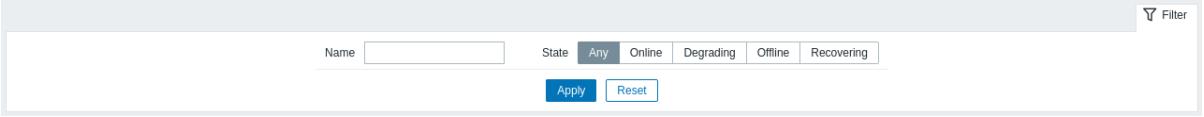
- *Eliminar* - elimina los grupos de proxy.

Para usar estas opciones, marque las casillas de verificación delante de los respectivos grupos de proxy y, a continuación, haga clic en el botón requerido.

Uso del filtro

Puede usar el filtro para mostrar solo los grupos de proxy que le interesen. Para un mejor rendimiento de búsqueda, los datos se buscan con las macros sin resolver.

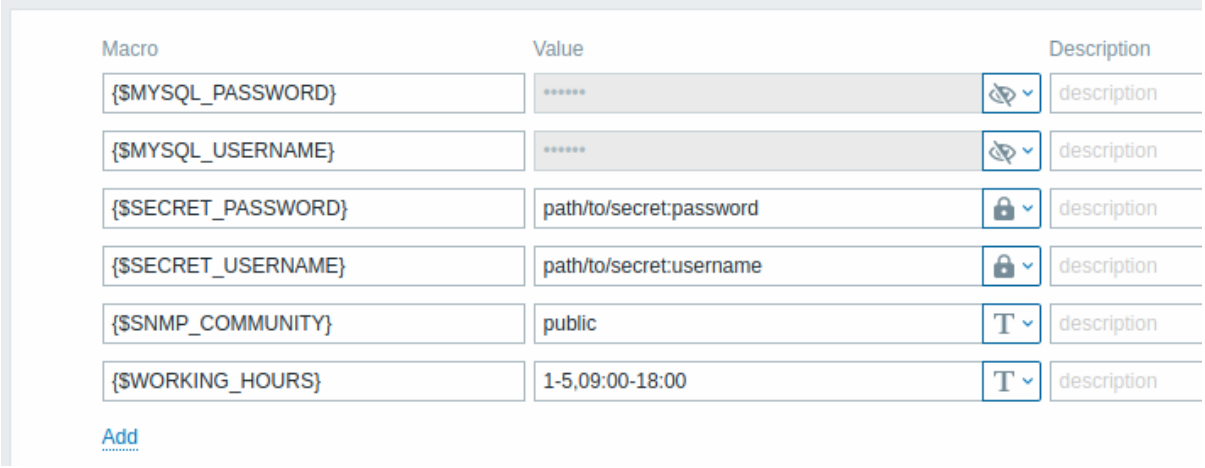
El enlace *Filter* está disponible encima de la lista de grupos de proxy. Si hace clic en él, se mostrará un filtro en el que puede filtrar los grupos de proxy por nombre y estado.



Macros

Descripción general

Esta sección permite definir macros de usuario para todo el sistema como pares de nombre y valor. Tenga en cuenta que los valores de las macros pueden conservarse como texto sin formato, texto secreto o secreto de Vault. También se admite añadir una descripción.



Cola

Descripción general

En la sección *Administración* > *Cola* se muestran los items que están esperando a ser actualizados.

Idealmente, al abrir esta sección todo debería estar en "verde", lo que significa que no hay items en la cola. Si todos los items se actualizan sin demora, no habrá ninguno esperando. Sin embargo, debido a un rendimiento insuficiente del server, algunos items pueden retrasarse y la información se muestra en esta sección. Para obtener más información, consulte la sección Cola.

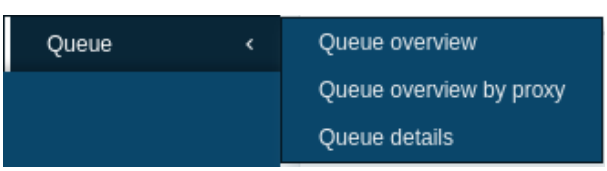
Note:
El proxy de Zabbix no tiene conocimiento de los períodos de mantenimiento; consulte Cálculo de las colas durante el mantenimiento para obtener más información.

Note:
La cola solo está disponible si el server de Zabbix está en ejecución. Los items no se contabilizan en la cola si la interfaz del item deja de estar disponible debido a problemas de conexión o a que el agent no funciona correctamente.

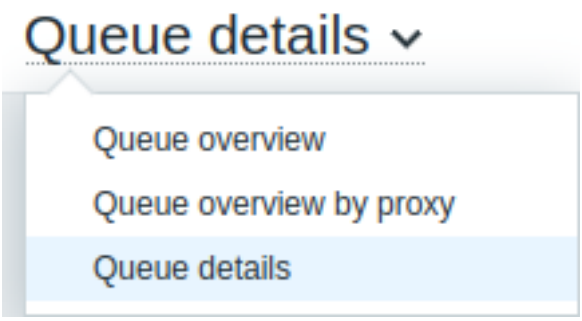
La sección *Administración* > *Cola* contiene las siguientes páginas:

- Descripción general de la cola: muestra la cola por tipo de item;
- Descripción general de la cola por proxy: muestra la cola por proxy;
- Detalles de la cola: muestra una lista de items retrasados.

La lista de páginas disponibles aparece al hacer clic en *Cola* en la sección del menú *Administración*. También es posible cambiar entre las páginas mediante el menú desplegable del título situado en la esquina superior izquierda.



Menú de tercer nivel.



Menú desplegable del título.

Resumen por tipo de métrica

En esta pantalla es fácil localizar si el problema está relacionado con uno o varios tipos de métricas.

Queue overview

Items	5 seconds	10 seconds	30 seconds	1 minute	5 minutes	More than 10 minutes
Zabbix agent	1	11	1	0	0	0
Zabbix agent (active)	0	0	0	0	0	0
Simple check	0	0	0	0	0	0
SNMPv1 agent	0	0	0	0	0	0
SNMPv2 agent	0	0	0	0	0	0
SNMPv3 agent	0	0	0	0	0	0
Zabbix internal	0	0	0	0	0	0
Zabbix aggregate	0	0	0	0	0	0
External check	0	0	0	0	0	0
Database monitor	0	0	0	0	0	0
HTTP agent	0	0	0	0	0	0

Cada línea contiene un tipo de métrica. Cada columna muestra el número de métricas en espera: esperando durante 5-10 segundos/10-30 segundos/30-60 segundos/1-5 minutos/5-10 minutos o más de 10 minutos respectivamente.

Resumen por proxy

En esta pantalla es fácil localizar si el problema está relacionado con uno de los proxies o con el servidor.

Queue overview by proxy

Proxy	5 seconds	10 seconds	30 seconds	1 minute	5 minutes	More than 10 minutes
Remote proxy	0	8	11	0	0	0
Server	0	0	0	0	0	0

Total: 2

Cada línea contiene un proxy, con el servidor al final de la lista. Cada columna muestra el número de métricas en espera: esperando durante 5-10 segundos/10-30 segundos/30-60 segundos/1-5 minutos/5-10 minutos o más de 10 minutos respectivamente.

Lista de métricas en espera

En esta pantalla, se muestra cada métrica en espera.

Queue details

Scheduled check	Delayed by	Host	Name	Proxy
2019-09-02 11:46:40	58s	My host	CPU idle time	Remote proxy
2019-09-02 11:46:41	57s	My host	CPU interrupt time	Remote proxy
2019-09-02 11:46:42	56s	My host	CPU iowait time	Remote proxy
2019-09-02 11:46:43	55s	My host	CPU nice time	Remote proxy
2019-09-02 11:46:44	54s	My host	CPU softirq time	Remote proxy
2019-09-02 11:46:45	53s	My host	CPU steal time	Remote proxy
2019-09-02 11:46:46	52s	My host	CPU system time	Remote proxy

Datos mostrados:

Columna	Descripción
Comprobación programada	Se muestra la hora en que debía realizarse la comprobación.
Retrasado por	Se muestra la duración del retraso.
Equipo	Se muestra el equipo de la métrica.
Nombre	Se muestra el nombre de la métrica en espera.
Proxy	Se muestra el nombre del proxy, si el equipo es monitorizado por proxy.

Posibles mensajes de error

Puede encontrarse con una situación en la que no se muestran datos y aparece el siguiente mensaje de error:

Details — Cannot display item queue.

Permission denied.

En este caso, el mensaje de error es el siguiente:

```
Cannot display item queue. Permission denied
```

Esto ocurre cuando los parámetros de configuración de PHP del archivo de configuración del frontend - `$ZBX_SERVER` o `$ZBX_SERVER` y `$ZBX_SERVER_PORT` - apuntan a un server de Zabbix existente que utiliza una base de datos diferente.

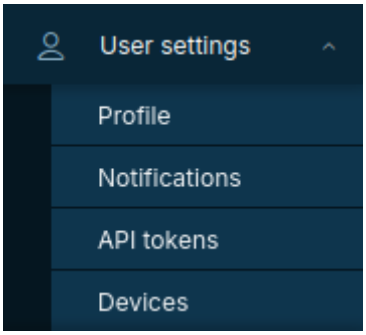
Configuración de usuario

Descripción general

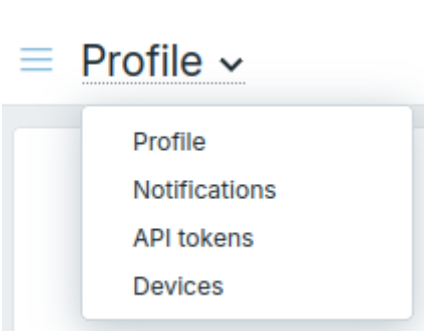
Según tu rol de usuario, la sección *Configuración del usuario* puede contener las siguientes páginas:

- *Perfil* o *Perfil de usuario*: para personalizar determinadas funciones del frontend de Zabbix;
- *Notificaciones*: para personalizar las notificaciones del usuario actual;
- *Tokens de API*: para gestionar los tokens de API asignados al usuario actual;
- *Dispositivos*: para gestionar los dispositivos móviles pertenecientes al usuario actual.

La lista de páginas disponibles aparece al hacer clic en el icono de usuario situado cerca de la parte inferior del menú de Zabbix (no disponible para el usuario *guest*). También es posible cambiar entre las páginas mediante el menú desplegable del título situado en la esquina superior izquierda.



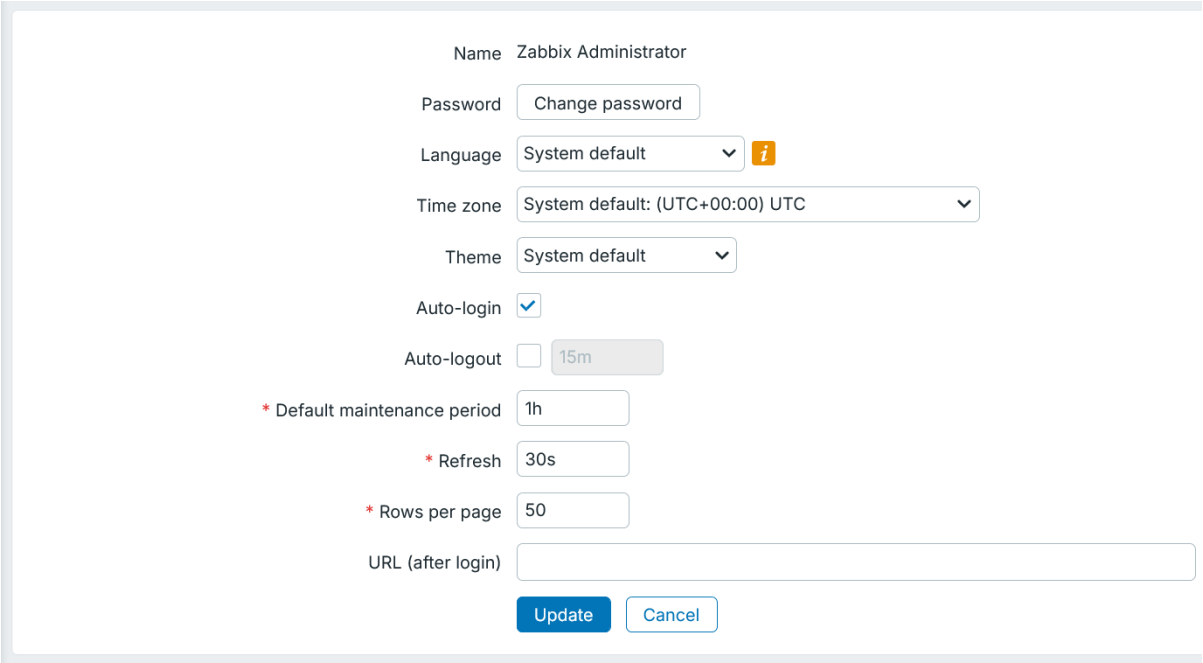
Menú de tercer nivel.



Menú desplegable del título.

Perfil de usuario

La sección *Perfil de usuario* proporciona opciones para establecer un idioma personalizado de la interfaz, un tema de color, el número de filas mostradas en las listas, etc. Los cambios realizados aquí se aplicarán únicamente al usuario actual.



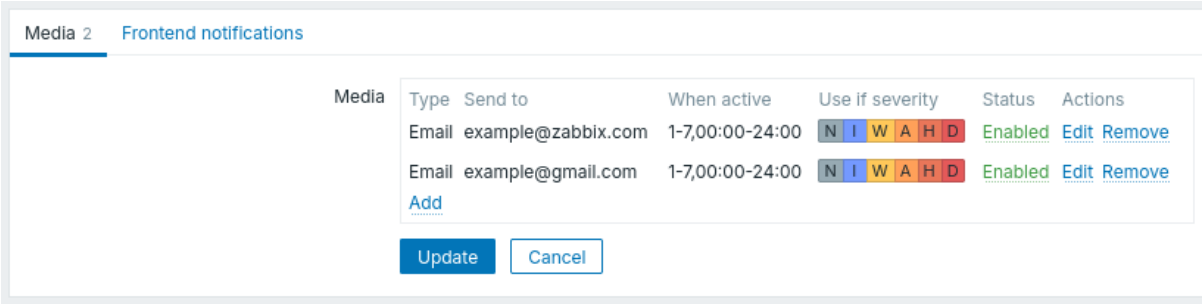
Parámetro	Descripción
Nombre	Se muestran el nombre y los apellidos del usuario. Si no se han especificado el nombre y los apellidos, se muestra el nombre de usuario.

Parámetro	Descripción
Contraseña	Haga clic en el botón *Cambiar contraseña* para abrir tres campos: *Contraseña anterior*, *Nueva contraseña*, *Nueva contraseña (una vez más)*. Después de cambiar correctamente la contraseña, se cerrará la sesión del usuario en todas las sesiones activas. Tenga en cuenta que la contraseña solo se puede cambiar para usuarios que utilicen la autenticación interna de Zabbix.
Idioma	Seleccione el idioma de la interfaz que prefiera o seleccione **Predeterminado del sistema** para utilizar la configuración predeterminada del sistema. Al seleccionar *Inglés (en_US)* también se habilitará el formato de hora y fecha de EE. UU. en el frontend. Para obtener más información, consulte Instalación de idiomas adicionales del frontend.
Zona horaria	Seleccione la zona horaria para sustituir la zona horaria global a nivel de usuario, o seleccione **Predeterminado del sistema** para utilizar la configuración global de zona horaria.
Tema	Seleccione un tema de color específico para su perfil: **Predeterminado del sistema**: utiliza la configuración predeterminada del sistema; **Azul**: tema azul estándar; **Azul (clásico)**: tema azul estándar con fuentes heredadas; **Oscuro**: tema oscuro alternativo; **Azul oscuro**: tema azul oscuro; **Oscuro (clásico)**: tema oscuro alternativo con fuentes heredadas; **Claro de alto contraste**: tema claro con alto contraste; **Oscuro de alto contraste**: tema oscuro con alto contraste.
Inicio de sesión automático	Marque esta casilla para que Zabbix le recuerde y le conecte automáticamente durante 30 días. Al iniciar sesión con *Recordarme durante 30 días* habilitado: - El temporizador de cierre de sesión automático se restablece (la sesión persiste hasta que finaliza el período de 30 días). - El inicio de sesión automático se habilita en las visitas posteriores dentro de los 30 días. Al iniciar sesión sin *Recordarme durante 30 días*: - El parámetro de inicio de sesión automático se borra. - El cierre de sesión automático sigue el tiempo de espera configurado, como de costumbre. Para ello se utilizan cookies del navegador.
Cierre de sesión automático	Con esta casilla marcada, se cerrará la sesión automáticamente después del número de segundos establecido (mínimo 90 segundos, máximo 1 día). Se admiten sufijos de tiempo, por ejemplo: 90s, 5m, 2h, 1d. Tenga en cuenta que esta opción no funcionará en los siguientes casos: * Cuando las páginas del menú *Monitorización* realizan actualizaciones de información en segundo plano. Si se dejan abiertas páginas que actualizan datos en un intervalo específico (paneles, gráficos, datos más recientes, etc.), la duración de la sesión se prolonga y, por tanto, se deshabilita la función de cierre de sesión automático. * Si se inicia sesión con la opción *Recordarme durante 30 días* marcada; en este caso, se anula el temporizador de cierre de sesión automático y la sesión permanece activa durante todo el período. * Cuando la autenticación se realiza mediante un proveedor de identidad SSO (IdP), el cierre de sesión automático solo finaliza la sesión de Zabbix. La sesión SSO con el IdP permanece activa, por lo que el usuario puede volver a iniciar sesión sin introducir una contraseña. *Cierre de sesión automático* también puede aceptar ”0”, lo que significa que la función de cierre de sesión automático se deshabilita después de actualizar la configuración del perfil.
Período de mantenimiento predeterminado	Establezca el período de mantenimiento predeterminado para el mantenimiento ad hoc.
Actualización	Establezca con qué frecuencia se actualizará la información de las páginas del menú *Monitorización* (mínimo 0 segundos, máximo 1 hora). Se admiten sufijos de tiempo, por ejemplo: 30s, 90s, 1m, 1h.
Filas por página	Establezca cuántas filas se mostrarán por página en las listas. Un número menor de filas (y de registros que mostrar) da como resultado tiempos de carga más rápidos.
URL (después de iniciar sesión)	Establezca una URL específica que se mostrará después de iniciar sesión. En lugar de los *Paneles* predeterminados, puede ser, por ejemplo, la URL de *Monitorización > Triggers*. Este campo está deshabilitado si un usuario *Superadministrador* ha restringido la URL posterior al inicio de sesión a una página específica para su rol de usuario. Si el campo está deshabilitado o el usuario no especifica ningún valor, el valor predeterminado se muestra debajo del campo, por ejemplo, `zabbix.php?action=dashboard.view`

Notificaciones

La sección *Notificaciones* ofrece opciones para personalizar las notificaciones del usuario actual.

La pestaña *Media* le permite especificar los detalles de media para el usuario, como los tipos de media y las direcciones que se utilizarán, así como cuándo usarlos para entregar notificaciones.



Si el tipo de media ha sido deshabilitado:

- se muestra un icono informativo amarillo después del nombre;
- se muestra "Disabled" en la columna *Status*.

Note:
Los permisos de usuario para cambiar sus propios detalles de media pueden concederse o revocarse según su rol de usuario (consulte la opción *Create and edit own media*). Los permisos del usuario Super admin para cambiar los detalles de media de otros usuarios también pueden concederse o revocarse según su rol de usuario (consulte la opción *Create and edit user media*)

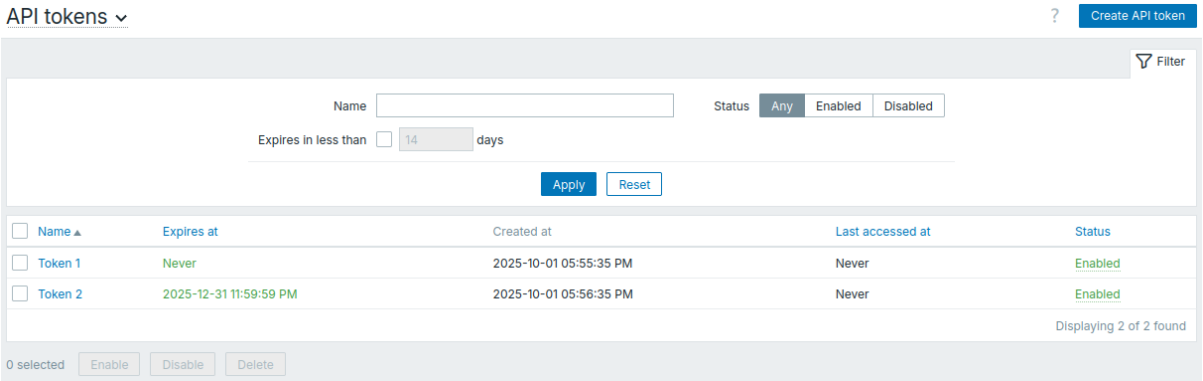
Tenga en cuenta que, para los usuarios aprovisionados:

- el media de usuario aprovisionado no puede eliminarse;
- el media de usuario aprovisionado puede deshabilitarse/habilitarse;
- los campos de media de usuario aprovisionado como *When active*, *Use if severity* y *Enabled* pueden editarse manualmente;
- se puede añadir media de usuario adicional (por ejemplo, direcciones de correo electrónico adicionales) manualmente para los usuarios aprovisionados;
- el media de usuario añadido manualmente puede eliminarse.

La pestaña *Frontend notifications* le permite configurar las notificaciones globales.

Tokens de API

La sección *Tokens de API* le permite ver los tokens asignados al usuario, editar los detalles del token y crear nuevos tokens. Esta sección está disponible para un usuario solo si la acción *Manage API tokens* está permitida en la configuración de user role.



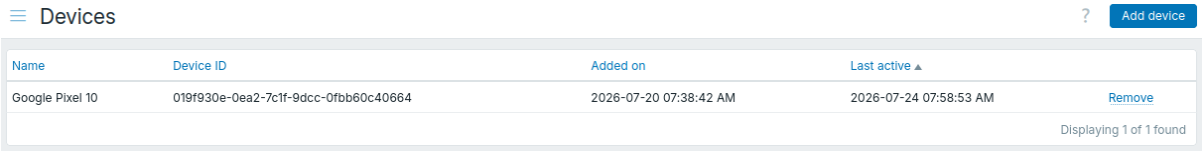
Puede filtrar los tokens de API por nombre, fecha de expiración o estado (*Enabled/Disabled*). Haga clic en el estado del token en la lista para habilitar o deshabilitar rápidamente un token. También puede habilitar o deshabilitar varios tokens a la vez seleccionándolos en la lista y haciendo clic en los botones *Enable/Disable* debajo de la lista.

Attention:
Los usuarios no pueden ver el valor del *Auth token* de los tokens asignados a ellos en Zabbix. El valor del *Auth token* se muestra solo una vez, inmediatamente después de crear un token. Si el token se ha perdido, debe regenerarse.

Dispositivos

La sección *Dispositivos* permite agregar o eliminar sus dispositivos móviles para utilizarlos con la aplicación móvil de Zabbix.

Esta sección se muestra si `EnableMobileDevices` está habilitado.



Para agregar un dispositivo:

1. Haga clic en *Agregar dispositivo*.
2. Escanee el código QR en su aplicación móvil de Zabbix.

Note:
Al agregar un dispositivo, Zabbix también agrega el tipo de medio de notificación push a sus medios de usuario. Al eliminar el dispositivo, esos medios de usuario también se eliminan. Si tiene configurados varios tipos de medios de notificación push, Zabbix agrega el que está habilitado y aparece primero alfabéticamente.

Los dispositivos agregados aparecen en una lista. La lista muestra su nombre, ID y las fechas y horas en que se agregaron y estuvieron activos por última vez.

También puede eliminar dispositivos. Al eliminar un dispositivo, sus credenciales de acceso se revocan y eliminan de Zabbix, después de lo cual ya no puede acceder a Zabbix.

Los superadministradores también pueden administrar los dispositivos de otros usuarios en *Usuarios* > *Dispositivos*.

Notificaciones globales

Descripción general

Las notificaciones globales proporcionan una forma de mostrar problemas en tiempo real directamente en su pantalla actual dentro de la interfaz de Zabbix.

Sin notificaciones globales, al trabajar fuera de las secciones *Problemas* o *Panel de control*, no recibiría ninguna información sobre los problemas actuales. Las notificaciones globales garantizan que esta información se muestre, independientemente de su ubicación actual dentro de la interfaz de Zabbix.

Las notificaciones globales incluyen tanto mostrar un mensaje como reproducir un sonido.

Attention:
La reproducción automática de sonidos puede estar deshabilitada (por defecto) en las versiones recientes de los navegadores. En tales casos, debe habilitar esta configuración manualmente.

Configuración

Las notificaciones globales se pueden habilitar por usuario en la pestaña **Notificaciones en la interfaz** de la sección Notificaciones.

profile_c.png

Parámetro	Descripción
Notificaciones en la interfaz	Marque la casilla para habilitar las notificaciones globales.
Tiempo de visualización del mensaje	Establezca la duración durante la cual se mostrará el mensaje. Por defecto, los mensajes permanecen en pantalla durante 60 segundos. Se admiten sufijos de tiempo, por ejemplo: 30s, 5m, 2h, 1d.
Reproducir sonido	Establezca la duración durante la cual se reproducirá el sonido. **Una vez**: el sonido se reproduce una vez y completamente; **10 segundos**: el sonido se repite durante 10 segundos; **Tiempo de visualización del mensaje**: el sonido se repite mientras el mensaje esté visible.
Gravedad del disparador	Establezca las gravedades de disparador para las que se activarán las notificaciones globales y los sonidos. También puede seleccionar sonidos apropiados para varias gravedades. Si no se marca ninguna gravedad, no se mostrarán mensajes. Además, los mensajes de recuperación solo se mostrarán para las gravedades marcadas. Por ejemplo, si se marcan *Recuperación* y *Desastre*, se mostrarán notificaciones globales para problemas y recuperaciones de disparadores de gravedad *Desastre*.

Parámetro	Descripción
Mostrar problemas suprimidos	Marque la casilla para mostrar notificaciones de problemas que de otro modo serían suprimidos (no mostrados) debido al mantenimiento del equipo.

Mensajes globales mostrados

A medida que llegan los mensajes, se muestran en una sección flotante en el lado derecho. Puede reposicionar libremente esta sección arrastrando el encabezado de la sección.

global_messages.png

Para esta sección, hay varios controles disponibles:

- El botón **Posponer** silencia el sonido de alarma actualmente activo;
- El botón **Silenciar/Activar sonido** alterna entre reproducir o no reproducir los sonidos de alarma.

Sonido en navegadores

Descripción general

El sonido se utiliza en las notificaciones globales.

Para que los sonidos se reproduzcan en el frontend de Zabbix, las *Notificaciones del frontend* deben estar habilitadas en la pestaña *Notificaciones del frontend* del perfil de usuario, con todas las severidades de disparadores marcadas. Además, los sonidos deben estar habilitados en la ventana emergente de notificaciones globales.

Si, por alguna razón, no se puede reproducir audio en el dispositivo, el botón en la ventana emergente de notificaciones globales permanecerá permanentemente en estado ”silenciado”, acompañado del mensaje ”No se puede soportar audio de notificación para este dispositivo” al pasar el cursor sobre el botón .

Los sonidos, incluidos los clips de audio predeterminados, solo son compatibles en formato MP3.

Los sonidos del frontend de Zabbix se han probado con éxito en las versiones recientes de los navegadores Firefox y Opera en Linux, y en los navegadores Chrome, Firefox, Microsoft Edge y Opera en Windows.

Attention:
La reproducción automática de sonidos puede estar deshabilitada (por defecto) en las versiones recientes de los navegadores. En tales casos, debe habilitar esta configuración manualmente.

Búsqueda global

Es posible buscar en el frontend de Zabbix hosts, grupos de hosts, templates y grupos de templates.

El cuadro de entrada de búsqueda se encuentra debajo del logotipo de Zabbix en el menú. La búsqueda se puede iniciar pulsando *Enter* o haciendo clic en el icono de búsqueda .



Si hay un host que contiene la cadena introducida en cualquier parte del nombre, aparecerá una lista desplegable con todos esos hosts (con la parte coincidente resaltada en naranja). La lista desplegable también mostrará un host si el nombre visible de ese host coincide con el nombre técnico introducido como cadena de búsqueda; el host coincidente se mostrará, pero sin ningún resaltado.

Atributos buscables

Los equipos pueden buscarse por las siguientes propiedades:

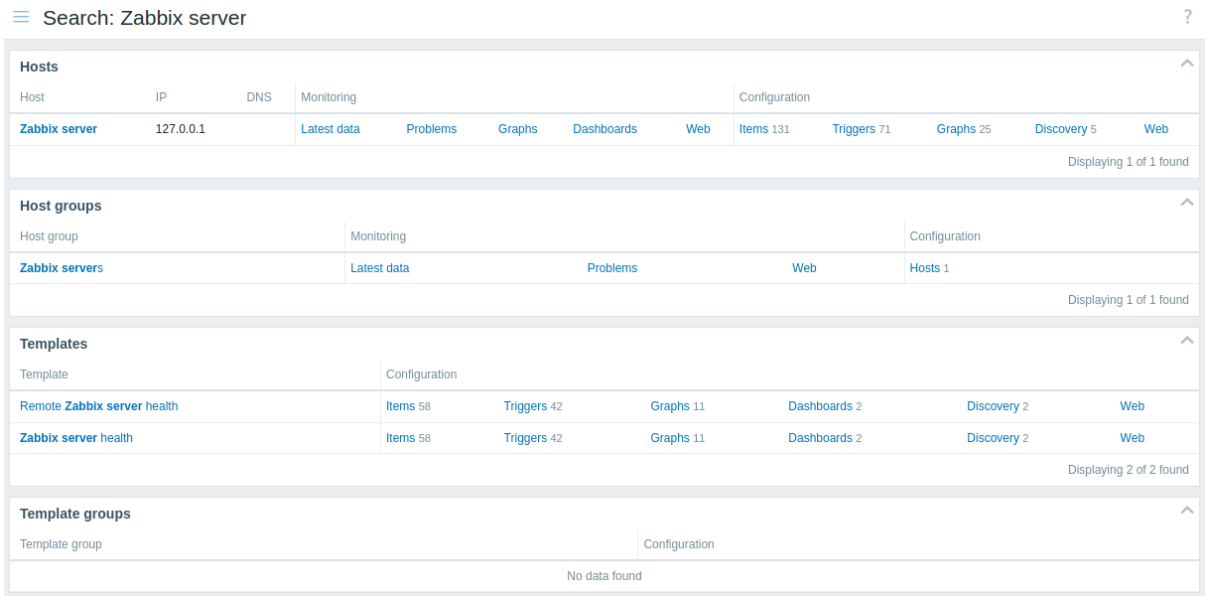
- Nombre del equipo
- Nombre visible
- Dirección IP
- Nombre DNS

Las plantillas pueden buscarse por nombre o nombre visible. Si busca por un nombre que es diferente al nombre visible (de una plantilla/equipo), en los resultados de búsqueda se muestra debajo del nombre visible entre paréntesis.

Los grupos de equipos y plantillas pueden buscarse por nombre. Especificar un grupo padre selecciona implícitamente todos los grupos anidados.

Resultados de búsqueda

Los resultados de búsqueda constan de cuatro bloques independientes para hosts, grupos de hosts, templates y grupos de templates.



Es posible contraer/expandir cada bloque individual. El número de entradas se muestra en la parte inferior de cada bloque; por ejemplo, *Mostrando 13 de 13 encontradas*. Si no hay entradas, el número de entradas no se muestra. El total de entradas mostradas dentro de un bloque está limitado a 100.

Cada entrada proporciona enlaces a datos de monitorización y configuración. Consulte la lista completa de enlaces.

Para todos los datos de configuración (como items, triggers, gráficos), la cantidad de entidades encontradas se muestra mediante un número junto al nombre de la entidad, en gris. **Tenga en cuenta** que si hay cero entidades, no se muestra ningún número.

Los hosts habilitados se muestran en azul y los hosts deshabilitados en rojo.

Enlaces disponibles

Para cada entrada están disponibles los siguientes enlaces:

- Equipos
  - Monitorización
    - Últimos datos
    - Problemas
    - Gráficas
    - Paneles de equipo
    - Escenarios web
  - Configuración
    - Métricas
    - Disparadores
    - Gráficas
    - Reglas de descubrimiento
    - Escenarios web
- Grupos de equipos
  - Monitorización
    - Últimos datos
    - Problemas
    - Escenarios web
  - Configuración
    - Equipos
- Plantillas
  - Configuración
    - Métricas
    - Disparadores
    - Gráficas
    - Paneles de plantilla
    - Reglas de descubrimiento
    - Escenarios web
- Grupos de plantillas
  - Configuración
    - Plantillas

Modo de mantenimiento del frontend

Descripción general

Es posible deshabilitar temporalmente la interfaz web de Zabbix para restringir el acceso. Esto es útil para proteger la base de datos de Zabbix de cambios iniciados por los usuarios, preservando su integridad.

Mientras la interfaz web de Zabbix está en modo de mantenimiento, puede detener la base de datos de forma segura y realizar tareas de mantenimiento.

Los usuarios desde direcciones IP definidas podrán interactuar con la interfaz web normalmente durante el modo de mantenimiento.

Configuración

Para habilitar el modo de mantenimiento, abra el archivo `maintenance.inc.php` (ubicado en `/conf`, dentro del directorio de documentos HTML de Zabbix en el server web) y quite las marcas de comentario de las siguientes líneas:

```
// Modo de mantenimiento.
define('ZBX_DENY_GUI_ACCESS', 1);

// Matriz de direcciones IP autorizadas para conectarse al frontend (opcional).
$ZBX_GUI_ACCESS_IP_RANGE = array('127.0.0.1');

// Mensaje mostrado en la pantalla de advertencia (opcional).
$ZBX_GUI_ACCESS_MESSAGE = 'Estamos actualizando la base de datos MySQL hasta las 15:00. Manténgase atento.
```

Note:
En la mayoría de los casos, el archivo `maintenance.inc.php` se encuentra en `/conf`, dentro del directorio de documentos HTML de Zabbix en el web server. Sin embargo, algunos sistemas operativos y web servers pueden utilizar una ubicación diferente.
Por ejemplo, la ubicación para:

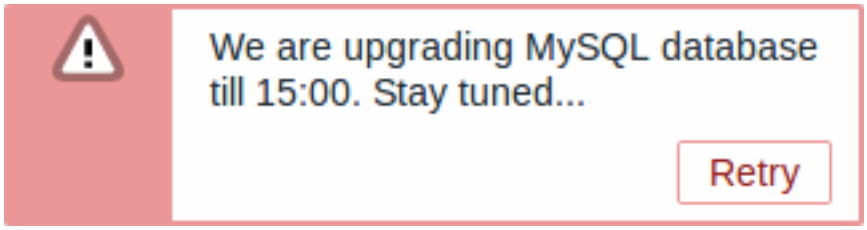
- SUSE y RedHat es `/etc/zabbix/web/maintenance.inc.php`.
- Los sistemas basados en Debian es `/usr/share/zabbix/conf/`.

Consulte también Copiar archivos PHP.

Parameter	Details
ZBX_DENY_GUI_ACCESS	Si se define con cualquier valor, se habilitará el modo de mantenimiento. Para deshabilitar el modo de mantenimiento, comente o elimine la definición.
ZBX_GUI_ACCESS_IP_RANGE	Array de direcciones IP autorizadas a conectarse al frontend (opcional). Por ejemplo: `array('192.168.1.1', '192.168.1.2')`
ZBX_GUI_ACCESS_MESSAGE	Mensaje para informar a los usuarios sobre el mantenimiento (opcional). Si no se define, se utilizará el mensaje predeterminado *'Zabbix is under maintenance'*.

Visualización

Los usuarios verán la siguiente pantalla al intentar acceder a la interfaz web de Zabbix mientras está en modo de mantenimiento. La pantalla se actualiza cada 30 segundos para volver al estado normal sin intervención del usuario cuando finaliza el mantenimiento.



Las direcciones IP definidas en *ZBX_GUI_ACCESS_IP_RANGE* podrán acceder a la interfaz web como de costumbre.

Parámetros de la página

Descripción general

La mayoría de las páginas de la interfaz web de Zabbix admiten varios parámetros HTTP GET que controlan lo que se mostrará. Se pueden pasar especificando pares de parámetro=valor después de la URL, separados de la URL por un signo de interrogación (?) y entre sí por signos de ampersand (&).

Monitoring > Problems

Se admiten los siguientes parámetros:

Parameter	Description	Example
show	Opción de filtro *Show*. Valores posibles: 1 - problemas recientes; 2 - todos; 3 - en estado de problema.	show=1
name	Opción de filtro *Problem*: cadena libre.	name=Zabbix agent

Parameter	Description	Example
severities	Opción de filtro *Severity*: array de severidades seleccionadas en el formato `severities[*]=*` (reemplace `*` por el nivel de severidad). Valores posibles: 0 - no clasificado; 1 - información; 2 - advertencia; 3 - promedio; 4 - alto; 5 - desastre.	severities[3]=3
inventory	Opción de filtro *Host inventory*: array de campos de inventario `[field],[value]`	inventory[0][field]=type&invento
evaltype	Opción de filtro *Tags*: método de evaluación de la etiqueta evaluation method. Valores posibles: 0 - and/or; 2 - or.	evaltype=0
tags	Opción de filtro *Tags*: array de etiquetas definidas `[tag], [operator], [value]` Valores posibles de operator: 0 - contiene; 1 - igual a; 2 - no contiene; 3 - no es igual a; 4 - existe; 5 - no existe.	tags[0][tag]=target&tags[0][ope
show_tags	Opción de filtro *Show tags*. Valores posibles: 0 - ninguno; 1 - uno; 2 - dos; 3 - tres.	show_tags=3
tag_name_format	Opción de filtro *Tag name*. Valores posibles: 0 - nombre completo; 1 - nombre abreviado; 2 - ninguno.	tag_name_format=1
tag_priority	Opción de filtro *Tag display priority*: cadena separada por comas de prioridad de visualización de etiquetas	tag_priority=customer, target
show_suppressed	Opción de filtro *Show suppressed problems*. Valores posibles: 1 - mostrar; 2 - no mostrar.	show_suppressed=1
acknowledgement_status	Opción de filtro *Acknowledgement status*. Valores posibles: 0 - todos; 1 - no reconocidos; 2 - reconocidos.	acknowledgement_status=0
acknowledged_by_me	Opción de filtro *By me*; solo compatible con `acknowledgement_status=2`. Valores posibles: 0 - deshabilitado; 1 - habilitado.	acknowledged_by_me=1

Parameter	Description	Example
compact_view	Opción de filtro *Compact view*. Valores posibles: 0 - deshabilitado; 1 - habilitado.	compact_view=1
highlight_row	Opción de filtro *Highlight whole row*. Valores posibles: 0 - deshabilitado; 1 - habilitado.	highlight_row=1
filter_name	Opción de propiedades del filtro *Name*: cadena libre	filter_name=Databases
filter_show_counter	Opción de propiedades del filtro *Show number of records*. Valores posibles: 0 - deshabilitado; 1 - habilitado.	filter_show_counter=1
filter_custom_time	Opción de propiedades del filtro *Set custom time period*. Valores posibles: 0 - deshabilitado; 1 - habilitado.	filter_custom_time=1
sort	Columna por la que ordenar. Valores posibles: clock - ordenar por la columna *Time*; host - ordenar por la columna *Host*; severity - ordenar por la columna *Severity*; name - ordenar por la columna *Problem*.	sort=clock
sortorder	Orden de clasificación de los resultados. Valores posibles: DESC - descendente; ASC - ascendente.	sortorder=DESC
age_state	Opción de filtro *Age less than*; solo compatible con `show`=3. Valores posibles: 0 - deshabilitar el parámetro `age`; 1 - habilitar el parámetro `age`.	age_state=1
age	Opción de filtro *Age less than*: entero, número de días; solo compatible con `age_state`=1 y `show`=3.	age=7
groupids	Opción de filtro *Host groups*: array de IDs de grupos de host.	groupids[]=4
hostids	Opción de filtro *Hosts*: array de IDs de host	hostids[]=10084
triggerids	Opción de filtro *Triggers*: array de IDs de trigger	triggerids[]=22382
show_timeline	Opción de filtro *Show timeline*; no compatible con `compact_view`=1. Valores posibles: 0 - no mostrar; 1 - mostrar.	show_timeline=1
details	Opción de filtro *Show details*. Valores posibles: 0 - no mostrar; 1 - mostrar.	details=1
from	Inicio del intervalo de fechas, puede ser relativo (por ejemplo, now-1m); solo compatible con `filter_custom_time`=1	from=now-2h
to	Fin del intervalo de fechas, puede ser relativo (por ejemplo, now-1m); solo compatible con `filter_custom_time`=1	to=now

Véase también: opciones de filtro de la página Problems.

Modo quiosco

El modo quiosco en las páginas compatibles del frontend se puede activar mediante parámetros de URL. Por ejemplo, en los paneles:

- `/zabbix.php?action=dashboard.view&kiosk=1` - activar el modo quiosco
- `/zabbix.php?action=dashboard.view&kiosk=0` - activar el modo normal

Presentación de diapositivas

Es posible activar una presentación de diapositivas en el dashboard:

- `/zabbix.php?action=dashboard.view&slideshow=on` - activar la presentación de diapositivas

Definiciones

Descripción general

Si bien muchas cosas en la interfaz web pueden configurarse utilizando la propia interfaz, algunas personalizaciones actualmente solo son posibles editando un archivo de definiciones.

Este archivo es `defines.inc.php` ubicado en /include del directorio de documentos HTML de Zabbix.

Parámetros

Parámetros en este archivo que podrían ser de interés para los usuarios:

- ZBX_MIN_PERIOD

Período mínimo del gráfico, en segundos. Un minuto de forma predeterminada.

- GRAPH_YAXIS_SIDE_DEFAULT

Ubicación predeterminada del eje Y en los gráficos simples y valor predeterminado del cuadro desplegable al agregar items a gráficos personalizados. Valores posibles: 0 - izquierda, 1 - derecha.

Predeterminado: 0

- ZBX_SESSION_NAME

Cadena utilizada como nombre de la cookie de sesión del frontend de Zabbix.

Predeterminado: zbx_sessionid

- ZBX_DATA_CACHE_TTL

Tiempo de espera TTL, en segundos, usado para invalidar la caché de datos de Vault response. Establezca 0 para deshabilitar el almacenamiento en caché de la respuesta de Vault.

Predeterminado: 60

- SUBFILTER_VALUES_PER_GROUP

Número de valores de subfiltro por grupo (por ejemplo, en el subfiltro de latest data).

Predeterminado: 1000

- ZBX_MAX_WIDGET_LINES

Número máximo de líneas de widget que se mostrarán.

Predeterminado: 1000

Idiomas adicionales del frontend

Descripción general Para usar cualquier idioma distinto del inglés en la interfaz web de Zabbix, su configuración regional debe estar instalada en el web server. Además, se requiere la extensión gettext de PHP para que las traducciones funcionen.

Instalación de locales Para listar todos los idiomas instalados, ejecute:

```
locale -a
```

Si algunos idiomas necesarios no aparecen en la lista, abra el archivo `/etc/locale.gen` y descomente los locales requeridos. Como Zabbix usa codificación UTF-8, debe seleccionar locales con juego de caracteres UTF-8.

Ahora ejecute:

```
locale-gen
```

Reinicie el web server. Tenga en cuenta que, si PHP-FPM está instalado y es utilizado por el web server, también debe reiniciarse junto con el web server.

Los locales ya deberían estar instalados. Puede ser necesario recargar la página del frontend de Zabbix en el navegador usando `Ctrl` + `F5` para que aparezcan los nuevos idiomas.

Instalación de Zabbix Si instala Zabbix directamente desde el repositorio git de Zabbix, los archivos de traducción deben generarse manualmente. Para generar los archivos de traducción, ejecute:

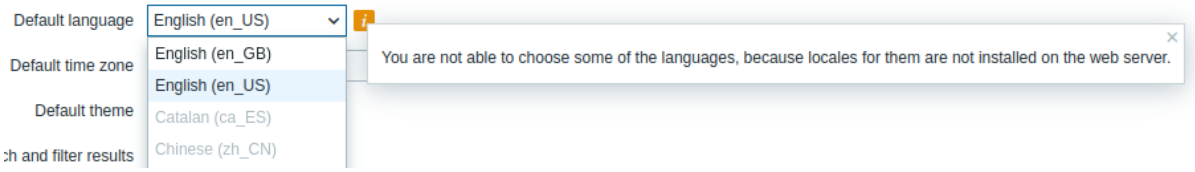
```
make gettext
locale/make_mo.sh
```

Este paso no es necesario al instalar Zabbix desde paquetes o archivos fuente `tar.gz`.

Selección de un idioma Hay varias formas de seleccionar un idioma en la interfaz web de Zabbix:

- Al instalar la interfaz web, en el asistente de instalación del frontend. El idioma seleccionado se establecerá como predeterminado del sistema.
- Después de la instalación, el idioma predeterminado del sistema se puede cambiar en *Administración* > *General* > *GUI*.
- El idioma de un usuario concreto se puede cambiar en el perfil de usuario.

Si no está instalado en la máquina un locale para un idioma, ese idioma aparecerá atenuado en el selector de idiomas de Zabbix. Se muestra un icono naranja junto al selector de idiomas si falta al menos un locale. Al hacer clic en este icono se mostrará el siguiente mensaje: "No puede elegir algunos de los idiomas, porque no están instalados los locales correspondientes en el web server."



Creación de tu propio tema

Descripción general

De forma predeterminada, Zabbix proporciona una serie de temas predefinidos del frontend. Para crear un tema personalizado, siga el procedimiento paso a paso que se proporciona aquí.

Paso 1: Cree su tema

Para crear su propio tema:

1. Cree un archivo CSS desde cero o copie un tema existente (como `blue-theme.css`, `dark-theme.css` u otro) del directorio `assets/styles/` y modifíquelo.
2. Guarde su tema (por ejemplo, `custom-theme.css`) en el directorio `assets/styles/`.

Paso 2: Habilitar el tema en la lista de temas

Para que el tema personalizado esté disponible en el frontend, debe agregarlo a la lista de temas que devuelve el método `APP::getThemes()`. Esto se hace sobrescribiendo el método `ZBase::getThemes()` dentro de la clase `APP`, de la siguiente manera:

1. Abra el archivo `include/classes/core/APP.php` y localice la clase `APP`:

```
class APP extends ZBase {

}
```

2. Dentro de la clase, antes de la llave de cierre final }, agregue el siguiente método:

```
public static function getThemes() {
          return array_merge(parent::getThemes(), [
              'custom-theme' => _('Custom theme')
          ]);
}
```

donde:

- `custom-theme`: identificador interno del tema; debe coincidir con el nombre del archivo CSS sin la extensión `.css`;
- `Custom theme`: nombre visible que verá en el frontend.

Para agregar varios temas, inclúyalos en el array y separe cada entrada con una coma. La coma final después de la última entrada es opcional.

Por ejemplo:

```
      public static function getThemes() {
          return array_merge(parent::getThemes(), [
              'custom-theme' => _('Custom theme'),
              'anothertheme' => _('Another theme'),
              'onemoretheme' => _('One more theme')
          ]);
      }
```

Paso 3: Cambiar los colores de los gráficos

Para cambiar los colores de los gráficos, debe agregarse un registro para el nuevo tema en la tabla de base de datos `graph_theme`.

Ejemplo de cómo agregar un registro a una base de datos MySQL/MariaDB:

```
    mysql -u zabbix -p
    # Enter password:
```

```
mysql> USE zabbix;
mysql> INSERT INTO graph_theme (
    graphthemeid,
    theme,
    backgroundcolor,
    graphcolor,
    gridcolor,
    maingridcolor,
    gridbordercolor,
    textcolor,
    highlightcolor,
    leftpercentilecolor,
    rightpercentilecolor,
    nonworktimecolor,
    colorpalette
)
SELECT
    MAX(graphthemeid) + 1,
    'custom-theme',
    'FFFFFF',
    'FFFFFF',
    'CCD5D9',
    'ACBBC2',
    'ACBBC2',
    '1F2C33',
    'CC745E',
    '5ECCAB',
    'CC745E',
    'EBEBEB',
    '329F7E,C2583D,346D91,B26E44,CC6C91,7A6DC2,C4AA56,7C2731,BF75B8,73A350,B04833,633A60,879CCC,7FAD6C,324
FROM graph_theme;
```

Paso 4: Usar el nuevo tema

En el frontend de Zabbix, puede establecer este tema como el tema predeterminado o seleccionarlo en un perfil de usuario.

Modo de depuración

Visión general

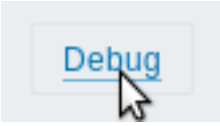
El modo de depuración puede utilizarse para diagnosticar problemas de rendimiento con las páginas del frontend.

Configuración

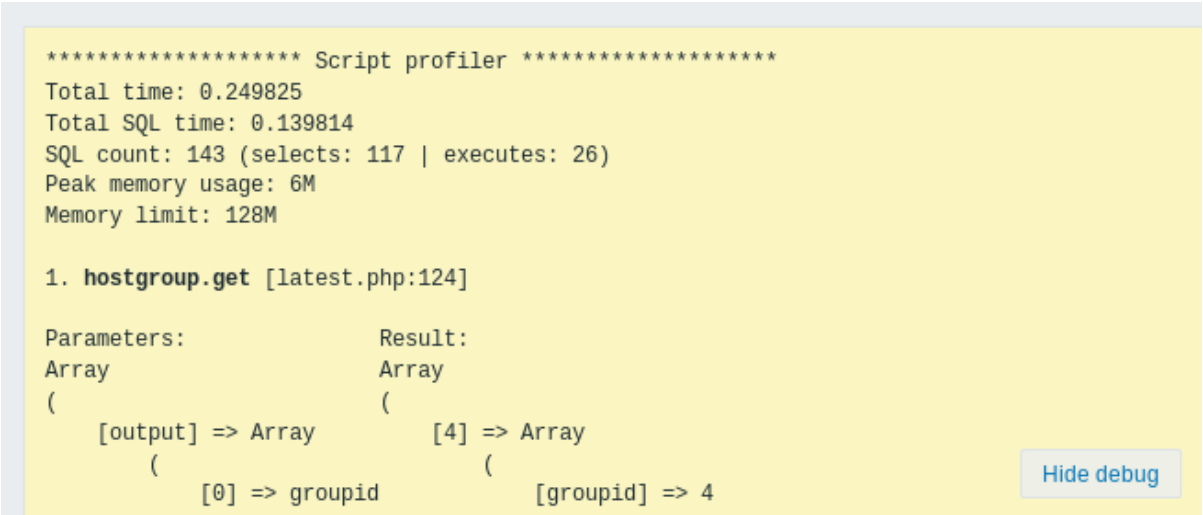
El modo de depuración puede activarse para usuarios individuales que pertenezcan a un grupo de usuarios:

- al configurar un grupo de usuarios;
- al ver los grupos de usuarios configurados.

Cuando *Debug mode* está habilitado para un grupo de usuarios, sus usuarios verán un botón *Debug* en la esquina inferior derecha de la ventana del navegador:



Al hacer clic en el botón *Debug*, se abre una nueva ventana debajo del contenido de la página que contiene las estadísticas SQL de la página, junto con una lista de llamadas a la API y sentencias SQL individuales:



En caso de problemas de rendimiento con la página, esta ventana puede utilizarse para buscar la causa raíz del problema.

Warning:
Habilitar *Debug mode* afecta negativamente al rendimiento del frontend.

Cookies utilizados por Zabbix

Descripción general

Esta página proporciona una lista de las cookies utilizadas por Zabbix.

Nombre	Descripción	Valores	Expira/Max-Age	HttpOnly[1]	Secure[2]
ZBX_SESSION_NAME	Datos de sesión del frontend de Zabbix, almacenados como JSON codificado en base64		Sesión (expira cuando finaliza la sesión de navegación)	+	+ (solo si HTTPS está habilitado en el servidor web)

Nombre	Descripción	Valores	Expira/Max-Age	HttpOnly[1]	Secure[2]
tab	Número de pestaña activa; esta cookie solo se utiliza en páginas con varias pestañas (por ejemplo, la página de configuración de *Equipo*, *Disparador* o *Acción*) y se crea cuando un usuario navega de una pestaña principal a otra pestaña (como la pestaña *Etiquetas* o *Dependencias*). 0 se utiliza para la pestaña principal.	Ejemplo: 1	Sesión (expira cuando finaliza la sesión de navegación)	-	-
browserwarning_ignore	Indica si se debe ignorar una advertencia sobre el uso de un navegador desactualizado.	yes	Sesión (expira cuando finaliza la sesión de navegación)	-	-
system-message-ok	Un mensaje para mostrar tan pronto como se recargue la página.	Mensaje de texto plano	Sesión (expira cuando finaliza la sesión de navegación) o tan pronto como se recargue la página	+	-
system-message-error	Un mensaje de error para mostrar tan pronto como se recargue la página.	Mensaje de texto plano	Sesión (expira cuando finaliza la sesión de navegación) o tan pronto como se recargue la página	+	-

Note:
Forzar la bandera 'HttpOnly' en las cookies de Zabbix mediante una directiva del servidor web no es compatible.

Notas

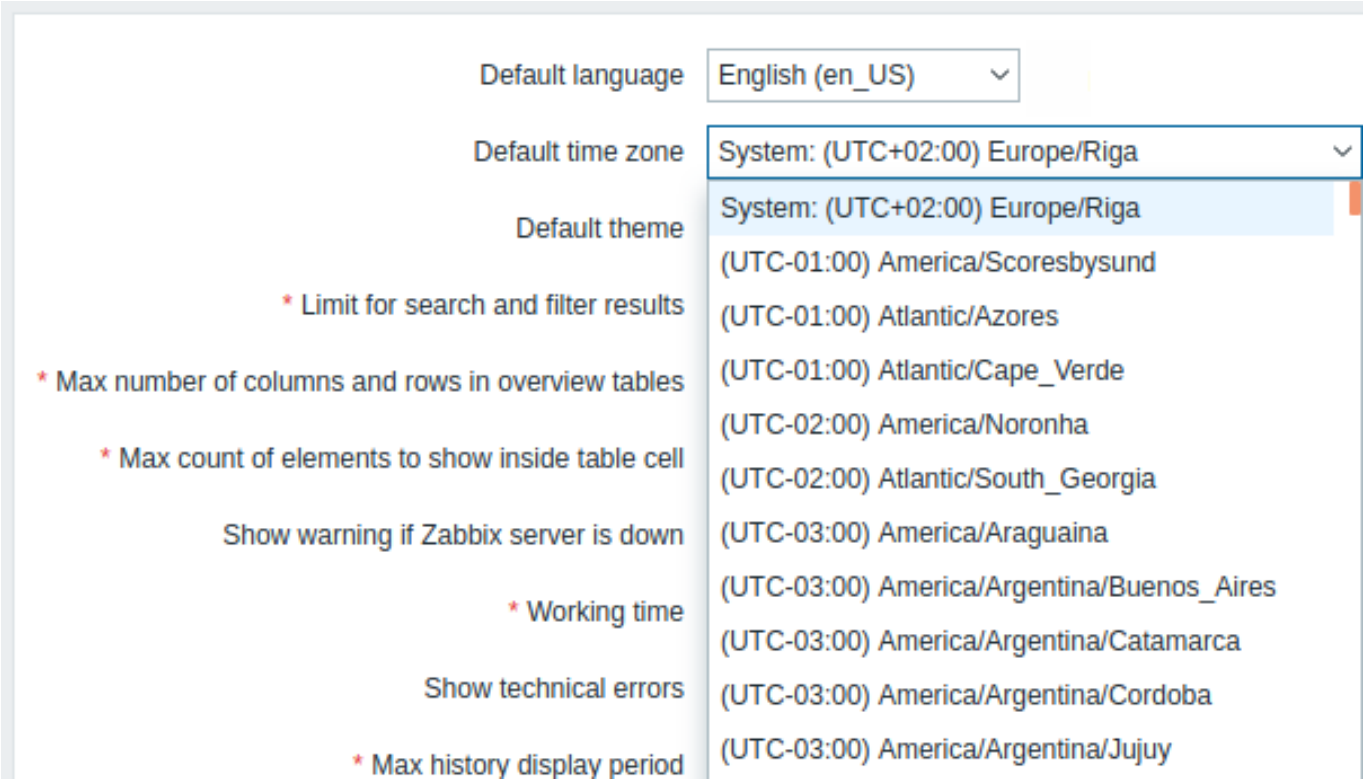
[1] Cuando `HttpOnly` es 'true', la cookie solo será accesible a través del protocolo HTTP. Esto significa que la cookie no será accesible por lenguajes de scripting, como JavaScript. Esta configuración puede ayudar efectivamente a reducir el robo de identidad a través de ataques XSS (aunque no es compatible con todos los navegadores).

[2] `Secure` indica que la cookie solo debe transmitirse a través de una conexión HTTPS segura desde el cliente. Cuando se establece en 'true', la cookie solo se establecerá si existe una conexión segura.

Zonas horarias

Descripción general

La zona horaria del frontend se puede establecer globalmente en el frontend y ajustar para usuarios individuales.



Si se selecciona *System*, se utilizará la zona horaria del servidor web para el frontend (incluido el valor de `date.timezone` en `php.ini`, si está establecido), mientras que Zabbix server utilizará la zona horaria de la máquina en la que se está ejecutando.

Note:
Zabbix server solo utilizará la zona horaria global o del usuario especificada al expandir macros en las notificaciones (por ejemplo, `{EVENT.TIME}` puede expandirse a una zona horaria diferente para cada usuario) y para el límite de tiempo al enviar notificaciones (consulte la configuración "When active" en la configuración de medios del usuario).

La elección de la zona horaria no afecta al formato de fecha y hora del frontend. En su lugar, puede ajustar el idioma de la interfaz (durante la instalación o en la sección configuración del usuario); al seleccionar *English (en_US)* también se habilitará el formato de fecha y hora de EE. UU. en el frontend.

Configuración

La zona horaria global:

- Se puede establecer manualmente al instalar el frontend.
- Se puede modificar en *Administración* > *General* > *GUI*.

La zona horaria a nivel de usuario:

- Se puede establecer al configurar/actualizar un usuario.
- Cada usuario puede establecerla en su perfil de usuario.

Véase también: Alineación de zonas horarias al usar intervalos de programación.

Restablecimiento de contraseña

Descripción general Esta sección describe los pasos para restablecer las contraseñas de los usuarios en Zabbix.

Pasos Póngase en contacto con el administrador de Zabbix si ha olvidado su contraseña de Zabbix y no puede iniciar sesión.

Un usuario Super administrador puede cambiar las contraseñas de todos los usuarios en el formulario de configuración de usuarios.

Si un Super administrador ha olvidado su contraseña y no puede iniciar sesión, debe ejecutar la siguiente consulta SQL para aplicar la contraseña predeterminada al usuario Super administrador (reemplace 'Admin' por el nombre de usuario de Super administrador correspondiente):

```
UPDATE users SET passwd = '$2a$10$ZXIvHAEP2ZM.dLXTm6uPHOMVlARXX7cqjbhM6Fn0cANzkCQBWpMrS' WHERE username =
```

Después de ejecutar esta consulta, la contraseña del usuario se establecerá en *zabbix*. Asegúrese de cambiar la contraseña predeterminada en el primer inicio de sesión.

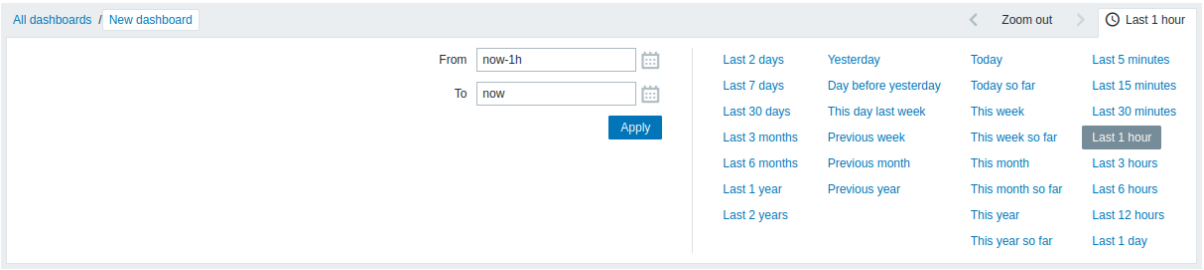
Período de tiempo y selectores de host

Descripción general

Esta página describe el período de tiempo y los selectores de host disponibles en los dashboards. Estos selectores permiten ajustar dinámicamente los datos mostrados en todos los widgets del dashboard compatibles sin editar la configuración de cada widget.

Selector de período de tiempo

El selector de período de tiempo permite seleccionar períodos que se necesitan con frecuencia con un solo clic del ratón. Se puede expandir o contraer haciendo clic en el botón del selector de período de tiempo situado en la esquina superior derecha del dashboard.



Las opciones como *Hoy*, *Esta semana*, etc., muestran el período completo, incluidas las horas o los días futuros. Las opciones como *Hasta ahora hoy*, *Hasta ahora esta semana*, etc., muestran únicamente las horas transcurridas.

Una vez seleccionado un período, se puede desplazar hacia atrás y hacia delante en el tiempo haciendo clic en los botones de flecha . El botón *Alejar* permite reducir el período un 50 % en cada dirección.

Note:
Para los gráficos (excepto aquellos con el período de tiempo *Personalizado*), también es posible seleccionar el período de tiempo mostrado resaltando un área del gráfico con el botón izquierdo del ratón. Al soltar el botón izquierdo del ratón, el gráfico ampliará el área resaltada. También es posible alejar la vista haciendo doble clic en el gráfico.

Los campos *Desde/Hasta* muestran el período seleccionado mediante una sintaxis de tiempo absoluta (con el formato `Y-m-d H:i:s`) o una sintaxis de tiempo relativa. Un período de tiempo relativo puede contener una o varias operaciones matemáticas (- o +), por ejemplo, `now-1d` o `now-1d-2h+5m`.

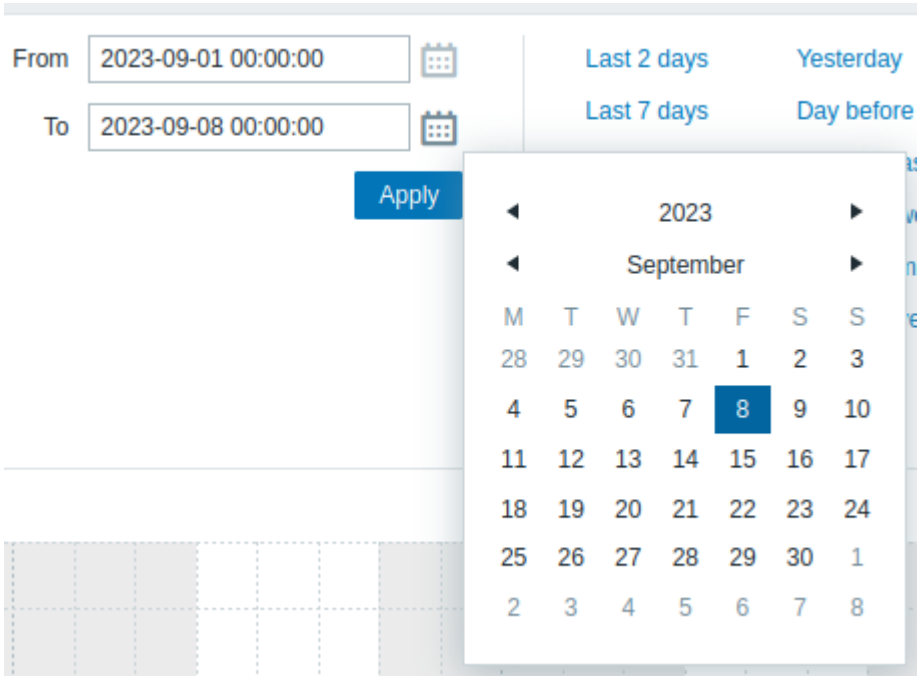
Se admiten las siguientes abreviaturas de tiempo relativo:

- `now`
- `s` (segundos)
- `m` (minutos)
- `h` (horas)
- `d` (días)
- `w` (semanas)
- `M` (meses)
- `y` (años)

El selector de período de tiempo admite precisión (por ejemplo, `/M` en `now-1d/M`). Detalles de la precisión:

Precisión	Desde	Hasta
`m`	Y-m-d H:m:00	Y-m-d H:m:59
`h`	Y-m-d H:00:00	Y-m-d H:59:59
`d`	Y-m-d 00:00:00	Y-m-d 23:59:59
`w`	Lunes de la semana 00:00:00	Domingo de la semana 23:59:59
`M`	Primer día del mes 00:00:00	Último día del mes 23:59:59
`y`	1 de enero del año 00:00:00	31 de diciembre del año 23:59:59

También es posible seleccionar un período de tiempo mediante el selector de fecha. Para abrirlo, haga clic en el icono del calendario situado junto a los campos *Desde/Hasta*.



Note:
En el selector de fecha, puede desplazarse entre el año, el mes y la fecha mediante `Tab`, `Shift+Tab` y los botones de flecha del teclado. Al pulsar `Enter`, se confirma la selección.

Ejemplos:

Desde	Hasta	Período seleccionado
`now/d`	`now/d`	00:00 - 23:59 hoy
`now/d`	`now/d+1d`	00:00 hoy - 23:59 mañana
`now/w`	`now/w`	Lunes 00:00:00 - Domingo 23:59:59 esta semana
`now-1y/w`	`now-1y/w`	La semana de Lunes 00:00:00 - Domingo 23:59:59 hace un año

Attention:
Usar "now/M+1M" para el parámetro *Hasta* puede sumar 31 días, lo que puede provocar que la fecha se desplace entre 1 y 3 días dependiendo del número de días del mes. Por ejemplo, si se usa en enero, el resultado puede ser 02 de marzo en lugar del 28 de febrero esperado. Para evitar este problema, use "now/M-3d+1M/M", que ajusta la longitud del mes con precisión. De manera similar, si configura *Desde* para retroceder, use "now/M+3d-1M/M".

Selector de host

El selector de host permite seleccionar un host para mostrar sus datos en los widgets de dashboard compatibles.

Está disponible después de editar y guardar un dashboard, si al menos un widget está configurado con el parámetro *Override host* establecido para utilizar el dashboard como fuente de datos.



Parámetros de configuración: frontend de Zabbix

Descripción general Los parámetros de configuración permiten personalizar el comportamiento del frontend de Zabbix.

Especifique los valores de los parámetros en el archivo de configuración del frontend (`zabbix.conf.php`).

Esta página está estructurada de la siguiente manera:

- Descripción general de los parámetros (para consultar todos los detalles de un parámetro, haga clic en su nombre)
- Detalles de los parámetros

Parámetro	Descripción
$DB['TYPE']	Tipo de base de datos.
$DB['SERVER']	Nombre del host o dirección IP de la base de datos.
$DB['PORT']	Puerto de la base de datos.
$DB['DATABASE']	Nombre de la base de datos que utiliza el frontend de Zabbix para conectarse a la base de datos configurada.
$DB['USER']	Nombre de usuario que utiliza el frontend de Zabbix para conectarse a la base de datos.
$DB['PASSWORD']	Contraseña que utiliza el frontend de Zabbix para conectarse a la base de datos.
$DB['SCHEMA']	Nombre del esquema de la base de datos utilizado para PostgreSQL.
$DB['ENCRYPTION']	Activa o desactiva el cifrado TLS para las conexiones entre el frontend de Zabbix y la base de datos.
$DB['KEY_FILE']	Ruta al archivo de clave privada del cliente utilizado para la conexión TLS a la base de datos.
$DB['CERT_FILE']	Ruta al archivo de certificado del cliente utilizado para la conexión TLS a la base de datos.
$DB['CA_FILE']	Ruta al archivo de certificado de la CA utilizado para la conexión TLS a la base de datos.
$DB['VERIFY_HOST']	Activa o desactiva la verificación del nombre de host del server de la base de datos en el certificado TLS.
$DB['CIPHER_LIST']	Lista de cifrados TLS permitidos para la conexión a la base de datos.
$DB['VAULT']	Especifique el gestor de secretos Vault utilizado para almacenar las credenciales de la base de datos.
$DB['VAULT_URL']	URL del gestor de secretos Vault utilizado para recuperar las credenciales de la base de datos.
$DB['VAULT_PREFIX']	Prefijo de la ruta o consulta de Vault utilizada para recuperar las credenciales de la base de datos.
$DB['VAULT_DB_PATH']	Ruta a las credenciales de la base de datos en el gestor de secretos Vault.
$DB['VAULT_TOKEN']	Token de autenticación utilizado para acceder al gestor de secretos Vault.
$DB['VAULT_APP_ROLE_ID']	ID de rol de AppRole utilizado para autenticarse en HashiCorp Vault.
$DB['VAULT_APP_SECRET_ID']	ID secreto de AppRole utilizado para autenticarse en HashiCorp Vault.
$DB['VAULT_CERT_FILE']	Ruta al archivo de certificado del cliente utilizado para autenticarse en el gestor de secretos Vault.
$DB['VAULT_KEY_FILE']	Ruta al archivo de clave privada del cliente utilizado para autenticarse en el gestor de secretos Vault.
$DB['VAULT_CACHE']	Activa o desactiva el almacenamiento en caché de las credenciales recuperadas del gestor de secretos Vault.
$ZBX_SERVER	Especifique un valor para sobrescribir el nombre de host o la dirección IP de Zabbix.
$ZBX_SERVER_PORT	Especifique un valor para sobrescribir el puerto del server de Zabbix.
$ZBX_SERVER_NAME	Nombre del server de Zabbix.
$IMAGE_FORMAT_DEFAULT	Formato de imagen predeterminado.
$HISTORY_PROVIDERS[]	Configuración de los proveedores de almacenamiento del historial para Elasticsearch o ClickHouse.
$SSO['SETTINGS']	Parámetros de configuración del inicio de sesión mediante SAML.
$SSO['CERT_STORAGE']	Especifique dónde se almacenan los certificados SAML.
$SSO['SP_KEY']	Ruta al archivo de clave privada del proveedor de servicios (SP) SAML cuando la clave privada está almacenada en el sistema de archivos.
$SSO['SP_CERT']	Ruta al archivo de certificado del proveedor de servicios (SP) SAML cuando el certificado está almacenado en el sistema de archivos.
$SSO['IDP_CERT']	Ruta al archivo de certificado del proveedor de identidad SAML cuando el certificado está almacenado en el sistema de archivos.
$ZBX_FEATURE_FLAGS['banners_enabled']	Activa o desactiva la compatibilidad con banners.
$ZBX_FEATURE_FLAGS['http_auth_enabled']	Activa o desactiva la autenticación HTTP de usuarios.
$ZBX_FEATURE_FLAGS['modules_configurable']	Activa o desactiva el acceso a los módulos.
$ZBX_FEATURE_FLAGS['media_type_definitions']	Desactiva la edición de tipos de medios preconfigurados específicos.
$ZBX_SERVER_TLS['ACTIVE']	Especifica si TLS está habilitado para las conexiones al server de Zabbix.
$ZBX_SERVER_TLS['CA_FILE']	Ruta al archivo de certificado de la autoridad certificadora (CA) utilizado para verificar el certificado del server.
$ZBX_SERVER_TLS['KEY_FILE']	Ruta al archivo de clave privada del cliente correspondiente al certificado del cliente.
$ZBX_SERVER_TLS['CERT_FILE']	Ruta al archivo de certificado del cliente cuando se requiere autenticación TLS mutua.
$ZBX_SERVER_TLS['CERTIFICATE_ISSUER']	Nombre distinguido (DN) del emisor que debe coincidir con el certificado del server.
$ZBX_SERVER_TLS['CERTIFICATE_SUBJECT']	Nombre distinguido (DN) del sujeto que debe coincidir con el certificado del server.
$TELEMETRY_PROVIDERS[]	Configuración de una fuente de datos de telemetría.

Detalles de los parámetros

$DB['TYPE']

Tipo de base de datos.

Valores: MYSQL - MySQL, POSTGRESQL - PostgreSQL.

$DB['SERVER']

Nombre de host o dirección IP de la base de datos.

Si $DB['SERVER'] no está definido, se utiliza el socket de dominio Unix predeterminado. En MySQL, establecer $DB['SERVER'] en localhost también da como resultado una conexión mediante el socket Unix predeterminado.
 En PostgreSQL, $DB['SERVER'] puede contener varias entradas host:port separadas por comas. Si no se define un puerto para un host de la lista separada por comas, se utiliza como puerto predeterminado el valor definido en $DB['PORT']. Si se especifica una ruta de socket en $DB['SERVER'] (por ejemplo, /var/run/pgbouncer), se utiliza ese socket de dominio Unix.

Ejemplos:

```
$DB['SERVER'] = 'localhost';
$DB['SERVER'] = 127.0.0.1;
```

$DB['PORT']

Puerto de la base de datos.

Si no se define, se utiliza el valor 0.

$DB['DATABASE']

Nombre de la base de datos que utiliza el frontend de Zabbix para conectarse a la base de datos configurada.

Ejemplo:

```
$DB['DATABASE'] = 'zabbix';
```

$DB['USER']

Nombre de usuario utilizado por el frontend de Zabbix para conectarse a la base de datos.

$DB['PASSWORD']

Contraseña utilizada por el frontend de Zabbix para conectarse a la base de datos.

$DB['SCHEMA']

Nombre del esquema de base de datos utilizado para PostgreSQL.

$DB['ENCRYPTION']

Habilita o deshabilita el cifrado de las conexiones entre el frontend de Zabbix y la base de datos.

Valores: true: habilitado, false: deshabilitado.

$DB['KEY_FILE']

Ruta al archivo de clave privada del cliente utilizado para la conexión TLS de la base de datos.

Ejemplo:

```
// Para MySQL:
$DB['KEY_FILE'] = '/etc/ssl/mysql/client-key.pem';
```

$DB['CERT_FILE']

Ruta al archivo de certificado del cliente utilizado para la conexión TLS de la base de datos.

Ejemplo:

```
// Para MySQL:
$DB['CERT_FILE'] = '/etc/ssl/mysql/client-cert.pem';
```

$DB['CA_FILE']

Ruta al archivo de certificado de CA utilizado para la conexión TLS de la base de datos.

Ejemplos:

```
// Para MySQL:
$DB['CA_FILE'] = '/etc/ssl/mysql/ca.pem';
// Para PostgreSQL:
$DB['CA_FILE'] = '/etc/ssl/pgsql/root.crt';
```

$DB['VERIFY_HOST']

Habilita o deshabilita la verificación del nombre de host del servidor de base de datos en el certificado TLS.

Valores: `true` - habilitado, `false` - deshabilitado.

$DB['CIPHER_LIST']

Lista de cifrados TLS permitidos para la conexión a la base de datos.

Ejemplos:

```
// Para MySQL:
$DB['CIPHER_LIST'] = 'TLS_AES_256_GCM_SHA384:TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256:TLS_AES_1
```

$DB['VAULT']

Especifique el gestor de secretos Vault utilizado para almacenar las credenciales de la base de datos.

Valores: `CyberArk` o `HashiCorp`.

Ejemplos:

```
// Para CyberArk:
$DB['VAULT'] = 'CyberArk';
// Para HashiCorp:
$DB['VAULT'] = 'HashiCorp';
```

$DB['VAULT_URL']

URL del gestor de secretos Vault utilizado para recuperar las credenciales de la base de datos.

Ejemplos:

```
// Para CyberArk:
$DB['VAULT_URL'] = 'https://127.0.0.1:1858';
// Para HashiCorp:
$DB['VAULT_URL'] = 'https://localhost:8200';
```

$DB['VAULT_PREFIX']

Prefijo de la ruta o consulta de Vault utilizada para recuperar las credenciales de la base de datos.

Ejemplo:

```
// Para HashiCorp:
$DB['VAULT_PREFIX'] = '/v1/secret/data/zabbix/';
```

$DB['VAULT_DB_PATH']

Ruta a las credenciales de la base de datos en el gestor de secretos Vault.

Ejemplos:

```
// Para CyberArk:
$DB['VAULT_DB_PATH'] = 'AppID=foo&Query=Safe=bar;Object=buzz';
// Para HashiCorp:
$DB['VAULT_DB_PATH'] = 'database';
```

$DB['VAULT_TOKEN']

Token de autenticación utilizado para acceder al gestor de secretos Vault.

Ejemplo:

```
// Para HashiCorp:
$DB['VAULT_TOKEN'] = 'hvs.CAESIIG_PILmULFYOsEyWHxkZ2mF2a8VPKNLE8eHqd4autYGGh4KHGh2cy5aeTY0NFNSaUp3ZnpWbDF1
```

$DB['VAULT_APP_ROLE_ID']

ID del rol VaultAppSecretID utilizado para autenticarse en HashiCorp Vault.

Ejemplo:

```
$DB['VAULT_APP_ROLE_ID'] = '6f1567f1-538d-7d70-e8e4-5d3e59471b2e';
```

$DB['VAULT_APP_SECRET_ID']

VaultAppSecretID utilizado para autenticarse en HashiCorp Vault.

Ejemplo:

```
$DB['VAULT_APP_SECRET_ID'] = '89dac8b1-f7d6-96a7-2ecf-f94d2de3d6f1';
```

$DB['VAULT_CERT_FILE']

Ruta al archivo de certificado del cliente utilizado para autenticarse en el gestor de secretos Vault.

Ejemplo:

```
// Para CyberArk:
$DB['VAULT_CERT_FILE'] = 'conf/certs/cyberark-cert.pem';
```

$DB['VAULT_KEY_FILE']

Ruta al archivo de clave privada del cliente utilizado para autenticarse en el gestor de secretos Vault.

Ejemplo:

```
// Para CyberArk:
$DB['VAULT_KEY_FILE'] = 'conf/certs/cyberark-key.pem';
```

$DB['VAULT_CACHE']

Activa o desactiva el almacenamiento en caché de las credenciales recuperadas del gestor de secretos Vault.

Valores: `true` - activado, `false` - desactivado.

$ZBX_SERVER

Especifique un valor para anular el nombre de host o la dirección IP de Zabbix.
 Si se especifica `$ZBX_SERVER` sin `$ZBX_SERVER_PORT`, se utiliza el puerto predeterminado (10051).

$ZBX_SERVER_PORT

Especifique un valor para anular el puerto del server de Zabbix.
 Este parámetro se utiliza junto con `$ZBX_SERVER`.

$ZBX_SERVER_NAME

Nombre del server de Zabbix.

$IMAGE_FORMAT_DEFAULT

Formato de imagen predeterminado.

Predeterminado: `IMAGE_FORMAT_PNG`
 Valores: - `IMAGE_FORMAT_PNG` - Formato PNG - `IMAGE_FORMAT_JPEG` - Formato JPEG - `IMAGE_FORMAT_GIF` - Formato GIF

$HISTORY_PROVIDERS[]

Configuración de los proveedores de almacenamiento del historial para Elasticsearch o ClickHouse.

Parámetros de configuración compatibles: - `types`: matriz de tipos de datos que se almacenarán en el almacenamiento externo; - `provider`: tipo de proveedor del historial: `elasticsearch` o `clickhouse`; - `url`: URL del proveedor del historial; - `db`: nombre de la base de datos (se utiliza para ClickHouse); - `username`: usuario de la base de datos (se utiliza para ClickHouse); - `password`: contraseña de la base de datos (se utiliza para ClickHouse).

Ejemplos:

```
// Para ClickHouse:
$HISTORY_PROVIDERS[] = [
    'types' => ['uint', 'dbl', 'str', 'log', 'text', 'json'],
    'provider' => 'clickhouse',
    'url' => 'http://localhost:8123',
    'db' => 'zabbix',
    'username' => 'zabbix',
    'password' => 'zabbix'
];
// Para Elasticsearch:
$HISTORY_PROVIDERS[] = [
    'types' => ['uint', 'dbl', 'str', 'log', 'text', 'json'],
    'provider' => 'elasticsearch',
```

```
    'url' => 'http://localhost:9200'
];
```

$SSO['SETTINGS']

Configuración para el inicio de sesión mediante SAML. El array `$SSO['SETTINGS']` debe seguir la estructura esperada por la biblioteca *SAML PHP Toolkit* (incluida con Zabbix).

Solo se pueden especificar las siguientes opciones en `$SSO['SETTINGS']`:

- `strict`
- `baseurl`
- `compress`
- `contactPerson`
- `organization`
- `sp` (solo las opciones especificadas en esta lista)
  - `attributeConsumingService`
  - `x509certNew`
- `idp` (solo las opciones especificadas en esta lista)
  - `singleLogoutService` (solo una opción)
    * `responseUrl`
  - `certFingerprint`
  - `certFingerprintAlgorithm`
  - `x509certMulti`
- `security` (solo las opciones especificadas en esta lista)
  - `signMetadata`
  - `wantNameId`
  - `requestedAuthnContext`
  - `requestedAuthnContextComparison`
  - `wantXMLValidation`
  - `relaxDestinationValidation`
  - `destinationStrictlyMatches`
  - `rejectUnsolicitedResponsesWithInResponseTo`
  - `signatureAlgorithm`
  - `digestAlgorithm`
  - `lowercaseUrlencoding`

Todas las demás opciones se obtienen de la base de datos y no se pueden sobrescribir. La opción `debug` se ignorará.

Si la interfaz de usuario de Zabbix está detrás de un proxy o un balanceador de carga, se puede utilizar la opción personalizada `use_proxy_headers`:

- `false` (predeterminado): ignora los encabezados HTTP `X-Forwarded-*`;
- `true`: utiliza los encabezados HTTP `X-Forwarded-*` para crear la URL base.

Para obtener una descripción completa de las opciones de configuración disponibles, consulte la documentación oficial de la biblioteca.

Ejemplo:

```
$SSO['SETTINGS'] = [
    'security' => [
        'signatureAlgorithm' => 'http://www.w3.org/2001/04/xmldsig-more#rsa-sha384',
        'digestAlgorithm' => 'http://www.w3.org/2001/04/xmldsig-more#sha384',
        // ...
    ],
    // ...
];
```

$SSO['CERT_STORAGE']

Especifique dónde se almacenan los certificados SAML.

Valor predeterminado: `file<br>` Valores: - `database`: la clave privada y los certificados se almacenan en la base de datos; - `file`: la clave privada y los certificados se almacenan en el sistema de archivos.

Ejemplo:

```
$SSO['CERT_STORAGE'] = 'database';
```

$SSO['SP_KEY']

Ruta al archivo de clave privada del proveedor de servicios (SP) SAML cuando la clave privada está almacenada en el sistema de archivos.

Ejemplo:

```
$SSO['SP_KEY'] = 'conf/certs/sp.key';
```

$SSO['SP_CERT']

Ruta al archivo de certificado del proveedor de servicios (SP) SAML cuando el certificado está almacenado en el sistema de archivos.

Ejemplo:

```
$SSO['SP_CERT'] = 'conf/certs/sp.crt';
```

$SSO['IDP_CERT']

Ruta al archivo de certificado del proveedor de identidad SAML cuando el certificado está almacenado en el sistema de archivos.

Ejemplo:

```
$SSO['IDP_CERT'] = 'conf/certs/idp.crt';
```

$ZBX_FEATURE_FLAGS['banners_enabled']

Habilita o deshabilita la compatibilidad con banners.

Valor predeterminado: `true`
 Valores: `true` - habilitado, `false` - deshabilitado.

$ZBX_FEATURE_FLAGS['http_auth_enabled']

Habilita o deshabilita la autenticación HTTP de los usuarios.

Predeterminado: `true`
 Valores: `true` - habilitada, `false` - deshabilitada.

$ZBX_FEATURE_FLAGS['modules_config_enabled']

Habilita o deshabilita el acceso a los módulos.

Valor predeterminado: `true`
 Valores: `true` - habilitado, `false` - deshabilitado.

$ZBX_FEATURE_FLAGS['media_type_denylist']

Deshabilita la edición de tipos de medios preconfigurados específicos.

Valores: `email`, `script`, `sms`, `webhook`, `push`.

Ejemplo:

```
$ZBX_FEATURE_FLAGS['media_type_denylist'] = ['webhook','push'];
```

$ZBX_SERVER_TLS['ACTIVE']

Especifica si está habilitada una conexión TLS entre el frontend de Zabbix y el server de Zabbix.

Valor predeterminado: `false`
 Valores: `true` - habilitada, `false` - deshabilitada.

$ZBX_SERVER_TLS['CA_FILE']

Ruta al archivo de certificado de la entidad de certificación (CA) utilizado para verificar el certificado del server.

Ejemplo:

```
$ZBX_SERVER_TLS['CA_FILE'] = '/etc/zabbix/ssl/ca.crt';
```

$ZBX_SERVER_TLS['KEY_FILE']

Ruta al archivo de clave privada del cliente correspondiente al certificado de cliente utilizado para la autenticación TLS mutua.

Ejemplo:

```
$ZBX_SERVER_TLS['KEY_FILE'] = '/etc/zabbix/ssl/frontend.key';
```

$ZBX_SERVER_TLS['CERT_FILE']

Ruta al archivo de certificado del cliente cuando se requiere autenticación TLS mutua.

Ejemplo:

```
$ZBX_SERVER_TLS['CERT_FILE'] = '/etc/zabbix/ssl/frontend.crt';
```

$ZBX_SERVER_TLS['CERTIFICATE_ISSUER']

Nombre distinguido (DN) del emisor que se debe comparar con el certificado del server.

Ejemplo:

```
$ZBX_SERVER_TLS['CERTIFICATE_ISSUER']  = 'CN=MyZabbixCA';
```

$ZBX_SERVER_TLS['CERTIFICATE_SUBJECT']

Nombre distinguido (DN) del sujeto que se debe comparar con el certificado del server.

Ejemplo:

```
$ZBX_SERVER_TLS['CERTIFICATE_SUBJECT'] = 'CN=zabbix-server.example.com';
```

$TELEMETRY_PROVIDERS[]

Configuración de una fuente de datos de telemetría.

Parámetros de configuración compatibles con ClickHouse: - `url`: establece la URL HTTP o HTTPS utilizada para conectarse a la fuente de datos de telemetría. - `db`: establece el nombre de la base de datos. - `username`: establece el nombre de usuario utilizado para iniciar sesión en la base de datos. - `password`: establece la contraseña utilizada para iniciar sesión en la base de datos. - `ssl_cert_file`, `ssl_key_file`, `ssl_key_password`, `ssl_verify_peer`, `ssl_verify_host`, `ssl_ca_location`, `ssl_ca_file`: establece estas opciones para configurar una conexión segura (TLS/SSL) a la base de datos. - `vault_path`: establece la ruta a un secreto de Vault que contiene los valores de `username` y `password`. No se debe utilizar junto con las opciones `username` y `password`.

Ejemplo:

```
$TELEMETRY_PROVIDERS[] = [
    'provider' => 'clickhouse',
    'url' => 'http://localhost:8123',
    'db' => 'zabbix',
    'username' => 'zabbix',
    'password' => 'zabbix'
];
```

API

Descripción general La API de Zabbix le permite recuperar y modificar programáticamente la configuración de Zabbix y proporciona acceso a datos históricos. Se utiliza ampliamente para:

- Crear nuevas aplicaciones para trabajar con Zabbix.
- Integrar Zabbix en software de terceros.
- Automatizar tareas rutinarias.

La API de Zabbix es una API basada en HTTP y se distribuye como parte del web frontend. Utiliza el protocolo JSON-RPC 2.0, lo que significa dos cosas:

- La API consta de un conjunto de métodos independientes.
- Las solicitudes y respuestas entre los clientes y la API se codifican usando el formato JSON.

Para obtener más información sobre el protocolo y JSON, consulte la especificación JSON-RPC 2.0 y la página principal del formato JSON.

Para obtener más información sobre cómo integrar la funcionalidad de Zabbix en sus aplicaciones Python, consulte la biblioteca Python para Zabbix.

> **Note:**
> El acceso de usuario en Zabbix, tanto para la configuración como para los datos históricos, depende del tipo de usuario, el rol de usuario asignado y los grupos de usuarios.

Estructura La API consta de varios métodos que, nominalmente, se agrupan en APIs separadas. Cada uno de los métodos realiza una tarea específica. Por ejemplo, el método `host.create` pertenece a la API de *host* y se utiliza para crear nuevos hosts. Históricamente, a veces se hace referencia a las APIs como "clases".

La mayoría de las APIs contienen al menos cuatro métodos: `get`, `create`, `update` y `delete` para recuperar, crear, actualizar y eliminar datos, respectivamente, pero algunas APIs pueden proporcionar un conjunto de métodos totalmente diferente.

Realización de solicitudes Una vez que haya configurado el frontend, puede usar solicitudes HTTP remotas para llamar a la API. Para ello, debe enviar solicitudes HTTP POST al archivo `api_jsonrpc.php` ubicado en el directorio del frontend.
 Por ejemplo, si su frontend de Zabbix está instalado en `https://example.com/zabbix`, una solicitud HTTP para llamar al método `apiinfo.version` puede verse así:

```
curl --request POST \
  --url 'https://example.com/zabbix/api_jsonrpc.php' \
  --header 'Content-Type: application/json-rpc' \
  --data '{"jsonrpc":"2.0","method":"apiinfo.version","params":{},"id":1}'
```

La solicitud debe tener el encabezado `Content-Type` establecido en uno de estos valores: `application/json-rpc`, `application/json` o `application/jsonrequest`.

El objeto de solicitud debe contener las siguientes propiedades:

- `jsonrpc` - la versión del protocolo JSON-RPC utilizada por la API (la API de Zabbix implementa JSON-RPC versión 2.0);
- `method` - el método de la API que se está llamando;
- `params` - los parámetros que se pasarán al método de la API;
- `id` - un identificador arbitrario de la solicitud (si se omite, la API trata la solicitud como una notificación).

Si la solicitud es correcta, la respuesta devuelta por la API debería verse así:

```
{
    "jsonrpc": "2.0",
    "result": "8.0.0",
    "id": 1
}
```

A su vez, el objeto de respuesta contiene las siguientes propiedades:

- `jsonrpc` - la versión del protocolo JSON-RPC;
- `result` - los datos devueltos por el método;
- `id` - un identificador de la solicitud correspondiente.

Autenticación Para acceder a cualquier dato en Zabbix, debe:

- Usar un token de API existente creado en el frontend de Zabbix, sección *Users* > *API tokens*, o creado mediante la Token API.
- Usar un token de autenticación obtenido con el método `user.login`.

Por ejemplo, si quisiera obtener un nuevo token de autenticación iniciando sesión como un usuario estándar *Admin*, una solicitud JSON se vería así:

```
curl --request POST \
  --url 'https://example.com/zabbix/api_jsonrpc.php' \
  --header 'Content-Type: application/json-rpc' \
  --data '{"jsonrpc":"2.0","method":"user.login","params":{"username":"Admin","password":"zabbix"},"id":1}
```

Si proporcionó las credenciales correctamente, la respuesta devuelta por la API debería contener el token de autenticación del usuario:

```
{
    "jsonrpc": "2.0",
    "result": "0424bd59b807674191e7d77572075f33",
    "id": 1
}
```

Métodos de autorización Por cabecera "Authorization"

Todas las solicitudes a la API requieren autenticación o un token de API. Puede proporcionar las credenciales utilizando la cabecera Authorization en la solicitud:

```
curl --request POST \
  --url 'https://example.com/zabbix/api_jsonrpc.php' \
  --header 'Authorization: Bearer 0424bd59b807674191e7d77572075f33'
```

Attention:
Si experimenta problemas de autenticación, consulte Reenvío de la cabecera Authorization.

La API de Zabbix acepta las cabeceras sin distinguir mayúsculas de minúsculas (por ejemplo, `authorization`, `Authorization` y `AUTHORIZATION` se tratan igual).

La cabecera Authorization es compatible en solicitudes de origen cruzado (CORS).

Por cookie de Zabbix

Se utiliza una cookie *"zbx_session"* para autorizar una solicitud de API desde la interfaz de Zabbix realizada mediante JavaScript (desde un módulo o un widget personalizado).

Flujo de trabajo de ejemplo La siguiente sección le guía a través de varios ejemplos de uso con más detalle.

Recuperación de hosts Ahora dispone de un token de autenticación de usuario válido (representado como una variable en los siguientes ejemplos) que puede utilizarse para acceder a los datos en Zabbix. Por ejemplo, puede usar el método `host.get` para recuperar los IDs, nombres de host e interfaces de todos los hosts configurados:

Solicitud:

```
curl --request POST \
  --url 'https://example.com/zabbix/api_jsonrpc.php' \
  --header 'Authorization: Bearer ${AUTHORIZATION_TOKEN}' \
  --header 'Content-Type: application/json-rpc' \
  --data @data.json
```

Note:
`data.json` es un archivo que contiene una consulta JSON. En lugar de un archivo, puede pasar la consulta en el argumento `--data`.

`data.json`

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": [
            "hostid",
            "host"
        ],
        "selectInterfaces": [
            "interfaceid",
            "ip"
        ]
    },
    "id": 2
}
```

El objeto de respuesta contendrá los datos solicitados sobre los hosts:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "hostid": "10084",
            "host": "Zabbix server",
            "interfaces": [
                {
                    "interfaceid": "1",
                    "ip": "127.0.0.1"
```

```
                }
            ]
        }
    ],
    "id": 2
}
```

Note:
Por motivos de rendimiento, siempre se recomienda enumerar las propiedades del objeto que desea recuperar. De este modo, evitará recuperar todo.

Creación de un nuevo item Ahora, cree un nuevo item en el host "Zabbix server" usando los datos que ha obtenido de la solicitud `host.get` anterior. Esto se puede hacer usando el método `item.create`:

```
curl --request POST \
  --url 'https://example.com/zabbix/api_jsonrpc.php' \
  --header 'Authorization: Bearer ${AUTHORIZATION_TOKEN}' \
  --header 'Content-Type: application/json-rpc' \
  --data '{"jsonrpc":"2.0","method":"item.create","params":{"name":"Free disk space on /home/joe/","key_":
```

Una respuesta correcta contendrá el ID del item recién creado, que se puede usar para referenciar el item en las siguientes solicitudes:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "24759"
        ]
    },
    "id": 3
}
```

Note:
El método `item.create`, así como otros *métodos de creación*, también pueden aceptar matrices de objetos y crear varios items con una sola llamada a la API.

Creación de múltiples triggers Así, si los *métodos de creación* aceptan matrices, puede agregar múltiples triggers, por ejemplo, este:

```
curl --request POST \
  --url 'https://example.com/zabbix/api_jsonrpc.php' \
  --header 'Authorization: Bearer ${AUTHORIZATION_TOKEN}' \
  --header 'Content-Type: application/json-rpc' \
  --data '{"jsonrpc":"2.0","method":"trigger.create","params":[{"description":"Processor load is too high
```

La respuesta correcta contendrá los IDs de los triggers recién creados:

```
{
    "jsonrpc": "2.0",
    "result": {
        "triggerids": [
            "17369",
            "17370"
        ]
    },
    "id": 4
}
```

Actualización de un item Habilite un item estableciendo su estado en 0:

```
curl --request POST \
  --url 'https://example.com/zabbix/api_jsonrpc.php' \
```

```
  --header 'Authorization: Bearer ${AUTHORIZATION_TOKEN}' \
  --header 'Content-Type: application/json-rpc' \
  --data '{"jsonrpc":"2.0","method":"item.update","params":{"itemid":"10092","status":0},"id":5}'
```

La respuesta correcta contendrá el ID del item actualizado:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "10092"
        ]
    },
    "id": 5
}
```

Note:
El método `item.update`, así como otros *métodos de actualización*, también puede aceptar matrices de objetos y actualizar varios items con una sola llamada a la API.

Actualización de múltiples triggers Habilite múltiples triggers estableciendo su estado en 0:

```
curl --request POST \
  --url 'https://example.com/zabbix/api_jsonrpc.php' \
  --header 'Authorization: Bearer ${AUTHORIZATION_TOKEN}' \
  --header 'Content-Type: application/json-rpc' \
  --data '{"jsonrpc":"2.0","method":"trigger.update","params":[{"triggerid":"13938","status":0},{"triggeri
```

La respuesta correcta contendrá los IDs de los triggers actualizados:

```
{
    "jsonrpc": "2.0",
    "result": {
        "triggerids": [
            "13938",
            "13939"
        ]
    },
    "id": 6
}
```

Manejo de errores Hasta este momento, todo lo que has probado ha funcionado correctamente. Pero, ¿qué pasaría si intentaras hacer una llamada incorrecta a la API? Intenta crear otro host llamando a `host.create` pero omitiendo el parámetro obligatorio `groups`:

```
curl --request POST \
  --url 'https://example.com/zabbix/api_jsonrpc.php' \
  --header 'Authorization: Bearer ${AUTHORIZATION_TOKEN}' \
  --header 'Content-Type: application/json-rpc' \
  --data '{"jsonrpc":"2.0","method":"host.create","params":{"host":"Linux server","interfaces":[{"type":1,
```

La respuesta contendrá entonces un mensaje de error:

```
{
    "jsonrpc": "2.0",
    "error": {
        "code": -32602,
        "message": "Invalid params.",
        "data": "No groups for host \"Linux server\"."
    },
    "id": 7
}
```

Si se ha producido un error, en lugar de la propiedad `result`, el objeto de respuesta contendrá la propiedad `error` con los siguientes datos:

- `code` - un código de error;
- `message` - un breve resumen del error;
- `data` - un mensaje de error más detallado.

Los errores pueden producirse en varios casos, como el uso de valores de entrada incorrectos, un tiempo de espera de sesión agotado o intentar acceder a objetos inexistentes. Tu aplicación debe ser capaz de gestionar correctamente este tipo de errores.

Versiones de la API Para simplificar el versionado de la API, desde Zabbix 2.0.4, la versión de la API coincide con la versión de Zabbix en sí. Puede usar el método `apiinfo.version` para averiguar la versión de la API con la que está trabajando. Esto puede ser útil para ajustar su aplicación y usar funciones específicas de una versión.

Zabbix garantiza la compatibilidad con versiones anteriores de las funciones dentro de una versión principal. Al realizar cambios incompatibles con versiones anteriores entre versiones principales, Zabbix suele dejar las funciones antiguas como obsoletas en la siguiente versión, y solo las elimina en la versión posterior. En ocasiones, Zabbix puede eliminar funciones entre versiones principales sin proporcionar compatibilidad con versiones anteriores. Es importante que nunca dependa de ninguna función obsoleta y que migre a alternativas más nuevas lo antes posible.

> **Note:**
> Puede seguir todos los cambios realizados en la API en el registro de cambios de la API.

Lectura adicional Ahora ya tiene suficientes conocimientos para empezar a trabajar con la API de Zabbix; sin embargo, no se detenga aquí. Para ampliar la información, le recomendamos consultar la lista de APIs disponibles.

Referencia de métodos

Esta sección proporciona una visión general de las funciones proporcionadas por la API de Zabbix y le ayudará a orientarse entre las clases y métodos disponibles.

Monitorización La API de Zabbix le permite acceder al historial y otros datos recopilados durante la monitorización.

Paneles

Gestione paneles y genere informes programados basados en ellos.

API de paneles | API de paneles de plantilla | API de informes

Clúster de alta disponibilidad

Recupera una lista de nodos del servidor y su estado.

API de clúster de alta disponibilidad

Historial

Recupera los valores históricos recopilados por los procesos de monitorización de Zabbix para su presentación o procesamiento posterior.

API de Historial

Tendencias

Recuperar valores de tendencias calculados por el servidor Zabbix para su presentación o procesamiento posterior.

API de Tendencias

Eventos

Recuperar eventos generados por disparadores, descubrimiento de red y otros sistemas Zabbix para una gestión de situaciones más flexible o integración con herramientas de terceros.

API de eventos

Problemas

Recuperar problemas según los parámetros dados.

API de Problemas

Mapas

Configure mapas para crear representaciones dinámicas y detalladas de su infraestructura de TI.

API de Mapas

Tareas

Interactúe con el gestor de tareas del servidor Zabbix, creando tareas y recuperando respuestas.

API de Tareas

APM La API de Zabbix permite acceder a los datos de telemetría recopilados para la monitorización del rendimiento de las aplicaciones (APM).

Trazas

Recupere las trazas y los spans que las componen.

API de trazas de APM | API de spans de APM

Métricas

Recuperar puntos de datos de métricas.

API de métricas de APM

Registros

Recupere registros de log.

API de logs de APM

Servicios La API de Zabbix le permite acceder a los datos recopilados durante la monitorización de servicios.

Acuerdo de Nivel de Servicio

Defina Objetivos de Nivel de Servicio (SLO), obtenga información detallada de los Indicadores de Nivel de Servicio (SLI) sobre el rendimiento del servicio.

SLA API

Servicios

Administre los servicios para la supervisión de los niveles de servicio y obtenga información detallada sobre el SLA de cualquier servicio.

API de servicios

Recopilación de datos La API de Zabbix permite gestionar la configuración de su sistema de monitorización.

Hosts y grupos de hosts

Administre grupos de hosts, hosts y todo lo relacionado con ellos, incluidas las interfaces de host, las macros de host y los períodos de mantenimiento.

API de host | API de grupos de hosts | API de macros de usuario | API de mapas de valores | API de mantenimiento

Métricas

Defina las métricas a monitorizar.

API de Métricas

Triggers

Configure triggers to notify you about problems in your system. Manage trigger dependencies.

Trigger API

Gráficas

Edite gráficas o métricas de gráficas separadas para una mejor presentación de los datos recopilados.

API de Gráficas | API de Métrica de Gráfica

Plantillas y grupos de plantillas

Gestione plantillas y enlácelas a equipos u otras plantillas.

API de plantilla | API de grupo de plantillas | API de mapa de valores

Descubrimiento de bajo nivel

Configure reglas de descubrimiento de bajo nivel, así como prototipos de métricas, disparadores y gráficos para monitorizar entidades dinámicas.

API de reglas LLD | API de prototipo de métrica | API de prototipo de disparador | API de prototipo de gráfico | API de prototipo de equipo

Complex event processing

Create complex event processing rules.

Complex event processing API

Correlación de eventos

Cree reglas personalizadas de correlación de eventos.

API de correlación

Descubrimiento de red

Gestione reglas de descubrimiento a nivel de red para encontrar y monitorizar automáticamente nuevos equipos. Obtenga acceso completo a la información sobre los servicios y equipos descubiertos.

API de regla de descubrimiento | API de comprobación de descubrimiento | API de equipo descubierto | API de servicio descubierto

Exportar e importar

Exporte e importe datos de configuración de Zabbix para copias de seguridad de configuración, migración o actualizaciones de configuración a gran escala.

API de configuración

Monitorización web

Configure escenarios web para monitorizar sus aplicaciones y servicios web.

API de escenario web

Alertas La API de Zabbix le permite gestionar las acciones y alertas de su sistema de monitorización.

Acciones y alertas

Defina acciones y operaciones para notificar a los usuarios sobre ciertos eventos o ejecutar automáticamente comandos remotos. Obtenga acceso a información sobre alertas generadas y sus destinatarios.

API de acción | API de alerta

Tipos de medios

Configure los tipos de medios y las múltiples formas en que los usuarios recibirán alertas.

API de tipo de medio

Scripts

Configure y ejecute scripts para ayudarle con sus tareas diarias.

API de Script

Usuarios La API de Zabbix le permite gestionar los usuarios de su sistema de monitorización.

Usuarios y grupos de usuarios

Agregue usuarios que tendrán acceso a Zabbix, asígnelos a grupos de usuarios y otorgue permisos. Cree roles para una gestión granular de los derechos de los usuarios.

API de usuario | API de grupo de usuarios | API de directorio de usuarios | API de rol de usuario

Tokens de API

Gestionar tokens de autorización.

Token API

Autenticación

Cambiar las opciones de configuración de autenticación.

API de autenticación

Dispositivos

Administre dispositivos móviles para utilizarlos con la aplicación móvil de Zabbix.

API de dispositivos

Administración Con la API de Zabbix puede cambiar la configuración de administración de su sistema de monitorización.

General

Cambie ciertas opciones de configuración global.

API de Autoregistro | API de mapa de iconos | API de imagen | API de configuración | API de expresión regular | API de módulo | API de conector

Registro de auditoría

Rastree los cambios de configuración que cada usuario ha realizado.

API de registro de auditoría

Mantenimiento

Configurar el mantenimiento.

API de mantenimiento

Proxies y grupos de proxies

Gestione los proxies utilizados en su configuración de monitorización distribuida.

API de proxy | API de grupo de proxies

Macros

Gestionar macros.

API de macro de usuario

Información de Zabbix Obtenga la versión de la API de Zabbix o la hora de la última actualización del server de Zabbix.

API info API | Server info API

Acción

Esta clase está diseñada para trabajar con acciones.

Referencias de objetos:

- Acción
- Operación de acción
  - Mensaje de operación de acción
  - Condición de operación de acción
- Operación de recuperación de acción
- Operación de actualización de acción
- Filtro de acción
  - Condición de filtro de acción

Métodos disponibles:

- action.create - crear nuevas acciones
- action.delete - eliminar acciones
- action.get - obtener acciones
- action.update - actualizar acciones

Objeto acción

Los siguientes objetos están directamente relacionados con la API de `acción`.

Acción

El objeto de acción tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
actionid	ID	ID de la acción. Comportamiento de la propiedad: - *solo lectura* - *obligatoria* para las operaciones de actualización
esc_period	string	Duración predeterminada del paso de operación. Debe ser de al menos 60 segundos. Acepta un número de segundos (por ejemplo, 60, 300), un valor con un sufijo de tiempo (por ejemplo, 30s, 1m, 2h) o una macro de usuario. Comportamiento de la propiedad: - *compatible* si `eventsource` está establecido en ”evento creado por un trigger”, ”evento interno” o ”evento creado al actualizar el estado del servicio”
eventsource	integer	Tipo de eventos que gestionará la acción. Consulte la propiedad `source` del evento para obtener una lista de los tipos de eventos compatibles. Comportamiento de la propiedad: - *constante* - *obligatoria* para las operaciones de creación
name	string	Nombre de la acción. Comportamiento de la propiedad: - *obligatoria* para las operaciones de creación
status	integer	Indica si la acción está habilitada o deshabilitada. Valores posibles: 0 - habilitada *(predeterminado)*; 1 - deshabilitada.
pause_symptoms	integer	Indica si se debe pausar la escalación cuando el evento es un evento de síntoma. Valores posibles: 0 - No pausar la escalación para problemas de síntomas; 1 - Pausar la escalación para problemas de síntomas *(predeterminado)*. Comportamiento de la propiedad: - *compatible* si `eventsource` está establecido en ”evento creado por un trigger”
pause_suppressed	integer	Indica si se debe pausar la escalación durante los períodos de mantenimiento. Valores posibles: 0 - No pausar la escalación; 1 - Pausar la escalación *(predeterminado)*. Comportamiento de la propiedad: - *compatible* si `eventsource` está establecido en ”evento creado por un trigger”

Propiedad	Tipo	Descripción
notify_if_canceled	integer	Indica si se debe enviar una notificación cuando se cancela la escalación. Valores posibles: 0 - No enviar una notificación cuando se cancela la escalación; 1 - Enviar una notificación cuando se cancela la escalación *(predeterminado)*. Comportamiento de la propiedad: - *compatible* si `eventsource` está establecido en ”evento creado por un trigger”

Operación de la acción

El objeto de operación de la acción define una operación que se realizará cuando se ejecute una acción. Tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
operationtype	integer	Tipo de operación. Valores posibles: 0 - enviar mensaje; 1 - script global; 2 - añadir host; 3 - eliminar host; 4 - añadir a un grupo de hosts; 5 - eliminar de un grupo de hosts; 6 - vincular template; 7 - desvincular template; 8 - habilitar host; 9 - deshabilitar host; 10 - establecer el modo de inventario del host; 13 - añadir etiquetas al host; 14 - eliminar etiquetas del host. Valores posibles si `eventsource` del objeto Action está establecido en ”evento creado por un trigger” o ”evento creado al actualizar el estado del servicio”: 0 - ”enviar mensaje”; 1 - ”script global”. Valores posibles si `eventsource` del objeto Action está establecido en ”evento interno”: 0 - ”enviar mensaje”. Comportamiento de la propiedad: - *obligatoria*
esc_period	string	Duración de un paso de escalado en segundos. Debe ser superior a 60 segundos. Acepta un número de segundos (por ejemplo, 60, 300), un valor con un sufijo de tiempo (por ejemplo, `30s`, `1m`, `2h`) o una macro de usuario. Si se establece en 0 o 0s, se utilizará el período de escalado predeterminado de la acción. Predeterminado: 0s. Comportamiento de la propiedad: - *compatible* si `eventsource` del objeto Action está establecido en ”evento creado por un trigger”, ”evento interno” o ”evento creado al actualizar el estado del servicio”

Propiedad	Tipo	Descripción
esc_step_from	integer	Paso desde el que se inicia el escalado. Predeterminado: 1. Comportamiento de la propiedad: - *compatible* si `eventsource` del objeto Action está establecido en ”evento creado por un trigger”, ”evento interno” o ”evento creado al actualizar el estado del servicio”
esc_step_to	integer	Paso en el que finaliza el escalado. Predeterminado: 1. Comportamiento de la propiedad: - *compatible* si `eventsource` del objeto Action está establecido en ”evento creado por un trigger”, ”evento interno” o ”evento creado al actualizar el estado del servicio”
evaltype	integer	Método de evaluación de las condiciones de la operación. Valores posibles: 0 - *(predeterminado)* And/Or; 1 - And; 2 - Or.
opcommand	object	Script global que se ejecutará. El script global debe tener definida la propiedad `scriptid`. Comportamiento de la propiedad: - *obligatoria* si `operationtype` está establecido en ”script global”
opcommand_grp	array	Grupos de hosts en los que se ejecutarán los scripts globales. Los grupos de hosts deben tener definida la propiedad `groupid`. Comportamiento de la propiedad: - *obligatoria* si `operationtype` está establecido en ”script global” y `opcommand_hst` no está establecido
opcommand_hst	array	Host en el que se ejecutarán los scripts globales. Los hosts deben tener definida la propiedad `hostid`. Comportamiento de la propiedad: - *obligatoria* si `operationtype` está establecido en ”script global” y `opcommand_grp` no está establecido
opconditions	array	Condiciones de la operación utilizadas para las acciones de trigger. El objeto de condición de la operación se describe detalladamente a continuación.
opgroup	array	Grupos de hosts a los que se añadirán los hosts. Los grupos de hosts deben tener definida la propiedad `groupid`. Comportamiento de la propiedad: - *obligatoria* si `operationtype` está establecido en ”añadir a un grupo de hosts” o ”eliminar de un grupo de hosts”
opmessage	object	Objeto que contiene los datos sobre el mensaje enviado por la operación. El objeto de mensaje de la operación se describe detalladamente a continuación. Comportamiento de la propiedad: - *obligatoria* si `operationtype` está establecido en ”enviar mensaje”

Propiedad	Tipo	Descripción
opmessage_grp	array	Grupos de usuarios a los que se enviarán los mensajes. Los grupos de usuarios deben tener definida la propiedad `usrgrpid`. Comportamiento de la propiedad: - *obligatoria* si `operationtype` está establecido en ”enviar mensaje” y `opmessage_usr` no está establecido
opmessage_usr	array	Usuarios a los que se enviarán los mensajes. Los usuarios deben tener definida la propiedad `userid`. Comportamiento de la propiedad: - *obligatoria* si `operationtype` está establecido en ”enviar mensaje” y `opmessage_grp` no está establecido
optemplate	array	Templates que se vincularán a los hosts. Los templates deben tener definida la propiedad `templateid`. Comportamiento de la propiedad: - *obligatoria* si `operationtype` está establecido en ”vincular template” o ”desvincular template”
opinventory	object	Modo de inventario que se establecerá en el host. El inventario debe tener definida la propiedad `inventory_mode`. Comportamiento de la propiedad: - *obligatoria* si `operationtype` está establecido en ”establecer el modo de inventario del host”
optag	array	Etiquetas del host que se añadirán o eliminarán. Las etiquetas deben tener definida la propiedad `tag`. La propiedad `value` es opcional. Comportamiento de la propiedad: - *compatible* si `operationtype` está establecido en ”añadir etiquetas al host” o ”eliminar etiquetas del host”.

Mensaje de operación de acción

El objeto de mensaje de operación contiene datos sobre el mensaje que será enviado por la operación. Tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
default_msg	integer	Indica si se debe usar el texto y asunto por defecto del mensaje de acción. Valores posibles: 0 - usar los datos de la operación; 1 - *(por defecto)* usar los datos del tipo de medio.
mediatypeid	ID	ID del tipo de medio que se utilizará para enviar el mensaje. Comportamiento de la propiedad: - *soportado* si `operationtype` del Objeto de operación de acción, Objeto de operación de recuperación de acción, o Objeto de operación de actualización de acción está establecido en ”enviar mensaje”, o si `operationtype` del Objeto de operación de actualización de acción está establecido en ”notificar a todos los involucrados”

Propiedad	Tipo	Descripción
message	string	Texto del mensaje de operación. Comportamiento de la propiedad: - *soportado* si `default_msg` está establecido en ”usar los datos de la operación”
subject	string	Asunto del mensaje de operación. Comportamiento de la propiedad: - *soportado* si `default_msg` está establecido en ”usar los datos de la operación”

Condición de operación de acción

El objeto de condición de operación de acción define una condición que debe cumplirse para realizar la operación actual. Tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
conditiontype	integer	Tipo de condición. Valores posibles: 14 - evento reconocido. Comportamiento de la propiedad: - *requerido*
value	string	Valor a comparar. Comportamiento de la propiedad: - *requerido*
operator	integer	Operador de condición. Valores posibles: 0 - *(por defecto)* =

Los siguientes operadores y valores son compatibles para cada tipo de condición de operación.

Condición	Nombre de la condición	Operadores soportados	Valor esperado
14	Evento reconocido	=	Si el evento está reconocido. Valores posibles: 0 - no reconocido; 1 - reconocido.

Operación de recuperación de acción

El objeto de operación de recuperación de acción define una operación que se realizará cuando se resuelva un problema. Las operaciones de recuperación son posibles **solo** para acciones de trigger, internas y de servicio. Tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
operationtype	integer	Tipo de operación. Valores posibles si `eventsource` del objeto Acción está establecido en ”evento creado por un trigger” o ”evento creado en la actualización del estado del servicio”: 0 - enviar mensaje; 1 - script global; 11 - notificar a todos los involucrados. Valores posibles si `eventsource` del objeto Acción está establecido en ”evento interno”: 0 - enviar mensaje; 11 - notificar a todos los involucrados. Comportamiento de la propiedad: - *requerido*
opcommand	object	Script global a ejecutar. El script global debe tener definida la propiedad `scriptid`. Comportamiento de la propiedad: - *requerido* si `operationtype` está establecido en ”script global”
opcommand_grp	array	Grupos de hosts en los que ejecutar scripts globales. Los grupos de hosts deben tener definida la propiedad `groupid`. Comportamiento de la propiedad: - *requerido* si `eventsource` del objeto Acción está establecido en ”evento creado por un trigger”, y `operationtype` está establecido en ”script global”, y `opcommand_hst` no está establecido
opcommand_hst	array	Host en los que ejecutar scripts globales. Los hosts deben tener definida la propiedad `hostid`. Comportamiento de la propiedad: - *requerido* si `eventsource` del objeto Acción está establecido en ”evento creado por un trigger”, y `operationtype` está establecido en ”script global”, y `opcommand_grp` no está establecido
opmessage	object	Objeto que contiene los datos sobre el mensaje enviado por la operación de recuperación. El objeto de mensaje de operación se describe en detalle arriba. Comportamiento de la propiedad: - *requerido* si `operationtype` está establecido en ”enviar mensaje”
opmessage_grp	array	Grupos de usuarios a los que enviar mensajes. Los grupos de usuarios deben tener definida la propiedad `usrgrpid`. Comportamiento de la propiedad: - *requerido* si `operationtype` está establecido en ”enviar mensaje” y `opmessage_usr` no está establecido
opmessage_usr	array	Usuarios a los que enviar mensajes. Los usuarios deben tener definida la propiedad `userid`. Comportamiento de la propiedad: - *requerido* si `operationtype` está establecido en ”enviar mensaje” y `opmessage_grp` no está establecido

Operación de actualización de acción

El objeto de operación de actualización de acción define una operación que se realizará cuando se actualice un problema (se comente, se reconozca, se cambie la severidad o se cierre manualmente). Las operaciones de actualización son posibles **solo** para acciones de trigger y de servicio. Tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
operationtype	integer	Tipo de operación. Valores posibles: 0 - enviar mensaje; 1 - script global; 12 - notificar a todos los involucrados. Comportamiento de la propiedad: - *requerido*
opcommand	object	Script global a ejecutar. El script global debe tener definida la propiedad `scriptid`. Comportamiento de la propiedad: - *requerido* si `operationtype` está establecido en ”script global”
opcommand_grp	array	Grupos de hosts en los que ejecutar scripts globales. Los grupos de hosts deben tener definida la propiedad `groupid`. Comportamiento de la propiedad: - *requerido* si `eventsource` del objeto Acción está establecido en ”evento creado por un trigger”, y `operationtype` está establecido en ”script global”, y `opcommand_hst` no está establecido
opcommand_hst	array	Host en los que ejecutar scripts globales. Los hosts deben tener definida la propiedad `hostid`. Comportamiento de la propiedad: - *requerido* si `eventsource` del objeto Acción está establecido en ”evento creado por un trigger”, y `operationtype` está establecido en ”script global”, y `opcommand_grp` no está establecido
opmessage	object	Objeto que contiene los datos sobre el mensaje enviado por la operación de actualización. El objeto mensaje de operación se describe en detalle arriba.
opmessage_grp	array	Grupos de usuarios a los que enviar mensajes. Los grupos de usuarios deben tener definida la propiedad `usrgrpid`. Comportamiento de la propiedad: - *requerido* si `operationtype` está establecido en ”enviar mensaje” y `opmessage_usr` no está establecido
opmessage_usr	array	Usuarios a los que enviar mensajes. Los usuarios deben tener definida la propiedad `userid`. Comportamiento de la propiedad: - *requerido* si `operationtype` está establecido en ”enviar mensaje” y `opmessage_grp` no está establecido

Filtro de acción

El objeto filtro de acción define un conjunto de condiciones que deben cumplirse para realizar las operaciones de acción configuradas. Tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
conditions	array	Conjunto de condiciones de filtro a utilizar para filtrar los resultados. Las condiciones se ordenarán en el orden de su colocación en la fórmula. Comportamiento de la propiedad: - *obligatorio*
evaltype	integer	Método de evaluación de la condición del filtro. Valores posibles: 0 - Y/O; 1 - Y; 2 - O; 3 - Expresión personalizada. Comportamiento de la propiedad: - *obligatorio*
eval_formula	string	Expresión generada que se utilizará para evaluar las condiciones del filtro. La expresión contiene IDs que hacen referencia a condiciones de filtro específicas por su `formulaid`. El valor de `eval_formula` es igual al valor de `formula` para los filtros con una expresión personalizada. Comportamiento de la propiedad: - *solo lectura*
formula	string	Expresión definida por el usuario que se utilizará para evaluar las condiciones de los filtros con una expresión personalizada. La expresión debe contener IDs que hagan referencia a condiciones de filtro específicas por su `formulaid`. Los IDs utilizados en la expresión deben coincidir exactamente con los definidos en las condiciones del filtro: ninguna condición puede quedar sin usar u omitida. Comportamiento de la propiedad: - *obligatorio* si `evaltype` está establecido en ”expresión personalizada”

Condición del filtro de la acción

El objeto de condición del filtro de la acción define una condición específica que debe comprobarse antes de ejecutar las operaciones de la acción.

Property	Type	Description
conditiontype	integer	Tipo de condición. Valores posibles si `eventsource` del Action object está establecido en "evento creado por un trigger": 0 - grupo de hosts; 1 - host; 2 - trigger; 3 - nombre del evento; 4 - gravedad del trigger; 6 - período de tiempo; 13 - template de host; 16 - el problema está suprimido; 25 - etiqueta del evento; 26 - valor de la etiqueta del evento. Valores posibles si `eventsource` del Action object está establecido en "evento creado por una regla de descubrimiento": 7 - IP del host; 8 - tipo de servicio descubierto; 9 - puerto del servicio descubierto; 10 - estado del descubrimiento; 11 - duración del tiempo de actividad o inactividad; 12 - valor recibido; 18 - regla de descubrimiento; 19 - comprobación de descubrimiento; 20 - proxy; 21 - objeto de descubrimiento; 29 - grupo de proxies. Valores posibles si `eventsource` del Action object está establecido en "evento creado por el registro automático de un agent activo": 20 - proxy; 22 - nombre del host; 24 - metadatos del host; 29 - grupo de proxies. Valores posibles si `eventsource` del Action object está establecido en "evento interno": 0 - grupo de hosts; 1 - host; 13 - template de host; 23 - tipo de evento; 25 - etiqueta del evento; 26 - valor de la etiqueta del evento. Valores posibles si `eventsource` del Action object está establecido en "evento creado al actualizar el estado del servicio": 25 - etiqueta del evento; 26 - valor de la etiqueta del evento; 27 - servicio; 28 - nombre del servicio. Property behavior: - *required*
value	string	Valor con el que se debe comparar. Property behavior: - *required*

Property	Type	Description
value2	string	Valor secundario con el que se debe comparar. Property behavior: - *required* si `eventsource` del Action object está establecido en "evento creado por un trigger", `conditiontype` está establecido en cualquier valor posible para las acciones de trigger y el tipo de condición (véase más abajo) es "26" - *required* si `eventsource` del Action object está establecido en "evento interno", `conditiontype` está establecido en cualquier valor posible para las acciones internas y el tipo de condición (véase más abajo) es "26" - *required* si `eventsource` del Action object está establecido en "evento creado al actualizar el estado del servicio", `conditiontype` está establecido en cualquier valor posible para las acciones de servicio y el tipo de condición (véase más abajo) es "26"
formulaid	string	ID único arbitrario que se utiliza para hacer referencia a la condición desde una expresión personalizada. Solo puede contener letras mayúsculas. El usuario debe definir el ID al modificar las condiciones del filtro, pero se generará uno nuevo al solicitarlas posteriormente.
operator	integer	Operador de la condición. Valores posibles: 0 - *(predeterminado)* es igual a; 1 - no es igual a; 2 - contiene; 3 - no contiene; 4 - está en; 5 - es mayor o igual que; 6 - es menor o igual que; 7 - no está en; 8 - coincide; 9 - no coincide; 10 - Sí; 11 - No.

Note:
Para comprender mejor cómo utilizar filtros con distintos tipos de expresiones, consulte los ejemplos de las páginas de los métodos action.get y action.create.

Los siguientes operadores y valores son compatibles con cada tipo de condición.

Condition	Condition name	Supported operators	Expected value
0	Grupo de hosts	es igual a, no es igual a	ID del grupo de hosts.
1	Host	es igual a, no es igual a	ID del host.
2	Trigger	es igual a, no es igual a	ID del trigger.
3	Nombre del evento	contiene, no contiene	Nombre del evento.
4	Gravedad del trigger	es igual a, no es igual a, es mayor o igual que, es menor o igual que	Gravedad del trigger. Consulte la propiedad `severity` del trigger para obtener una lista de las gravedades de trigger compatibles.
5	Valor del trigger	es igual a	Valor del trigger. Consulte la propiedad `value` del trigger para obtener una lista de los valores de trigger compatibles.

Condition	Condition name	Supported operators	Expected value
6	Período de tiempo	está en, no está en	Momento en que se activó el evento, expresado como un período de tiempo.
7	IP del host	es igual a, no es igual a	Uno o varios rangos de IP que se comprobarán, separados por comas. Consulte la sección configuración del descubrimiento de red para obtener más información sobre los formatos compatibles de los rangos de IP.
8	Tipo de servicio descubierto	es igual a, no es igual a	Tipo de servicio descubierto. El tipo de servicio coincide con el tipo de comprobación de descubrimiento utilizada para detectar el servicio. Consulte la propiedad `type` de la comprobación de descubrimiento para obtener una lista de los tipos compatibles.
9	Puerto del servicio descubierto	es igual a, no es igual a	Uno o varios rangos de puertos, separados por comas.
10	Estado del descubrimiento	es igual a	Estado de un objeto descubierto. Valores posibles: 0 - host o servicio activo; 1 - host o servicio inactivo; 2 - host o servicio descubierto; 3 - host o servicio perdido.
11	Duración del tiempo de actividad o inactividad	es mayor o igual que, es menor o igual que	Tiempo, en segundos, que indica cuánto tiempo ha permanecido el objeto descubierto en el estado actual.
12	Valores recibidos	es igual a, no es igual a, es mayor o igual que, es menor o igual que, contiene, no contiene	Valor devuelto al realizar una comprobación de descubrimiento de Zabbix agent, SNMPv1, SNMPv2 o SNMPv3.
13	Template de host	es igual a, no es igual a	ID del template vinculado.
16	El problema está suprimido	Sí, No	No se requiere ningún valor: utilizar el operador ”Sí” significa que el problema debe estar suprimido; ”No” significa que no debe estar suprimido.
18	Regla de descubrimiento	es igual a, no es igual a	ID de la regla de descubrimiento.
19	Comprobación de descubrimiento	es igual a, no es igual a	ID de la comprobación de descubrimiento.
20	Proxy	es igual a, no es igual a	ID del proxy.
21	Objeto de descubrimiento	es igual a	Tipo de objeto que activó el evento de descubrimiento. Valores posibles: 1 - host descubierto; 2 - servicio descubierto.
22	Nombre del host	contiene, no contiene, coincide, no coincide	Nombre del host. El uso de una expresión regular es compatible con los operadores *coincide* y *no coincide* en las condiciones de registro automático.
23	Tipo de evento	es igual a	Evento interno específico. Valores posibles: 0 - item en estado ”no compatible”; 1 - item en estado ”normal”; 2 - regla LLD en estado ”no compatible”; 3 - regla LLD en estado ”normal”; 4 - trigger en estado ”desconocido”; 5 - trigger en estado ”normal”.

Condition	Condition name	Supported operators	Expected value
24	Metadatos del host	contiene, no contiene, coincide, no coincide	Metadatos del host registrado automáticamente. El uso de una expresión regular es compatible con los operadores *coincide* y *no coincide*.
25	Etiqueta	es igual a, no es igual a, contiene, no contiene	Etiqueta del evento.
26	Valor de la etiqueta	es igual a, no es igual a, contiene, no contiene	Valor de la etiqueta del evento.
27	Servicio	es igual a, no es igual a	ID del servicio.
28	Nombre del servicio	es igual a, no es igual a	Nombre del servicio.
29	Grupo de proxies	es igual a, no es igual a	ID del grupo de proxies.

action.create

Descripción

`object action.create(object/array actions)`

Este método le permite crear nuevas acciones.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object/array)` Acciones a crear.

Además de las propiedades estándar de la acción, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
filter	object	Objeto filtro de acción para la acción.
operations	array	Operaciones de acción a crear para la acción.
recovery_operations	array	Operaciones de recuperación de acción a crear para la acción.
update_operations	array	Operaciones de actualización de acción a crear para la acción.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las acciones creadas bajo la propiedad `actionids`. El orden de los IDs devueltos coincide con el orden de las acciones pasadas.

Ejemplos

Crear una acción de disparador

Cree una acción de disparador que comenzará una vez que un disparador (con la palabra ”memory” en su nombre) del equipo ”10084” entre en estado de PROBLEMA. La acción tendrá 4 operaciones configuradas. La primera operación, inmediata, enviará un mensaje a todos los usuarios del grupo de usuarios ”7” a través del tipo de medio ”1”. Si el evento no se resuelve en 30 minutos, la segunda operación ejecutará el script ”5” (script con alcance ”Operación de acción”) en todos los equipos del grupo ”2”. Si el evento se resuelve, una operación de recuperación notificará a todos los usuarios que recibieron algún mensaje relacionado con el problema. Si el evento se actualiza, una operación de reconocimiento/actualización notificará (con un asunto y mensaje personalizados) a todos los usuarios que recibieron algún mensaje relacionado con el problema.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "action.create",
    "params": {
        "name": "Trigger action",
        "eventsource": 0,
        "esc_period": "30m",
        "filter": {
            "evaltype": 0,
            "conditions": [
                {
                    "conditiontype": 1,
                    "operator": 0,
                    "value": "10084"
                },
                {
                    "conditiontype": 3,
                    "operator": 2,
                    "value": "memory"
                }
            ]
        },
        "operations": [
            {
                "operationtype": 0,
                "esc_step_from": 1,
                "esc_step_to": 1,
                "opmessage_grp": [
                    {
                        "usrgrpid": "7"
                    }
                ],
                "opmessage": {
                    "default_msg": 1,
                    "mediatypeid": "1"
                }
            },
            {
                "operationtype": 1,
                "esc_step_from": 2,
                "esc_step_to": 2,
                "opconditions": [
                    {
                        "conditiontype": 14,
                        "operator": 0,
                        "value": "0"
                    }
                ],
                "opcommand_grp": [
                    {
                        "groupid": "2"
                    }
                ],
                "opcommand": {
                    "scriptid": "5"
                }
            }
        ],
        "recovery_operations": [
            {
                "operationtype": "11",
                "opmessage": {
```

```
                    "default_msg": 1
                }
            }
        ],
        "update_operations": [
            {
                "operationtype": "12",
                "opmessage": {
                    "default_msg": 0,
                    "message": "Custom update operation message body",
                    "subject": "Custom update operation message subject"
                }
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "actionids": [
            "17"
        ]
    },
    "id": 1
}
```

Crear una acción de descubrimiento

Cree una acción de descubrimiento que vincule la plantilla ”10001” a los equipos descubiertos.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "action.create",
    "params": {
        "name": "Discovery action",
        "eventsource": 1,
        "filter": {
            "evaltype": 0,
            "conditions": [
                {
                    "conditiontype": 21,
                    "operator": 0,
                    "value": "1"
                },
                {
                    "conditiontype": 10,
                    "operator": 0,
                    "value": "2"
                }
            ]
        },
        "operations": [
            {
                "operationtype": 6,
                "optemplate": [
                    {
                        "templateid": "10001"
                    }
                ]
```

```
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "actionids": [
            "18"
        ]
    },
    "id": 1
}
```

Uso de un filtro de expresión personalizada

Cree una acción de disparador que utilice una expresión personalizada - ”A and (B or C)” - para evaluar las condiciones de la acción. Una vez que un disparador con una severidad mayor o igual a ”Warning” del equipo ”10084” o del equipo ”10106” pase a estado de PROBLEMA, la acción enviará un mensaje a todos los usuarios del grupo de usuarios ”7” mediante el tipo de medio ”1”. Los IDs de fórmula ”A”, ”B” y ”C” se han elegido arbitrariamente.

Petición:

```
{
    "jsonrpc": "2.0",
    "method": "action.create",
    "params": {
        "name": "Trigger action",
        "eventsource": 0,
        "esc_period": "15m",
        "filter": {
            "evaltype": 3,
            "formula": "A and (B or C)",
            "conditions": [
                {
                    "conditiontype": 4,
                    "operator": 5,
                    "value": "2",
                    "formulaid": "A"
                },
                {
                    "conditiontype": 1,
                    "operator": 0,
                    "value": "10084",
                    "formulaid": "B"
                },
                {
                    "conditiontype": 1,
                    "operator": 0,
                    "value": "10106",
                    "formulaid": "C"
                }
            ]
        },
        "operations": [
            {
                "operationtype": 0,
                "esc_step_from": 1,
                "esc_step_to": 1,
                "opmessage_grp": [
                    {
```

```
                        "usrgrpid": "7"
                    }
                ],
                "opmessage": {
                    "default_msg": 1,
                    "mediatypeid": "1"
                }
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "actionids": [
            "18"
        ]
    },
    "id": 1
}
```

Crear regla de autorregistro de agente

Cree una acción de autorregistro que agregue un equipo al grupo de equipos ”2” cuando el nombre del equipo contenga ”SRV” o los metadatos contengan ”AlmaLinux”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "action.create",
    "params": {
        "name": "Register Linux servers",
        "eventsource": "2",
        "filter": {
            "evaltype": "2",
            "conditions": [
                {
                    "conditiontype": "22",
                    "operator": "2",
                    "value": "SRV"
                },
                {
                    "conditiontype": "24",
                    "operator": "2",
                    "value": "AlmaLinux"
                }
            ]
        },
        "operations": [
            {
                "operationtype": "4",
                "opgroup": [
                    {
                        "groupid": "2"
                    }
                ]
            }
        ]
    },
    "id": 1
```

```
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "actionids": [
            19
        ]
    },
    "id": 1
}
```

Crear una regla de autorregistro de agente con etiquetas de equipo

Cree una acción de autorregistro que agregue un equipo al grupo de equipos ”2” y agregue dos etiquetas de equipo.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "action.create",
    "params": {
        "name": "Registrar servidores Linux con etiquetas",
        "eventsource": "2",
        "operations": [
            {
                "operationtype": "4",
                "opgroup": [
                    {
                        "groupid": "2"
                    }
                ]
            },
            {
                "operationtype": "13",
                "optag": [
                    {
                        "tag": "location",
                        "value": "office"
                    },
                    {
                        "tag": "city",
                        "value": "Riga"
                    }
                ]
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "actionids": [
            20
        ]
    },
    "id": 1
}
```

Ver también

- Filtro de acción
- Operación de acción
- Script

Origen

CAction::create() en *ui/include/classes/api/services/CAction.php*.

action.delete

Descripción

`object action.delete(array actionIds)`

Este método permite eliminar acciones.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.
 El usuario no podrá eliminar una acción que contenga un proxy o grupo de proxies inaccesible en sus condiciones de filtro. Consulte la configuración del grupo de usuarios (*Lista de acceso a proxies*) para obtener más información.

Parámetros

`(array)` IDs de las acciones a eliminar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las acciones eliminadas bajo la propiedad `actionids`.

Ejemplos

Eliminar varias acciones

Eliminar dos acciones.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "action.delete",
    "params": [
        "17",
        "18"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "actionids": [
            "17",
            "18"
        ]
    },
    "id": 1
}
```

Fuente

CAction::delete() en *ui/include/classes/api/services/CAction.php*.

action.get

Descripción

`integer/array action.get(object parameters)`

El método permite recuperar acciones según los parámetros proporcionados.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
actionids	ID/array	Devolver solo las acciones con los IDs dados.
groupids	ID/array	Devolver solo las acciones que utilizan los grupos de hosts dados en las condiciones de la acción.
hostids	ID/array	Devolver solo las acciones que utilizan los hosts dados en las condiciones de la acción.
triggerids	ID/array	Devolver solo las acciones que utilizan los triggers dados en las condiciones de la acción.
mediatypeids	ID/array	Devolver solo las acciones que utilizan los tipos de medios dados para enviar mensajes.
usrgrpids	ID/array	Devolver solo las acciones que están configuradas para enviar mensajes a los grupos de usuarios dados.
userids	ID/array	Devolver solo las acciones que están configuradas para enviar mensajes a los usuarios dados.
scriptids	ID/array	Devolver solo las acciones que están configuradas para ejecutar los scripts dados.
selectFilter	query	Devolver una propiedad `filter` con el filtro de condición de la acción.
selectOperations	query	Devolver una propiedad `operations` con las operaciones de la acción.
selectRecoveryOperations	query	Devolver una propiedad `recovery_operations` con las operaciones de recuperación de la acción.
selectUpdateOperations	query	Devolver una propiedad `update_operations` con las operaciones de actualización de la acción.
sortfield	string/array	Ordenar el resultado por las propiedades dadas. Valores posibles: `actionid`, `name`, `status`.
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperar acciones de trigger

Recuperar todas las acciones de trigger configuradas junto con las condiciones de acción y operaciones.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "action.get",
    "params": {
        "output": "extend",
        "selectOperations": "extend",
        "selectRecoveryOperations": "extend",
        "selectUpdateOperations": "extend",
        "selectFilter": "extend",
        "filter": {
            "eventsource": 0
        }
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "actionid": "3",
            "name": "Report problems to Zabbix administrators",
            "eventsource": "0",
            "status": "1",
            "esc_period": "1h",
            "pause_suppressed": "1",
            "filter": {
                "evaltype": "0",
                "formula": "",
                "conditions": [],
                "eval_formula": ""
            },
            "operations": [
                {
                    "operationid": "3",
                    "actionid": "3",
                    "operationtype": "0",
                    "esc_period": "0",
                    "esc_step_from": "1",
                    "esc_step_to": "1",
                    "evaltype": "0",
                    "opconditions": [],
                    "opmessage": [
                        {
                            "default_msg": "1",
                            "subject": "",
                            "message": "",
                            "mediatypeid" => "0"
                        }
                    ],
                    "opmessage_grp": [
                        {
                            "usrgrpid": "7"
                        }
                    ]
                }
            ],
            "recovery_operations": [
                {
                    "operationid": "7",
                    "actionid": "3",
```

```
                    "operationtype": "11",
                    "evaltype": "0",
                    "opconditions": [],
                    "opmessage": {
                        "default_msg": "0",
                        "subject": "{TRIGGER.STATUS}: {TRIGGER.NAME}",
                        "message": "Trigger: {TRIGGER.NAME}\r\nTrigger state: {TRIGGER.STATUS}\r\nTrigger
                        "mediatypeid": "0"
                    }
                }
            ],
            "update_operations": [
                {
                    "operationid": "31",
                    "operationtype": "12",
                    "evaltype": "0",
                    "opmessage": {
                        "default_msg": "1",
                        "subject": "",
                        "message": "",
                        "mediatypeid": "0"
                    }
                },
                {
                    "operationid": "32",
                    "operationtype": "0",
                    "evaltype": "0",
                    "opmessage": {
                        "default_msg": "0",
                        "subject": "Updated: {TRIGGER.NAME}",
                        "message": "{USER.FULLNAME} updated problem at {EVENT.UPDATE.DATE} {EVENT.UPDATE.T
                        "mediatypeid": "1"
                    },
                    "opmessage_grp": [
                        {
                            "usrgrpid": "7"
                        }
                    ],
                    "opmessage_usr": []
                },
                {
                    "operationid": "33",
                    "operationtype": "1",
                    "evaltype": "0",
                    "opcommand": {
                        "scriptid": "3"
                    },
                    "opcommand_hst": [
                        {
                            "hostid": "10084"
                        }
                    ],
                    "opcommand_grp": []
                }
            ]
        }
    ],
    "id": 1
}
```


Recuperar acciones de descubrimiento

Recupera todas las acciones de descubrimiento configuradas junto con las condiciones y operaciones de la acción. El filtro utiliza

el tipo de evaluación "and", por lo que la propiedad `formula` está vacía y `eval_formula` se genera automáticamente.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "action.get",
    "params": {
        "output": "extend",
        "selectOperations": "extend",
        "selectFilter": "extend",
        "filter": {
            "eventsource": 1
        }
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "actionid": "2",
            "name": "Auto discovery. Linux servers.",
            "eventsource": "1",
            "status": "1",
            "esc_period": "0s",
            "pause_suppressed": "1",
            "filter": {
                "evaltype": "0",
                "formula": "",
                "conditions": [
                    {
                        "conditiontype": "10",
                        "operator": "0",
                        "value": "0",
                        "value2": "",
                        "formulaid": "B"
                    },
                    {
                        "conditiontype": "8",
                        "operator": "0",
                        "value": "9",
                        "value2": "",
                        "formulaid": "C"
                    },
                    {
                        "conditiontype": "12",
                        "operator": "2",
                        "value": "Linux",
                        "value2": "",
                        "formulaid": "A"
                    }
                ],
                "eval_formula": "A and B and C"
            },
            "operations": [
                {
                    "operationid": "1",
                    "actionid": "2",
                    "operationtype": "6",
                    "esc_period": "0s",
                    "esc_step_from": "1",
```

```
                    "esc_step_to": "1",
                    "evaltype": "0",
                    "opconditions": [],
                    "optemplate": [
                        {
                            "templateid": "10001"
                        }
                    ]
                },
                {
                    "operationid": "2",
                    "actionid": "2",
                    "operationtype": "4",
                    "esc_period": "0s",
                    "esc_step_from": "1",
                    "esc_step_to": "1",
                    "evaltype": "0",
                    "opconditions": [],
                    "opgroup": [
                        {
                            "groupid": "2"
                        }
                    ]
                }
            ]
        }
    ],
    "id": 1
}
```


Ver también

- Filtro de acción
- Operación de acción

Fuente

CAction::get() en *ui/include/classes/api/services/CAction.php*.

action.update

Descripción

`object action.update(object/array actions)`

Este método permite actualizar acciones existentes.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object/array)` Propiedades de la acción a actualizar.

La propiedad `actionid` debe estar definida para cada acción, todas las demás propiedades son opcionales. Sólo se actualizarán las propiedades pasadas, todas las demás permanecerán sin cambios.

Además de las propiedades estándar de la acción, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
filter	object	Objeto filtro de acción para reemplazar el filtro actual.
operations	array	Operaciones de acción para reemplazar las operaciones existentes.

Parámetro	Tipo	Descripción
recovery_operations	array	Operaciones de recuperación de la acción para reemplazar las operaciones de recuperación existentes. Comportamiento del parámetro: - *soportado* si `eventsource` del objeto Acción está establecido en "evento creado por un trigger", "evento interno" o "evento creado en la actualización del estado del servicio"
update_operations	array	Operaciones de actualización de la acción para reemplazar las operaciones de actualización existentes. Comportamiento del parámetro: - *soportado* si `eventsource` del objeto Acción está establecido en "evento creado por un trigger" o "evento creado en la actualización del estado del servicio"

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las acciones actualizadas bajo la propiedad `actionids`.

Ejemplos

Deshabilitar acción

Deshabilitar una acción, es decir, establecer su estado en "1".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "action.update",
    "params": {
        "actionid": "2",
        "status": "1"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "actionids": [
            "2"
        ]
    },
    "id": 1
}
```

Ver también

- Filtro de acción
- Operación de acción

Fuente

CAction::update() en *ui/include/classes/api/services/CAction.php*.

Alerta

Esta clase está diseñada para trabajar con alertas.

Referencias de objetos:

- Alerta

Métodos disponibles:

- alert.get - recuperar alertas

Objeto de alerta

Los siguientes objetos están directamente relacionados con la API de `alerta`.

Alerta

> **Note:**
> Las alertas son creadas por el servidor Zabbix y no pueden modificarse a través de la API.

El objeto de alerta contiene información sobre si ciertas operaciones de acción se han ejecutado correctamente. Tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
alertid	ID	ID de la alerta.
actionid	ID	ID de la acción que generó la alerta.
alerttype	integer	Tipo de alerta. Valores posibles: 0 - mensaje; 1 - comando remoto.
clock	timestamp	Hora en la que se generó la alerta.
error	string	Texto de error si hay problemas al enviar un mensaje o ejecutar un comando.
esc_step	integer	Paso de escalado de la acción durante el cual se generó la alerta.
eventid	ID	ID del evento que activó la acción.
mediatypeid	ID	ID del tipo de medio que se utilizó para enviar el mensaje.
message	text	Texto del mensaje. Comportamiento de la propiedad: - *soportado* si `alerttype` está establecido en ”mensaje”
retries	integer	Número de veces que Zabbix intentó enviar el mensaje.
sendto	string	Dirección, nombre de usuario u otro identificador del destinatario. Comportamiento de la propiedad: - *soportado* si `alerttype` está establecido en ”mensaje”
status	integer	Estado que indica si la operación de acción se ha ejecutado correctamente. Valores posibles si `alerttype` está establecido en ”mensaje”: 0 - mensaje no enviado; 1 - mensaje enviado; 2 - falló después de varios intentos; 3 - la nueva alerta aún no ha sido procesada por el gestor de alertas. Valores posibles si `alerttype` está establecido en ”comando remoto”: 0 - comando no ejecutado; 1 - comando ejecutado; 2 - se intentó ejecutar el comando en el agent de Zabbix, pero no estaba disponible.
subject	string	Asunto del mensaje. Comportamiento de la propiedad: - *soportado* si `alerttype` está establecido en ”mensaje”
userid	ID	ID del usuario al que se envió el mensaje.
p_eventid	ID	ID del evento de problema que generó la alerta.
acknowledgeid	ID	ID del reconocimiento que generó la alerta.

alert.get

Descripción

`integer/array alert.get(object parameters)`

El método permite recuperar alertas según los parámetros proporcionados.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
alertids	ID/array	Devolver solo las alertas con los IDs dados.
actionids	ID/array	Devolver solo las alertas generadas por las acciones dadas.
eventids	ID/array	Devolver solo las alertas generadas por los eventos dados.
groupids	ID/array	Devolver solo las alertas generadas por objetos de los grupos de host dados.
hostids	ID/array	Devolver solo las alertas generadas por objetos de los hosts dados.
mediatypeids	ID/array	Devolver solo las alertas de mensaje que utilizaron los tipos de medio dados.
objectids	ID/array	Devolver solo las alertas generadas por los objetos dados
userids	ID/array	Devolver solo las alertas de mensaje que se enviaron a los usuarios dados.
eventobject	integer	Devolver solo las alertas generadas por eventos relacionados con objetos del tipo dado. Consulte el evento `object` para ver una lista de los tipos de objeto soportados. Por defecto: 0 - trigger.
eventsource	integer	Devolver solo las alertas generadas por eventos del tipo dado. Consulte el evento `source` para ver una lista de los tipos de evento soportados. Por defecto: 0 - eventos de trigger.
time_from	timestamp	Devolver solo las alertas que se hayan generado después del tiempo dado.
time_till	timestamp	Devolver solo las alertas que se hayan generado antes del tiempo dado.
selectHosts	query	Devolver una propiedad `hosts` con los datos de los hosts que activaron la operación de acción.
selectMediatypes	query	Devolver una propiedad `mediatypes` con un array de los tipos de medio que se utilizaron para la alerta de mensaje. Consulte `mediatype.get` para ver las restricciones basadas en el tipo de usuario.
selectUsers	query	Devolver una propiedad `users` con un array de los usuarios a los que se dirigió el mensaje. Consulte `user.get` para ver las restricciones basadas en el tipo de usuario.
sortfield	string/array	Ordenar el resultado por las propiedades dadas. Valores posibles: `alertid`, `clock`, `eventid`, `mediatypeid`, `sendto`, `status`.
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
editable	boolean	
excludeSearch	boolean	

Parámetro	Tipo	Descripción
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperar alertas por ID de acción

Recuperar todas las alertas generadas por la acción "3".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "alert.get",
    "params": {
        "output": "extend",
        "actionids": "3"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "alertid": "1",
            "actionid": "3",
            "eventid": "21243",
            "userid": "1",
            "clock": "1362128008",
            "mediatypeid": "1",
            "sendto": "support@company.com",
            "subject": "PROBLEM: Zabbix agent on Linux server is unreachable for 5 minutes: ",
            "message": "Trigger: Zabbix agent on Linux server is unreachable for 5 minutes: \nTrigger stat
            "status": "0",
            "retries": "3",
            "error": "",
            "esc_step": "1",
            "alerttype": "0",
            "p_eventid": "0",
            "acknowledgeid": "0"
        }
    ],
    "id": 1
}
```

Ver también

- Host
- Tipo de medio

- Usuario

Fuente

CAlert::get() en *ui/include/classes/api/services/CAlert.php*.

Autenticación

Esta clase está diseñada para trabajar con la configuración de autenticación.

Referencias de objetos:

- Autenticación

Métodos disponibles:

- authentication.get - recuperar la configuración de autenticación
- authentication.update - actualizar la configuración de autenticación

Objeto de autenticación

Los siguientes objetos están directamente relacionados con la API de `authentication`.

Autenticación

El objeto de autenticación tiene las siguientes propiedades.

Property	Type	Description
authentication_type	integer	Autenticación predeterminada. Valores posibles: 0 - *(predeterminado)* Interna; 1 - LDAP.
http_auth_enabled	integer	Autenticación HTTP. Valores posibles: 0 - *(predeterminado)* Deshabilitada; 1 - Habilitada. Comportamiento de la propiedad: - *compatible* si `$ZBX_FEATURE_FLAGS['http_auth_enabled']` está establecido en `true` en el archivo de configuración del frontend.
http_login_form	integer	Formulario de inicio de sesión predeterminado. Valores posibles: 0 - *(predeterminado)* Formulario de inicio de sesión de Zabbix; 1 - Formulario de inicio de sesión HTTP. Comportamiento de la propiedad: - *compatible* si `$ZBX_FEATURE_FLAGS['http_auth_enabled']` está establecido en `true` en el archivo de configuración del frontend.
http_strip_domains	string	Nombre de dominio que se eliminará. Comportamiento de la propiedad: - *compatible* si `$ZBX_FEATURE_FLAGS['http_auth_enabled']` está establecido en `true` en el archivo de configuración del frontend.
http_case_sensitive	integer	Inicio de sesión HTTP con distinción entre mayúsculas y minúsculas. Valores posibles: 0 - Desactivado; 1 - *(predeterminado)* Activado. Comportamiento de la propiedad: - *compatible* si `$ZBX_FEATURE_FLAGS['http_auth_enabled']` está establecido en `true` en el archivo de configuración del frontend.

Property	Type	Description
ldap_auth_enabled	integer	Autenticación LDAP. Valores posibles: 0 - *(predeterminado)* Deshabilitada; 1 - Habilitada.
ldap_case_sensitive	integer	Inicio de sesión LDAP con distinción entre mayúsculas y minúsculas. Valores posibles: 0 - Desactivado; 1 - *(predeterminado)* Activado.
ldap_userdirectoryid	ID	ID del directorio de usuarios predeterminado para la autenticación LDAP. Se utiliza para los grupos de usuarios con `gui_access` establecido en LDAP o Predeterminado del sistema. Comportamiento de la propiedad: - *obligatorio* si `ldap_auth_enabled` está establecido en ”Habilitada”
saml_auth_enabled	integer	Autenticación SAML. Valores posibles: 0 - *(predeterminado)* Deshabilitada; 1 - Habilitada.
saml_case_sensitive	integer	Inicio de sesión SAML con distinción entre mayúsculas y minúsculas. Valores posibles: 0 - Desactivado; 1 - *(predeterminado)* Activado.
passwd_min_length	integer	Requisito de longitud mínima de la contraseña. El rango de valores posibles es de 1 a 70. Predeterminado: 8.
passwd_check_rules	integer	Reglas de comprobación de contraseñas. Valores de mapa de bits posibles: 0 - Comprobar la longitud de la contraseña; 1 - Comprobar si la contraseña utiliza letras latinas mayúsculas y minúsculas; 2 - Comprobar si la contraseña utiliza dígitos; 4 - Comprobar si la contraseña utiliza caracteres especiales; 8 - *(predeterminado)* Comprobar que la contraseña no esté en la lista de contraseñas de uso común y no contenga derivaciones de la palabra ”Zabbix” ni del nombre, apellido o nombre de usuario del usuario. Este es un campo de máscara de bits; se acepta cualquier suma de los valores de mapa de bits posibles (por ejemplo, 15 para comprobar todas las reglas).
ldap_jit_status	integer	Estado del aprovisionamiento LDAP. Valores posibles: 0 - Deshabilitado para los IdP LDAP configurados; 1 - Habilitado para los IdP LDAP configurados.
saml_jit_status	integer	Estado del aprovisionamiento SAML. Valores posibles: 0 - Deshabilitado para los IdP SAML configurados; 1 - Habilitado para los IdP SAML configurados.

Property	Type	Description
jit_provision_interval	string	Intervalo de tiempo entre las solicitudes de aprovisionamiento JIT para el usuario que ha iniciado sesión. Acepta un número de segundos (por ejemplo, 3600) o un valor con un sufijo de tiempo (por ejemplo, 3600s, 60m, 1h, 1d, 1M, 1y), incluido el uso de meses y años. Valor mínimo: 1h. Predeterminado: 1h. Disponible únicamente para el aprovisionamiento LDAP.
disabled_usrgrpid	ID	ID del grupo de usuarios al que se asignará el usuario cuyo aprovisionamiento se ha cancelado. El grupo de usuarios debe estar deshabilitado y no se puede habilitar ni eliminar mientras esté configurado. Comportamiento de la propiedad: - *obligatorio* si `ldap_jit_status` está establecido en "Habilitado para los IdP LDAP configurados" o `saml_jit_status` está establecido en "Habilitado para los IdP SAML configurados"
mfa_status	integer	Autenticación multifactor. Valores posibles: 0 - Deshabilitada (para todos los métodos MFA configurados); 1 - Habilitada (para todos los métodos MFA configurados).
mfaid	ID	Método MFA predeterminado para los grupos de usuarios con MFA habilitada. Comportamiento de la propiedad: - *obligatorio* si `mfa_status` está establecido en "Habilitada"

authentication.get

Descripción

`object authentication.get(object parameters)`

El método permite recuperar el objeto de autenticación según los parámetros proporcionados.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración del rol de usuario. Consulte User roles para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método solo admite un parámetro.

Parámetro	Tipo	Descripción
output	query	Este parámetro se describe en el comentario de referencia.

Valores de retorno

`(object)` Devuelve el objeto de autenticación.

Ejemplos

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "authentication.get",
    "params": {
```

```
        "output": "extend"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "authentication_type": "0",
        "http_auth_enabled": "0",
        "http_login_form": "0",
        "http_strip_domains": "",
        "http_case_sensitive": "1",
        "ldap_auth_enabled": "0",
        "ldap_case_sensitive": "1",
        "ldap_userdirectoryid": "0",
        "saml_auth_enabled": "0",
        "saml_case_sensitive": "0",
        "passwd_min_length": "8",
        "passwd_check_rules": "15",
        "jit_provision_interval": "1h",
        "saml_jit_status": "0",
        "ldap_jit_status": "0",
        "disabled_usrgrpid": "9",
        "mfa_status": "0",
        "mfaid": "0"
    },
    "id": 1
}
```

Fuente

CAuthentication::get() en *ui/include/classes/api/services/CAuthentication.php*.

authentication.update

Descripción

`object authentication.update(object authentication)`

Este método le permite actualizar la configuración de autenticación existente.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración del rol de usuario. Consulte User roles para obtener más información.

Parámetros

`(object)` Propiedades de autenticación a actualizar.

Valores de retorno

`(array)` Devuelve un array con los nombres de los parámetros actualizados.

Ejemplos

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "authentication.update",
    "params": {
        "http_auth_enabled": 1,
        "http_case_sensitive": 0,
        "http_login_form": 1
```

```
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        "http_auth_enabled",
        "http_case_sensitive",
        "http_login_form"
    ],
    "id": 1
}
```

Fuente

CAuthentication::update() en *ui/include/classes/api/services/CAuthentication.php*.

Autoregistro

Esta clase está diseñada para trabajar con el autoregistro.

Referencias de objetos:

- Autoregistro

Métodos disponibles:

- autoregistration.get - recuperar autoregistro
- autoregistration.update - actualizar autoregistro

Objeto de autorregistro

Los siguientes objetos están directamente relacionados con la API de `autoregistration`.

Autoregistro

El objeto de autoregistro tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
tls_accept	integer	Tipo de conexiones entrantes permitidas para el autoregistro. Valores posibles: 1 - permitir conexiones sin cifrar; 2 - permitir TLS con PSK; 3 - permitir tanto conexiones sin cifrar como TLS con PSK.
tls_psk_identity	string	Identidad PSK; debe estar emparejada con una sola PSK (en todo autoregistro, hosts y proxies). No incluya información sensible en la identidad PSK, ya que se envía sin cifrar a través de la red para informar al receptor qué PSK usar. Comportamiento de la propiedad: - *solo escritura*
tls_psk	string	Clave precompartida (PSK); debe tener al menos 32 dígitos hexadecimales. Comportamiento de la propiedad: - *solo escritura*

autoregistration.get

Descripción

`object autoregistration.get(object parameters)`

El método permite recuperar el objeto de autoregistro según los parámetros proporcionados.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración del rol de usuario. Consulte User roles para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método solo admite un parámetro.

Parámetro	Tipo	Descripción
output	query	Este parámetro se describe en el comentario de referencia.

Valores de retorno

`(object)` Devuelve el objeto de autorregistro.

Ejemplos

Solicitud:

```
{
   "jsonrpc": "2.0",
    "method": "autoregistration.get",
    "params": {
        "output": "extend"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "tls_accept": "3"
    },
    "id": 1
}
```

Fuente

CAutoregistration::get() en *ui/include/classes/api/services/CAutoregistration.php*.

autoregistration.update

Descripción

`object autoregistration.update(object autoregistration)`

Este método le permite actualizar una autoregistration existente.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(objeto)` Propiedades de autorregistro que se actualizarán.

Valores de retorno

`(boolean )` Devuelve verdadero booleano como resultado en caso de actualización exitosa.

Ejemplos

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "autoregistration.update",
    "params": {
        "tls_accept": "3",
        "tls_psk_identity": "PSK 001",
        "tls_psk": "11111595725ac58dd977beef14b97461a7c1045b9a1c923453302c5473193478"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": true,
    "id": 1
}
```

Fuente

CAutoregistration::update() en *ui/include/classes/api/services/CAutoregistration.php*.

Complex event processing

This class is designed to work with complex event processing.

Object references:

- CEP rule
- CEP rule filter
  - CEP rule filter condition
- CEP rule time window
- CEP rule operation
- CEP rule operation filter
- CEP rule operation filter condition

Available methods:

- ceprule.create - create new CEP rules
- ceprule.delete - delete CEP rules
- ceprule.get - retrieve CEP rules
- ceprule.update - update CEP rules
- ceprule.resettimewindows - reset time windows for CEP rule

CEP rule object

The following objects are directly related to the `ceprule` API.

CEP rule

The CEP rule object has the following properties.

Property	Type	Description
cep_ruleid	ID	ID of the CEP rule. Property behavior: - *read-only* - *required* for update operations
name	string	Name of the CEP rule. Property behavior: - *required* for create operations

Property	Type	Description
window_type	integer	Time window type: 0 - (default) None 1 - Simple (multiple); 2 - Cause and symptom grouping; 3 - Simple (once); 4 - Event pattern match.
stop	integer	Possible values: 0 - (default) Continue with the next CEP rule according to sort order; 1 - Stop after executing this rule.
sortorder	integer	Execution order of the CEP rule.
description	string	Description of the CEP rule.
status	integer	Whether the CEP rule is enabled or disabled. Possible values: 0 - *(default)* enabled; 1 - disabled.
error	string	Error text if the CEP rule processing failed on rule level. Property behavior: - *read-only*

CEP rule filter

The CEP rule filter object defines a set of conditions that must be met for events to be processed by CEP rule. It has the following properties.

Property	Type	Description
conditions	array	Set of filter conditions to use for filtering results. The conditions will be sorted in the order of their placement in the formula. Property behavior: - *required*
evaltype	integer	Filter condition evaluation method. Possible values: 0 - And/Or; 1 - And; 2 - Or; 3 - Custom expression. Property behavior: - *required*
eval_formula	string	Generated expression that will be used for evaluating filter conditions. The expression contains IDs that reference specific filter conditions by its `formulaid`. The value of `eval_formula` is equal to the value of `formula` for filters with a custom expression. Property behavior: - *read-only*
formula	string	User-defined expression to be used for evaluating conditions of filters with a custom expression. The expression must contain IDs that reference specific filter conditions by its `formulaid`. The IDs used in the expression must exactly match the ones defined in the filter conditions: no condition can remain unused or omitted. Property behavior: - *required* if `evaltype` is set to "custom expression"

CEP rule filter condition

The CEP rule filter condition object defines a specific condition that must be checked before processing events by the CEP rule.

Property	Type	Description
type	integer	Type of condition. Possible values: 3 - event name; 4 - severity; 6 - time period; 25 - tag name; 26 - tag value; 36 - host visible name; 37 - host group name. Property behavior: - *required*
operator	integer	Condition operator. Possible values: 0 - equals (default); 1 - does not equal; 2 - contains; 3 - does not contain; 4 - in; 5 - is greater than or equals; 6 - is less than or equals; 7 - not in. Property behavior: - *required*.
event_name	string	Event name for a specific operation type. Property behavior: - *required* if `type` is set to ”event name”
severity	string	Event severity for a specific operation type. Property behavior: - *required* if `type` is set to ”severity”
time_period	string	Time period for a specific operation type. Property behavior: - *required* if `type` is set to ”time period”
tag	string	Tag name for a specific operation type. Property behavior: - *required* if `type` is set to ”tag name” or ”tag value”
tag_value	string	Tag value for a specific operation type. Property behavior: - *required* if `type` is set to ”tag value”
event_severity	string	Event severity for a specific operation type. Property behavior: - *required* if `type` is set to ”severity”
host_name	string	Host name for a specific operation type. Property behavior: - *required* if `type` is set to ”host visible name”

Property	Type	Description
host_group_name	string	Host group name for a specific operation type. Property behavior: - *required* if `type` is set to ”host group name”
formulaid	string	Arbitrary unique ID that is used to reference the condition from a custom expression. Can only contain capital-case letters. The ID must be defined by the user when modifying filter conditions, but will be generated anew when requesting them afterward. Property behavior: - *required* if CEP rule filter `evaltype` is set to ”custom expression”

CEP rule time window

The CEP rule time window object defines the time window of the CEP rule. It has the following properties.

Property	Type	Description
duration	string	Duration of the CEP rule window (365 days maximum). Accepts seconds and time units with suffix. Default: `10m`
capacity	integer	Capacity of the CEP rule window. Supports user macros. Default: 0 (unlimited)
group_by_host_group	integer	Group by flag for host group. Possible values: 0 - no (default); 1 - yes.
group_by_host	integer	Group by flag for host. Possible values: 0 - no (default); 1 - yes.
group_by_tags	integer	Group by flag for tag. Possible values: 0 - no (default); 1 - yes.
tags	string	Tag names to use for grouping. Property behavior: - *required* if `group_by_tag` is set to ”1”
event_count_tag	string	Tag name for displaying the count of affected events. Property behavior: - *supported* if CEP rule `window_type` is set to ”2”; not allowed otherwise
script	string	Script to execute. Property behavior: - *supported* if CEP rule `window_type` is set to ”4”; not allowed otherwise

CEP rule operation

The CEP rule operation object defines an operation that will be performed when a CEP rule is executed. It has the following properties.

Property	Type	Description
sortorder	integer	Execution order (ascending) of the operation. Property behavior: - *required*
execute_when	integer	Type of condition for operation. Possible values: 0 - event occurred; 1 - event added; 2 - event evicted; 3 - window closed; 4 - event pattern matched; 5 - event closed. Property behavior: - *required*
scope	integer	Scope of the operation. Possible values: 0 - current event (default); 1 - other events; 2 - all events. Possible values when `execute_when` is ”Window closed” or ”Pattern matched”: 2 - all events. Possible values when `execute_when` is ”Event ocurred”: 0 - current event.
filter	object	Filter object.
scope_filter	object	Scope filter object. scope_filter is editable only if the `scope` field contains ”Other events” or ”All events” value. Otherwise scope_filter accepts only object with default values.

Property	Type	Description
type	integer	CEP operation type. Possible values: 1 - set event name; 2 - close event; 3 - discard event; 4 - set event severity; 5 - increase event severity; 6 - decrease event severity; 7 - suppress event; 8 - unsuppress event; 9 - copy first event as new; 10 - copy last event as new; 11 - add tag; 12 - set tag; 13 - set tag value; 14 - increase tag value; 15 - decrease tag value; 16 - rename tag; 17 - remove tag; 18 - close window. Property behavior: - *required* - *supported*: 1, 2, 3, 4, 5, 6, 7, 8, 11, 12, 13, 14, 15, 16, 17 - supported if ”Event occurred” (see `execute_when`) 1, 2, 4, 5, 6, 7, 8, 11, 12, 13, 14, 15, 16, 17, 18 - supported if ”Event added” (see `execute_when`) 1, 2, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18 - supported if ”Event evicted” (see `execute_when`) 1, 2, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18 - supported if ”Event closed” (see `execute_when`) 1, 2, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17 - supported if ”Window closed” (see `execute_when`) 1, 2, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18 - supported if ”Event pattern matched” (see `execute_when`)
event_name	string	Event name for a specific operation type. Property behavior: - *required* if `type` is set to ”Set event name”
tag	string	Tag name for specific operation types. Property behavior: - *required* if `type` is set to ”Add tag”, ”Set tag”, ”Set tag value”, ”Increase tag value”, ”Decrease tag value”, ”Rename tag”, ”Remove tag”
new_tag	string	New tag name for a specific operation type. Property behavior: - *required* if `type` is set to ”Rename tag”
tag_value	string	Tag value for specific operation types. Property behavior: - *required* if `type` is set to ”Add tag”, ”Set tag”, ”Set tag value”, ”Increase tag value”, ”Decrease tag value”, ”Rename tag”, ”Remove tag”
severity	integer	Event severity for a specific operation type. Property behavior: - *required* if `type` is set to ”Set event severity”

Property	Type	Description
suppress_duration	string	Duration of the event suppression. Accepts seconds and time unit with suffix. Property behavior: - *required* if `type` is set to ”Suppress event”

CEP rule operation filter

The CEP rule operation filter object defines a set of conditions that must be met to perform the configured CEP operations. It has the following properties.

Property	Type	Description
conditions	array	Set of operation filter conditions to use for filtering results. The conditions will be sorted in the order of their placement in the formula. Property behavior: - *required*
evaltype	integer	Filter condition evaluation method. Possible values: 0 - And/Or; 1 - And; 2 - Or; 3 - Custom expression. Property behavior: - *required*
eval_formula	string	Generated expression that will be used for evaluating filter conditions. The expression contains IDs that reference specific filter conditions by its `formulaid`. The value of `eval_formula` is equal to the value of `formula` for filters with a custom expression. Property behavior: - *read-only*
formula	string	Arbitrary unique ID that is used to reference the condition from a custom expression. Can only contain capital-case letters. The ID must be defined by the user when modifying filter conditions, but will be generated anew when requesting them afterward. Property behavior: - *required*

CEP rule operation filter condition

The CEP rule operation filter condition object defines a specific condition that must be checked before running the CEP rule operations.

Property	Type	Description
type	integer	Type of operation condition. Possible values: 25 - tag name; 26 - tag value; 30 - problem opened; 31 - first; 32 - last; 33 - symptom; 34 - cloned; 35 - suppressed. Property behavior: - *required* - *supported*: 25, 26, 30, 33, 34, 35 - supported if ”Event occurred” (see `execute_when`)
operator	integer	Condition operator. Possible values: 0 - equals (default); 1 - does not equal; 2 - contains; 3 - does not contain; 5 - is greater than or equals; 6 - is less than or equals; 10 - true; 11 - false. Property behavior: - *required* - *supported*: 0, 1, 2, 3 - supported values if `type` set to ”Tag name” 0, 1, 2, 3, 5, 6 - supported values if `type` set to ”Tag value” 10, 11 - supported values if `type` set to ”Problem opened”, ”First”, ”Last”, ”Symptom”, ”Cloned”, ”Suppressed”.
tag	string	Tag name of operation condition. Property behavior: - *required* if `type` is set to ”Tag name” or ”Tag value”
tag_value	string	Value to compare with. Property behavior: - *required* if `type` is set to ”Tag value”

CEP rule operation scope filter

The CEP rule operation scope filter object defines a set of conditions that must be met to apply the configured CEP operations to events. It has the following properties.

Property	Type	Description
conditions	array	Set of operation scope filter conditions to use for filtering results. The conditions will be sorted in the order of their placement in the formula. Property behavior: - *required*

Property	Type	Description
evaltype	integer	Filter condition evaluation method. Possible values: 0 - And/Or; 1 - And; 2 - Or; 3 - Custom expression. Property behavior: - *required*
eval_formula	string	Generated expression that will be used for evaluating filter conditions. The expression contains IDs that reference specific filter conditions by its `formulaid`. The value of `eval_formula` is equal to the value of `formula` for filters with a custom expression. Property behavior: - *read-only*
formula	string	Arbitrary unique ID that is used to reference the condition from a custom expression. Can only contain capital-case letters. The ID must be defined by the user when modifying filter conditions, but will be generated anew when requesting them afterward. Property behavior: - *required*

CEP rule operation scope filter condition

The CEP rule operation scope filter condition object defines a specific condition that must be checked before applying the CEP rule operations to events.

Property	Type	Description
type	integer	Type of operation condition. Possible values: 25 - tag name; 26 - tag value; 30 - problem opened; 31 - first; 32 - last; 33 - symptom; 34 - cloned; 35 - suppressed. Property behavior: - *required* - *supported*: 25, 26, 30, 33, 34, 35 - supported if ”Event occurred” (see `execute_when`)

Property	Type	Description
operator	integer	Condition operator. Possible values: 0 - equals (default); 1 - does not equal; 2 - contains; 3 - does not contain; 5 - is greater than or equals; 6 - is less than or equals; 10 - true; 11 - false. Property behavior: - *required* - *supported*: 0, 1, 2, 3 - supported values if `type` set to ”Tag name” 0, 1, 2, 3, 5, 6 - supported values if `type` set to ”Tag value” 10, 11 - supported values if `type` set to ”Problem opened”, ”First”, ”Last”, ”Symptom”, ”Cloned”, ”Suppressed”.
tag	string	Tag name of operation condition. Property behavior: - *required* if `type` is set to ”Tag name” or ”Tag value”
tag_value	string	Value to compare with. Property behavior: - *required* if `type` is set to ”Tag value”

ceprule.create

Description

`object ceprule.create(object/array CEP rules)`

This method allows you to create new CEP rules.

Note:
This method is only available to *Super admin* user type. Permissions to call the method can be revoked in user role settings. See User roles for more information.

Parameters

`(object/array)` CEP rules to create.

Additionally to the standard CEP properties, the method accepts the following parameters.

Parameter	Type	Description
operations	array	CEP rule operations to create for the CEP rule. Parameter behavior: - *required*
filter	object	CEP rule filter object for the CEP rule. Parameter behavior: - *required*

Return values

`(object)` Returns an object containing the IDs of the created CEP rules under the `cep_ruleids` property. The order of the returned IDs matches the order of the passed CEP rules.

Examples

Create a new CEP rule

Create a CEP rule using evaluation method `AND/OR` with one condition and one window. By default, the rule will be enabled.

Request:

```
{
    "jsonrpc": "2.0",
    "method": "ceprule.create",
    "params": {
      "name": "Group events by cause",
      "stop": "1",
      "sortorder": "0",
      "description": "",
      "status": "0",
      "window_type": "2",
      "filter": {
        "evaltype": "0",
        "formula": "",
        "conditions": [
          {
            "type": "3",
            "operator": "2",
            "event_name": "Printer",
            "tag": "",
            "tag_value": "",
            "severity": "0",
            "host_name": "",
            "host_group_name": "",
            "time_period": "",
            "formulaid": ""
          }
        ]
      },
      "window": {
        "duration": "1d",
        "capacity": "0",
        "group_by_host_group": "0",
        "group_by_host": "1",
        "group_by_tags": "1",
        "tags": [
          "cause"
        ],
        "event_count_tag": "",
        "script": ""
      },
      "operations": []
    },
    "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "cep_ruleids": [
      "1"
    ]
  },
  "id": 1
}
```

See also

- CEP rule

Source

CCepRule::create() in *ui/include/classes/api/services/CCepRule.php*.

ceprule.delete

Description

`object ceprule.delete(array cep_ruleids)`

This method allows you to delete CEP rules.

> **Note:**
> This method is only available to *Super admin* user type. Permissions to call the method can be revoked in user role settings. See User roles for more information.

Parameters

`(array)` IDs of the CEP rules to delete.

Return values

`(object)` Returns an object containing the IDs of the deleted CEP rules under the `cep_ruleids` property.

Example

Delete multiple CEP rules

Delete two CEP rules.

Request:

```
{
    "jsonrpc": "2.0",
    "method": "ceprule.delete",
    "params": [
        "3",
        "4"
    ],
    "id": 1
}
```

Response:

```
{
    "jsonrpc": "2.0",
    "result": {
        "cep_ruleids": [
            "3",
            "4"
        ]
    },
    "id": 1
}
```

Source

CCepRule::delete() in *ui/include/classes/api/services/CCepRule.php*.

ceprule.get

Description

`integer/array ceprule.get(object parameters)`

The method allows you to retrieve CEP rules according to the given parameters.

Note:
This method is available to users of any type. Permissions to call the method can be revoked in user role settings. See User roles for more information.

Parameters

`(object)` Parameters defining the desired output.

The method supports the following parameters.

Parameter	Type	Description
cep_ruleids	ID/array	Return only CEP rules with the given IDs.
selectFilter	query	Return a `filter` property with the CEP rule conditions.
selectOperations	query	Return an `operations` property with the CEP rule operations.
selectWindow	query	Return a `[window]`(/manual/api/reference/ceprule/object#cep-rule-time-window) property with the CEP rule time window.
sortfield	string/array	Sort the result by the given properties. Possible values: `cep_ruleid`, `name`, `stop`, `sortorder`, `status`.
countOutput	boolean	These parameters are described in the reference commentary.
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Return values

`(integer/array)` Returns either:

- an array of objects;
- the count of retrieved objects, if the `countOutput` parameter has been used.

Examples

Retrieve CEP rules

Retrieve enabled CEP rules, including conditions, windows, and operations.

Request:

```
{
    "jsonrpc": "2.0",
    "method": "ceprule.get",
    "params": {
        "output": "extend",
        "filter": {"status": 0},
        "selectFilter": "extend",
        "selectWindow": "extend",
        "selectOperations": "extend"
    },
    "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "cep_ruleid": "2",
```

```
      "name": "Group events by cause",
      "stop": "0",
      "sortorder": "1",
      "description": "",
      "status": "0",
      "window_type": "2",
      "error": "",
      "filter": {
        "evaltype": "0",
        "eval_formula": "A",
        "formula": "",
        "conditions": [
          {
            "type": "3",
            "operator": "2",
            "event_name": "Printer",
            "tag": "",
            "tag_value": "",
            "severity": "0",
            "host_name": "",
            "host_group_name": "",
            "time_period": "",
            "formulaid": "A"
          }
        ]
      },
      "window": {
        "duration": "1d",
        "capacity": "0",
        "group_by_host_group": "0",
        "group_by_host": "1",
        "group_by_tags": "1",
        "tags": [
          "cause"
        ],
        "event_count_tag": "symptoms",
        "script": ""
      },
      "operations": []
    },
    {
      "cep_ruleid": "10",
      "name": "Suppress consecutive events",
      "stop": "1",
      "sortorder": "2",
      "description": "",
      "status": "0",
      "window_type": "3",
      "error": "",
      "filter": {
        "evaltype": "0",
        "eval_formula": "A",
        "formula": "",
        "conditions": [
          {
            "type": "3",
            "operator": "2",
            "event_name": "Printer",
            "tag": "",
            "tag_value": "",
            "severity": "0",
            "host_name": "",
```

```
          "host_group_name": "",
          "time_period": "",
          "formulaid": "A"
        }
      ]
    },
    "window": {
      "duration": "3m",
      "capacity": "0",
      "group_by_host_group": "0",
      "group_by_host": "1",
      "group_by_tags": "1",
      "tags": [
        "cause"
      ],
      "event_count_tag": "",
      "script": ""
    },
    "operations": [
      {
        "sortorder": "1",
        "execute_when": "1",
        "type": "7",
        "event_name": "",
        "severity": "0",
        "suppress_duration": "6m",
        "tag": "",
        "new_tag": "",
        "tag_value": "",
        "scope": "0",
        "filter": {
          "evaltype": "0",
          "eval_formula": "A",
          "formula": "",
          "conditions": [
            {
              "type": "33",
              "operator": "10",
              "tag": "",
              "tag_value": "",
              "formulaid": "A"
            }
          ]
        },
        "scope_filter": {
          "evaltype": "0",
          "eval_formula": "",
          "formula": "",
          "conditions": []
        }
      },
      {
        "sortorder": "2",
        "execute_when": "2",
        "type": "8",
        "event_name": "",
        "severity": "0",
        "suppress_duration": "0",
        "tag": "",
        "new_tag": "",
        "tag_value": "",
        "scope": "0",
```

```
          "filter": {
            "evaltype": "0",
            "eval_formula": "A",
            "formula": "",
            "conditions": [
              {
                "type": "35",
                "operator": "10",
                "tag": "",
                "tag_value": "",
                "formulaid": "A"
              }
            ]
          },
          "scope_filter": {
            "evaltype": "0",
            "eval_formula": "",
            "formula": "",
            "conditions": []
          }
        }
      ]
    }
  ],
  "id": 1
}
```

See also

- CEP rule

Source

CCepRule::get() in *ui/include/classes/api/services/CCepRule.php*.

ceprule.resettimewindows

Description

`object ceprule.resettimewindows(object/array cep_ruleid)`

This method allows you to reset time windows for an existing CEP rule.

> **Note:**
> This method is only available to *Super admin* user type. Permissions to call the method can be revoked in user role settings. See User roles for more information.

Parameters

`(object)` CEP rule object must have a `cep_ruleid` property defined.

Return values

`(object)` Returns an object containing the ID of the CEP rule whose time windows have been reset under the `cep_ruleid` property.

Examples

Reset time windows for CEP rule

Request:

```
{
    "jsonrpc": "2.0",
    "method": "ceprule.resettimewindows",
    "params": {
        "cep_ruleid": "1"
    },
```

```
    "id": 1
}
```

Response:

```
{
    "jsonrpc": "2.0",
    "result": {
      "cep_ruleid": "1"
    },
    "id": 1
}
```

See also

- CEP rule

Source

CCepRule::resettimewindows() in *ui/include/classes/api/services/CCepRule.php*.

ceprule.update

Description

`object ceprule.update(object/array CEP rules)`

This method allows you to update existing CEP rules.

Note:
This method is only available to *Super admin* user type. Permissions to call the method can be revoked in user role settings. See User roles for more information.

Parameters

`(object/array)` CEP rule properties to be updated.

The `cep_ruleid` property must be defined for each CEP rule, all other properties are optional. Only the passed properties will be updated, all others will remain unchanged.

Additionally to the standard CEP rule properties, the method accepts the following parameters.

Parameter	Type	Description
filter	object	CEP rule filter object to replace the current filter.
window	array	CEP rule window to be updated. Property behavior: - *required* if `window_type` is changed to "Cause and symptom grouping" and "group_by_host_group", "group_by_host", "group_by_tags" are all set to NO; - *required* if `window_type` is changed to "Event pattern match"; - *supported* if `window_type` is "Simple (multiple)" or "Simple (once)"
operations	array	CEP rule operations to replace existing operations. Property behavior: - *required* if `window_type` is changed to "None" (0), "Simple (multiple)" (1), "Simple (once)" (3) or "Event pattern match" (4).

Return values

`(object)` Returns an object containing the IDs of the updated CEP rules under the `cep_ruleids` property.

Examples

Disable CEP rule

Request:

```
{
    "jsonrpc": "2.0",
    "method": "ceprule.update",
    "params": {
        "cep_ruleid": "1",
        "status": "1"
    },
    "id": 1
}
```

Response:

```
{
    "jsonrpc": "2.0",
    "result": {
        "cep_ruleids": [
            "1"
        ]
    },
    "id": 1
}
```

Replace condition

Request:

```
{
    "jsonrpc": "2.0",
    "method": "ceprule.update",
    "params": {
        "cep_ruleid": "3",
        "filter": {
        "evaltype": "0",
        "formula": "",
            "conditions": [
                {
                    "type": "3",
                    "operator": "2",
                    "event_name": "Inaccessible port",
                    "severity": "0"
                }
            ]
        }
    },
    "id": 1
}
```

Response:

```
{
    "jsonrpc": "2.0",
    "result": {
        "cep_ruleids": [
            "1"
        ]
    },
    "id": 1
}
```

See also

- CEP rule

Source

CCepRule::update() in *ui/include/classes/api/services/CCepRule.php*.

Comprobación de descubrimiento

Esta clase está diseñada para trabajar con comprobaciones de descubrimiento.

Referencias de objetos:

- Comprobación de descubrimiento

Métodos disponibles:

- dcheck.get - recuperar comprobaciones de descubrimiento

Objeto de comprobación de descubrimiento

Los siguientes objetos están directamente relacionados con la API `dcheck`.

Comprobación de descubrimiento

El objeto de comprobación de descubrimiento define una comprobación específica realizada por una regla de descubrimiento de red. Tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
dcheckid	ID	ID de la comprobación de descubrimiento.
druleid	ID	ID de la regla de descubrimiento a la que pertenece la comprobación.
key_	string	Clave del item (si `type` está establecido en ”Zabbix agent”) o SNMP OID (si `type` está establecido en ”SNMPv1 agent”, ”SNMPv2 agent” o ”SNMPv3 agent”). Comportamiento de la propiedad: - *obligatorio* si `type` está establecido en ”Zabbix agent”, ”SNMPv1 agent”, ”SNMPv2 agent” o ”SNMPv3 agent”
ports	string	Uno o varios rangos de puertos a comprobar, separados por comas. Por defecto: 0. Comportamiento de la propiedad: - *soportado* si `type` está establecido en ”SSH” (0), ”LDAP” (1), ”SMTP” (2), ”FTP” (3), ”HTTP” (4), ”POP” (5), ”NNTP” (6), ”IMAP” (7), ”TCP” (8), ”Zabbix agent” (9), ”SNMPv1 agent” (10), ”SNMPv2 agent” (11), ”SNMPv3 agent” (13), ”HTTPS” (14) o ”Telnet” (15)
snmp_community	string	Comunidad SNMP. Comportamiento de la propiedad: - *obligatorio* si `type` está establecido en ”SNMPv1 agent” o ”SNMPv2 agent”
snmpv3_authpassphrase	string	Frase de autenticación. Comportamiento de la propiedad: - *soportado* si `type` está establecido en ”SNMPv3 agent” y `snmpv3_securitylevel` está establecido en ”authNoPriv” o ”authPriv”

Propiedad	Tipo	Descripción
snmpv3_authprotocol	integer	Protocolo de autenticación. Valores posibles: 0 - *(por defecto)* MD5; 1 - SHA1; 2 - SHA224; 3 - SHA256; 4 - SHA384; 5 - SHA512. Comportamiento de la propiedad: - *soportado* si `type` está establecido en ”SNMPv3 agent” y `snmpv3_securitylevel` está establecido en ”authNoPriv” o ”authPriv”
snmpv3_contextname	string	Nombre de contexto SNMPv3. Comportamiento de la propiedad: - *soportado* si `type` está establecido en ”SNMPv3 agent”
snmpv3_privpassphrase	string	Frase de privacidad. Comportamiento de la propiedad: - *soportado* si `type` está establecido en ”SNMPv3 agent” y `snmpv3_securitylevel` está establecido en ”authPriv”
snmpv3_privprotocol	integer	Protocolo de privacidad. Valores posibles: 0 - *(por defecto)* DES; 1 - AES128; 2 - AES192; 3 - AES256; 4 - AES192C; 5 - AES256C. Comportamiento de la propiedad: - *soportado* si `type` está establecido en ”SNMPv3 agent” y `snmpv3_securitylevel` está establecido en ”authPriv”
snmpv3_securitylevel	string	Nivel de seguridad. Valores posibles: 0 - noAuthNoPriv; 1 - authNoPriv; 2 - authPriv. Comportamiento de la propiedad: - *soportado* si `type` está establecido en ”SNMPv3 agent”
snmpv3_securityname	string	Nombre de seguridad. Comportamiento de la propiedad: - *soportado* si `type` está establecido en ”SNMPv3 agent”

Propiedad	Tipo	Descripción
type	integer	Tipo de comprobación. Valores posibles: 0 - SSH; 1 - LDAP; 2 - SMTP; 3 - FTP; 4 - HTTP; 5 - POP; 6 - NNTP; 7 - IMAP; 8 - TCP; 9 - Zabbix agent; 10 - SNMPv1 agent; 11 - SNMPv2 agent; 12 - ICMP ping; 13 - SNMPv3 agent; 14 - HTTPS; 15 - Telnet. Comportamiento de la propiedad: - *obligatorio*
uniq	integer	Si se debe utilizar esta comprobación como criterio de unicidad del dispositivo. Sólo se puede configurar una comprobación única para una regla de descubrimiento. Valores posibles: 0 - *(por defecto)* no utilizar esta comprobación como criterio de unicidad; 1 - utilizar esta comprobación como criterio de unicidad. Comportamiento de la propiedad: - *soportado* si `type` está establecido en ”Zabbix agent”, ”SNMPv1 agent”, ”SNMPv2 agent” o ”SNMPv3 agent”
host_source	integer	Fuente para el nombre del host. Valores posibles: 1 - *(por defecto)* DNS; 2 - IP; 3 - valor de descubrimiento de esta comprobación.
name_source	integer	Fuente para el nombre visible. Valores posibles: 0 - *(por defecto)* no especificado; 1 - DNS; 2 - IP; 3 - valor de descubrimiento de esta comprobación.
allow_redirect	integer	Permitir la situación en la que el objetivo al que se hace ping ICMP responde desde una dirección IP diferente. Valores posibles: 0 - *(por defecto)* tratar las respuestas redirigidas como si el host de destino estuviera caído (fallo); 1 - tratar las respuestas redirigidas como si el host de destino estuviera activo (éxito). Comportamiento de la propiedad: - *soportado* si `type` está establecido en ”ICMP ping”

dcheck.get

Descripción

`integer/array dcheck.get(object parameters)`

El método permite recuperar comprobaciones de descubrimiento según los parámetros proporcionados.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método soporta los siguientes parámetros.

Parámetro	Tipo	Descripción
dcheckids	ID/array	Devolver solo los chequeos de descubrimiento con los IDs dados.
druleids	ID/array	Devolver solo los chequeos de descubrimiento que pertenecen a las reglas de descubrimiento dadas.
dserviceids	ID/array	Devolver solo los chequeos de descubrimiento que han detectado los servicios descubiertos dados.
selectDRules	query	Devolver las reglas de descubrimiento relacionadas con los chequeos de descubrimiento.
sortfield	string/array	Ordenar el resultado por las propiedades dadas. Valores posibles: `dcheckid`, `druleid`.
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperar comprobaciones de descubrimiento para una regla de descubrimiento

Recuperar todas las comprobaciones de descubrimiento utilizadas por la regla de descubrimiento "6".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "dcheck.get",
    "params": {
        "output": "extend",
        "dcheckids": "6"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "dcheckid": "6",
            "druleid": "4",
            "type": "3",
            "key_": "",
            "snmp_community": "",
            "ports": "21",
            "snmpv3_securityname": "",
            "snmpv3_securitylevel": "0",
            "snmpv3_authpassphrase": "",
            "snmpv3_privpassphrase": "",
            "uniq": "0",
            "snmpv3_authprotocol": "0",
            "snmpv3_privprotocol": "0",
            "snmpv3_contextname": "",
            "host_source": "1",
            "name_source": "0",
            "allow_redirect": "0"
        }
    ],
    "id": 1
}
```


Fuente

CDCheck::get() en *ui/include/classes/api/services/CDCheck.php*.

Conector

Esta clase está diseñada para trabajar con conectores.

Referencias de objetos:

- Conector
- Filtro de etiqueta

Métodos disponibles:

- connector.create - crear nuevos conectores
- connector.delete - eliminar conectores
- connector.get - obtener conectores
- connector.update - actualizar conectores

Objeto conector

Los siguientes objetos están directamente relacionados con la API de `connector`.

Conector

El objeto conector tiene las siguientes propiedades.

Property	Type	Description
connectorid	ID	ID del conector. Comportamiento de la propiedad: - *solo lectura* - *obligatoria* para las operaciones de actualización
name	string	Nombre del conector. Comportamiento de la propiedad: - *obligatoria* para las operaciones de creación

Property	Type	Description
url	string	URL del endpoint, es decir, la URL del receptor. Se admiten macros de usuario. Comportamiento de la propiedad: - *obligatoria* para las operaciones de creación
protocol	integer	Protocolo de comunicación. Valores posibles: 0 - *(predeterminado)* Zabbix Streaming Protocol v1.0.
data_type	integer	Tipo de datos. Valores posibles: 0 - *(predeterminado)* Valores de item; 1 - Eventos.
item_value_type	integer	Suma de los tipos de valores de item que se enviarán. Valores posibles: 1 - Numérico (flotante); 2 - Carácter; 4 - Registro; 8 - Numérico (sin signo); 16 - Texto; 32 - Binario; 64 - JSON. Predeterminado: 31 - Todos los tipos de item (excepto binario y JSON).
max_records	integer	Comportamiento de la propiedad: - *compatible* si `data_type` está establecido en ”Valores de item”. Número máximo de eventos o items que se pueden enviar en un mensaje. Valores posibles: 0-2147483647 (valor máximo de un entero con signo de 32 bits). Predeterminado: 0 - Ilimitado.
max_senders	integer	Número de procesos emisores que se ejecutarán para este conector. Valores posibles: 1-100. Predeterminado: 1.
max_attempts	integer	Número de intentos. Valores posibles: 1-5. Predeterminado: 1.
attempt_interval	string	Intervalo entre los intentos de reintento. Acepta segundos. Valores posibles: 0s-10s. Predeterminado: 5s.
timeout	string	Comportamiento de la propiedad: - *compatible* si `max_attempts` es mayor que 1. Tiempo de espera. Se admiten sufijos de tiempo (por ejemplo, `30s`, `1m`). Se admiten macros de usuario. Valores posibles: 1s-60s. Predeterminado: 5s.

Property	Type	Description
http_proxy	string	Cadena de conexión del proxy HTTP(S) indicada como *[protocol]://[username[:password]@]proxy.example.com[:port]*. Se admiten macros de usuario.
authtype	integer	Método de autenticación HTTP. Valores posibles: 0 - *(predeterminado)* Ninguno; 1 - Básica; 2 - NTLM; 3 - Kerberos; 4 - Digest; 5 - Bearer.
username	string	Nombre de usuario. Se admiten macros de usuario. Comportamiento de la propiedad: - *compatible* si `authtype` está establecido en ”Básica”, ”NTLM”, ”Kerberos” o ”Digest”
password	string	Contraseña. Se admiten macros de usuario. Comportamiento de la propiedad: - *compatible* si `authtype` está establecido en ”Básica”, ”NTLM”, ”Kerberos” o ”Digest”
token	string	Token Bearer. Se admiten macros de usuario. Comportamiento de la propiedad: - *obligatoria* si `authtype` está establecido en ”Bearer”
verify_peer	integer	Indica si se debe validar que el certificado del host sea auténtico. Valores posibles: 0 - No validar; 1 - *(predeterminado)* Validar.
verify_host	integer	Indica si se debe validar que el nombre del host de la conexión coincida con el del certificado del host. Valores posibles: 0 - No validar; 1 - *(predeterminado)* Validar.
ssl_cert_file	string	Ruta al archivo de clave pública SSL. Se admiten macros de usuario.
ssl_key_file	string	Ruta al archivo de clave privada SSL. Se admiten macros de usuario.
ssl_key_password	string	Contraseña del archivo de clave SSL. Se admiten macros de usuario.
description	text	Descripción del conector.
status	integer	Indica si el conector está habilitado. Valores posibles: 0 - Deshabilitado; 1 - *(predeterminado)* Habilitado.
tags_evaltype	integer	Método de evaluación de las etiquetas. Valores posibles: 0 - *(predeterminado)* Y/O; 2 - O.

Filtro de etiquetas

El filtro de etiquetas permite exportar solo los valores de item o eventos que coincidan. Si no se establece, se exportará todo. El objeto de filtro de etiquetas tiene las siguientes propiedades.

Property	Type	Description
tag	string	Nombre de la etiqueta. Property behavior: - *required*
operator	integer	Operador de la condición. Valores posibles: 0 - *(default)* Igual a; 1 - No es igual a; 2 - Contiene; 3 - No contiene; 12 - Existe; 13 - No existe.
value	string	Valor de la etiqueta.

connector.create

Descripción

`object connector.create(object/array connectors)`

Este método permite crear nuevos objetos de conector.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(objeto/array)` Objetos de conector a crear.

Además de las propiedades estándar del conector, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
tags	array	Filtro de etiquetas del conector.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los conectores creados bajo la propiedad `connectorids`. El orden de los IDs devueltos coincide con el orden de los conectores pasados.

Ejemplos

Creando un conector

Cree un conector para exportar eventos de disparadores con un filtro de etiquetas. La autenticación HTTP se realizará utilizando un token Bearer.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "connector.create",
    "params": [
        {
            "name": "Export of events",
            "data_type": 1,
            "url": "{$DATA_EXPORT_URL}",
            "authtype": 5,
            "token": "{$DATA_EXPORT_BEARER_TOKEN}",
```

```
            "tags": [
                {
                    "tag": "service",
                    "operator": 0,
                    "value": "mysqld"
                }
            ]
        }
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "connectorid": [
            "3"
        ]
    },
    "id": 1
}
```

Fuente

CConnector::create() en *ui/include/classes/api/services/CConnector.php*.

connector.delete

Descripción

`object connector.delete(array connectorids)`

Este método le permite eliminar entradas de conector.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(array)` IDs de los conectores a eliminar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los conectores eliminados bajo la propiedad `connectorids`.

Ejemplos

Eliminando múltiples conectores

Eliminar dos entradas de conector.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "connector.delete",
    "params": [
        3,
        5
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "connectorids": [
            "3",
            "5"
        ]
    },
    "id": 1
}
```

Fuente

CConnector::delete() en *ui/include/classes/api/services/CConnector.php*.

connector.get

Descripción

`integer/array connector.get(object parameters)`

El método permite recuperar objetos de conector según los parámetros proporcionados.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
connectorids	ID/array	Devolver solo los connectors con los IDs dados.
selectTags	query	Devuelve una propiedad `tags` con el filtro de etiquetas del connector. Admite `count`.
sortfield	string/array	Ordenar el resultado por las propiedades dadas. Valores posibles: `connectorid`, `name`, `data_type`, `status`.
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperar todos los conectores

Recupera todos los datos sobre todos los conectores y sus propiedades.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "connector.get",
    "params": {
        "output": "extend",
        "selectTags": ["tag", "operator", "value"],
        "preservekeys": true
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "connectorid": "1",
            "name": "Exportación de valores de métricas",
            "protocol": "0",
            "data_type": "0",
            "url": "{$DATA_EXPORT_VALUES_URL}",
            "item_value_type": "31",
            "authtype": "4",
            "username": "{$DATA_EXPORT_VALUES_USERNAME}",
            "password": "{$DATA_EXPORT_VALUES_PASSWORD}",
            "token": "",
            "max_records": "0",
            "max_senders": "4",
            "max_attempts": "2",
            "attempt_interval": "10s",
            "timeout": "10s",
            "http_proxy": "{$DATA_EXPORT_VALUES_PROXY}",
            "verify_peer": "1",
            "verify_host": "1",
            "ssl_cert_file": "{$DATA_EXPORT_VALUES_SSL_CERT_FILE}",
            "ssl_key_file": "{$DATA_EXPORT_VALUES_SSL_KEY_FILE}",
            "ssl_key_password": "",
            "description": "",
            "status": "1",
            "tags_evaltype": "0",
            "tags": [
                {
                    "tag": "component",
                    "operator": "0",
                    "value": "memory"
                }
            ]
        },
        {
            "connectorid": "2",
            "name": "Exportación de eventos",
            "protocol": "0",
            "data_type": "1",
            "url": "{$DATA_EXPORT_EVENTS_URL}",
            "item_value_type": "31",
            "authtype": "5",
            "username": "",
            "password": "",
            "token": "{$DATA_EXPORT_EVENTS_BEARER_TOKEN}",
            "max_records": "0",
            "max_senders": "2",
```

```
            "max_attempts": "1",
            "attempt_interval": "5s",
            "timeout": "5s",
            "http_proxy": "",
            "verify_peer": "1",
            "verify_host": "1",
            "ssl_cert_file": "",
            "ssl_key_file": "",
            "ssl_key_password": "",
            "description": "",
            "status": "1",
            "tags_evaltype": "0",
            "tags": [
                {
                    "tag": "scope",
                    "operator": "0",
                    "value": "performance"
                }
            ]
        }
    ],
    "id": 1
}
```

Fuente

CConnector:get() en *ui/include/classes/api/services/CConnector.php*.

connector.update

Descripción

`object connector.update(object/array connectors)`

Este método permite actualizar conectores existentes.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(objeto/array)` Propiedades del conector que se actualizarán.

La propiedad `connectorid` debe estar definida para cada conector, todas las demás propiedades son opcionales. Sólo se actualizarán las propiedades pasadas, todas las demás permanecerán sin cambios.

Además de las propiedades estándar del conector, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
tags	array	Filtro de etiquetas del conector para reemplazar el filtro de etiquetas actual.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los conectores actualizados bajo la propiedad `connectorids`.

Ejemplos

Cambiar el tipo de autenticación HTTP

Cambie el tipo de autenticación HTTP a Bearer para el conector con ID "3".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "connector.update",
    "params": {
        "connectorid": 3,
        "authtype": 5,
        "token": "{$DATA_EXPORT_BEARER_TOKEN}"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "connectorids": [
            "3"
        ]
    },
    "id": 1
}
```

Actualización del filtro de etiquetas

Cambiar el filtro de etiquetas para el conector con ID ”5”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "connector.update",
    "params": [
        {
            "connectorid": 5,
            "tags_evaltype": 2,
            "tags": [
                {
                    "tag": "service",
                    "operator": 0,
                    "value": "mysqld"
                },
                {
                    "tag": "error",
                    "operator": 12,
                    "value": ""
                }
            ]
        }
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "connectorids": [
            "5"
        ]
    },
    "id": 1
}
```

Fuente

CConnector::update() en *ui/include/classes/api/services/CConnector.php*.

Configuración

Esta clase está diseñada para trabajar con configuraciones administrativas comunes.

Referencias de objetos:

- Configuración

Métodos disponibles:

- settings.get - recuperar configuración
- settings.update - actualizar configuración

Objeto de configuración

Los siguientes objetos están directamente relacionados con la API de `settings`.

Configuración

El objeto de configuración tiene las siguientes propiedades.

Property	Type	Description
default_lang	string	Idioma del sistema de forma predeterminada. Predeterminado: `en_US`.
default_timezone	string	Zona horaria del sistema de forma predeterminada. Predeterminado: `system` - valor predeterminado del sistema. Para consultar la lista completa de zonas horarias compatibles, consulte la documentación de PHP.
default_theme	string	Tema predeterminado. Valores posibles: `blue-theme` - *(predeterminado)* Azul; `blue-classic-theme` - Azul (clásico); `dark-theme` - Oscuro; `dark-blue-theme` - Azul oscuro; `dark-classic-theme` - Oscuro (clásico); `hc-light` - Claro de alto contraste; `hc-dark` - Oscuro de alto contraste.
search_limit	integer	Límite para los resultados de búsqueda y filtrado. Predeterminado: 1000.
max_overview_table_size	integer	Número máximo de columnas y filas en los widgets del panel de resumen de datos y resumen de triggers. Predeterminado: 50.
max_in_table	integer	Número máximo de elementos que se mostrarán dentro de una celda de tabla. Predeterminado: 50.
server_check_interval	integer	Mostrar una advertencia si el server de Zabbix está inactivo. Valores posibles: 0 - No mostrar la advertencia; 10 - *(predeterminado)* Mostrar la advertencia.
work_period	string	Horario laboral. Predeterminado: 1-5,09:00-18:00.

Property	Type	Description
show_technical_errors	integer	Mostrar errores técnicos (PHP/SQL) a usuarios que no sean Superadministradores y a usuarios que no formen parte de grupos de usuarios con el modo de depuración activado. Valores posibles: 0 - *(predeterminado)* No mostrar errores técnicos; 1 - Mostrar errores técnicos.
history_period	string	Período máximo para mostrar datos históricos en los widgets del panel Últimos datos, Web y resumen de datos. Acepta un número de segundos (por ejemplo, 1, 60, 300) o un valor con un sufijo de tiempo (por ejemplo, 30s, 1m, 2h, 1d). Predeterminado: 24h.
period_default	string	Período predeterminado del filtro de tiempo. Acepta un número de segundos (por ejemplo, 1, 60, 300) o un valor con un sufijo de tiempo (por ejemplo, 30s, 1m, 2h, 1d, 1M, 1y), incluidos meses y años. Predeterminado: 1h.
max_period	string	Período máximo del filtro de tiempo. Acepta un número de segundos (por ejemplo, 1, 60, 300) o un valor con un sufijo de tiempo (por ejemplo, 30s, 1m, 2h, 1d, 1M, 1y), incluidos meses y años. Predeterminado: 2y.
severity_color_0	string	Color de la gravedad ”Sin clasificar” como código de color hexadecimal. Predeterminado: 97AAB3.
severity_color_1	string	Color de la gravedad ”Información” como código de color hexadecimal. Predeterminado: 7499FF.
severity_color_2	string	Color de la gravedad ”Advertencia” como código de color hexadecimal. Predeterminado: FFC859.
severity_color_3	string	Color de la gravedad ”Promedio” como código de color hexadecimal. Predeterminado: FFA059.
severity_color_4	string	Color de la gravedad ”Alta” como código de color hexadecimal. Predeterminado: E97659.
severity_color_5	string	Color de la gravedad ”Desastre” como código de color hexadecimal. Predeterminado: E45959.
severity_name_0	string	Nombre de la gravedad ”Sin clasificar”. Predeterminado: Sin clasificar.
severity_name_1	string	Nombre de la gravedad ”Información”. Predeterminado: Información.
severity_name_2	string	Nombre de la gravedad ”Advertencia”. Predeterminado: Advertencia.
severity_name_3	string	Nombre de la gravedad ”Promedio”. Predeterminado: Promedio.
severity_name_4	string	Nombre de la gravedad ”Alta”. Predeterminado: Alta.
severity_name_5	string	Nombre de la gravedad ”Desastre”. Predeterminado: Desastre.

Property	Type	Description
custom_color	integer	Usar colores personalizados para el estado de los eventos. Valores posibles: 0 - *(predeterminado)* No usar colores personalizados para el estado de los eventos; 1 - Usar colores personalizados para el estado de los eventos.
ok_period	string	Período de visualización de los triggers OK. Acepta un número de segundos (por ejemplo, 1, 60, 300) o un valor con un sufijo de tiempo (por ejemplo, 30s, 1m, 2h). Predeterminado: 5m.
blink_period	string	Período de parpadeo al cambiar el estado del trigger. Acepta un número de segundos (por ejemplo, 1, 60, 300) o un valor con un sufijo de tiempo (por ejemplo, 30s, 1m, 2h). Predeterminado: 2m.
problem_unack_color	string	Color de los eventos PROBLEM no reconocidos como código de color hexadecimal. Predeterminado: CC0000.
problem_ack_color	string	Color de los eventos PROBLEM reconocidos como código de color hexadecimal. Predeterminado: CC0000.
ok_unack_color	string	Color de los eventos RESOLVED no reconocidos como código de color hexadecimal. Predeterminado: 009900.
ok_ack_color	string	Color de los eventos RESOLVED reconocidos como código de color hexadecimal. Predeterminado: 009900.
problem_unack_style	integer	Parpadeo de los eventos PROBLEM no reconocidos. Valores posibles: 0 - No mostrar parpadeo; 1 - *(predeterminado)* Mostrar parpadeo.
problem_ack_style	integer	Parpadeo de los eventos PROBLEM reconocidos. Valores posibles: 0 - No mostrar parpadeo; 1 - *(predeterminado)* Mostrar parpadeo.
ok_unack_style	integer	Parpadeo de los eventos RESOLVED no reconocidos. Valores posibles: 0 - No mostrar parpadeo; 1 - *(predeterminado)* Mostrar parpadeo.
ok_ack_style	integer	Parpadeo de los eventos RESOLVED reconocidos. Valores posibles: 0 - No mostrar parpadeo; 1 - *(predeterminado)* Mostrar parpadeo.
url	string	URL del frontend.
discovery_groupid	ID	ID del grupo de hosts en el que se colocarán automáticamente los hosts descubiertos.
default_inventory_mode	integer	Modo de inventario de hosts predeterminado. Valores posibles: -1 - *(predeterminado)* Deshabilitado; 0 - Manual; 1 - Automático.

Property	Type	Description
alert_usrgrpid	ID	ID del grupo de usuarios al que se enviará el mensaje de alarma de caída de la base de datos. Si se establece en ”0”, no se enviará el mensaje de alarma.
snmptrap_logging	integer	Registrar traps SNMP no coincidentes. Valores posibles: 0 - No registrar traps SNMP no coincidentes; 1 - *(predeterminado)* Registrar traps SNMP no coincidentes.
login_attempts	integer	Número de intentos de inicio de sesión fallidos después del cual se bloqueará el formulario de inicio de sesión. Predeterminado: 5.
login_block	string	Intervalo durante el cual se bloqueará el formulario de inicio de sesión si el número de intentos de inicio de sesión fallidos supera el definido en el campo login_attempts. Acepta un número de segundos (por ejemplo, 1, 60) o un valor con un sufijo de tiempo (por ejemplo, 30s, 1m). Predeterminado: 30s.
validate_uri_schemes	integer	Validar esquemas URI. Valores posibles: 0 - No validar; 1 - *(predeterminado)* Validar.
uri_valid_schemes	string	Esquemas URI válidos. Predeterminado: http,https,ftp,file,mailto,tel,ssh.
x_frame_options	string	Encabezado HTTP X-Frame-Options. Predeterminado: SAMEORIGIN.
iframe_sandboxing_enabled	integer	Usar el aislamiento de iframe. Valores posibles: 0 - No usar; 1 - *(predeterminado)* Usar.
iframe_sandboxing_exceptions	string	Excepciones del aislamiento de iframe.
connect_timeout	string	Tiempo de espera de conexión con el server de Zabbix. Rango de valores posibles: 1-30s. Predeterminado: 3s. Comportamiento de la propiedad: - *required*
socket_timeout	string	Tiempo de espera de red predeterminado. Rango de valores posibles: 1-300s. Predeterminado: 3s. Comportamiento de la propiedad: - *required*
media_type_test_timeout	string	Tiempo de espera de red para la prueba del tipo de medio. Rango de valores posibles: 1-300s. Predeterminado: 65s. Comportamiento de la propiedad: - *required*

Property	Type	Description
item_test_timeout	string	Tiempo de espera de red para las pruebas de items. Rango de valores posibles: 1-600s. Predeterminado: 60s. Comportamiento de la propiedad: - *required*
script_timeout	string	Tiempo de espera de red para la ejecución de scripts. Rango de valores posibles: 1-300s. Predeterminado: 60s. Comportamiento de la propiedad: - *required*
report_test_timeout	string	Tiempo de espera de red para la prueba de informes programados. Rango de valores posibles: 1-300s. Predeterminado: 60s. Comportamiento de la propiedad: - *required*
device_link_timeout	string	Tiempo de espera para añadir un dispositivo móvil para usarlo con la aplicación móvil de Zabbix; se aplica a la comunicación entre la API de Zabbix y el adaptador bridge. Rango de valores posibles: 1-300s. Predeterminado: 60s. Comportamiento de la propiedad: - *required*
auditlog_enabled	integer	Indica si se debe activar el registro de auditoría. Valores posibles: 0 - Desactivar; 1 - *(predeterminado)* Activar.
auditlog_mode	integer	Indica si se debe activar el registro de auditoría de las actividades de descubrimiento de bajo nivel, descubrimiento de red y registro automático realizadas por el server (usuario System). Valores posibles: 0 - Desactivar; 1 - *(predeterminado)* Activar.
ha_failover_delay	string	Retraso de conmutación por error en segundos. Predeterminado: 1m. Comportamiento de la propiedad: - *read-only*

Property	Type	Description
geomaps_tile_provider	string	Proveedor de mosaicos de mapas geográficos. Valores posibles: `OpenStreetMap.Mapnik` - *(predeterminado)* OpenStreetMap Mapnik; `OpenTopoMap` - OpenTopoMap; `Stamen.TonerLite` - Stamen Toner Lite; `Stamen.Terrain` - Stamen Terrain; `USGS.USTopo` - USGS US Topo; `USGS.USImagery` - USGS US Imagery. Admite una cadena vacía para especificar valores personalizados de `geomaps_tile_url`, `geomaps_max_zoom` y `geomaps_attribution`.
geomaps_tile_url	string	URL de los mosaicos de mapas geográficos. Comportamiento de la propiedad: - *supported* si `geomaps_tile_provider` se establece en una cadena vacía
geomaps_max_zoom	integer	Nivel máximo de zoom de los mapas geográficos. Valores posibles: 0-30. Comportamiento de la propiedad: - *supported* si `geomaps_tile_provider` se establece en una cadena vacía
geomaps_attribution	string	Texto de atribución de los mapas geográficos. Comportamiento de la propiedad: - *supported* si `geomaps_tile_provider` se establece en una cadena vacía
vault_provider	integer	Proveedor de Vault. Valores posibles: 0 - *(predeterminado)* HashiCorp Vault; 1 - CyberArk Vault.
proxy_secrets_provider	integer	Resolver los valores de macros secretas mediante: Valores posibles: 0 - *(predeterminado)* solo el server; 1 - el server y los proxies de forma independiente.
timeout_zabbix_agent	string	No dedicar más de `timeout_*` segundos al procesamiento. Acepta un número de segundos (por ejemplo, 1, 60, 300) o un valor con un sufijo de tiempo (por ejemplo, 30s, 1m). También acepta macros de usuario. Rango de valores posibles: 1-600s. Predeterminado: 3s. Predeterminado para `timeout_browser`: 60s. Comportamiento de la propiedad: - *required*
timeout_simple_check		
timeout_snmp_agent		
timeout_external_check		
timeout_db_monitor		
timeout_http_agent		
timeout_ssh_agent		
timeout_telnet_agent		
timeout_script		
timeout_browser		

Property	Type	Description
timeout_telemetry_query		
apm_global_db	object	Configuración de la fuente de datos global.

Fuente de datos global

El objeto de fuente de datos global tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
status	integer	Estado de la fuente de datos global. Valores posibles: 0 - *(predeterminado)* no configurada; 1 - configurada.
url	string	URL utilizada para conectarse a la fuente de datos. Comportamiento de la propiedad: - *obligatoria* si el usuario cambia `status` a 1 (configurada).
authentication_type	integer	Tipo de autenticación. Valores posibles: 0 - *(predeterminado)* se utilizan un nombre de usuario y una contraseña; 1 - no se utiliza autenticación. Comportamiento de la propiedad: - *obligatoria* si el usuario cambia `status` a 1 (configurada).
username	string	Nombre de usuario utilizado para iniciar sesión en la fuente de datos. Comportamiento de la propiedad: - *compatible* si `status` está establecido en 1 (configurada) y `authentication_type` está establecido en 0 (nombre de usuario y contraseña). - *obligatorio* si el usuario cambia `authentication_type` a 0 (nombre de usuario y contraseña).
password	string	Contraseña utilizada para iniciar sesión en la fuente de datos. El valor de este parámetro puede estar vacío. Comportamiento de la propiedad: - *compatible* si `status` está establecido en 1 (configurada) y `authentication_type` está establecido en 0 (nombre de usuario y contraseña).
db	string	Nombre de la base de datos. El valor de este parámetro puede estar vacío. Comportamiento de la propiedad: - *compatible* si `status` está establecido en 1 (configurada).
ssl_verify_peer	integer	Verificar el certificado del server. Valores posibles: 0 - *(predeterminado)* deshabilitado; 1 - habilitado. Comportamiento de la propiedad: - *compatible* si `status` está establecido en 1 (configurada) y `url` utiliza el esquema `https://`.

Propiedad	Tipo	Descripción
ssl_verify_host	integer	Verificar el host. Valores posibles: 0 - deshabilitado; 1 - *(predeterminado)* habilitado. Comportamiento de la propiedad: - *compatible* si `status` está establecido en 1 (configurada) y `url` utiliza el esquema `https://`.

settings.get

Descripción

`object settings.get(object parameters)`

El método permite recuperar el objeto de configuración según los parámetros proporcionados.

Note:
Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método solo admite un parámetro.

Parámetro	Tipo	Descripción
output	query	Este parámetro se describe en el comentario de referencia.

Valores de retorno

`(object)` Devuelve el objeto de configuración.

Ejemplos

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "settings.get",
    "params": {
        "output": "extend"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "alert_usrgrpid": "7",
        "apm_global_db": {
            "status": 1,
            "url": "https://localhost:8443",
            "authentication_type": 0,
            "username": "zabbix",
            "db": "",
            "ssl_verify_peer": 1,
            "ssl_verify_host": 1
        },
```

```
        "auditlog_enabled": "1",
        "auditlog_mode": "1",
        "blink_period": "2m",
        "connect_timeout": "3s",
        "custom_color": "0",
        "default_inventory_mode": "-1",
        "default_lang": "en_US",
        "default_theme": "blue-theme",
        "default_timezone": "system",
        "device_link_timeout": "60s",
        "discovery_groupid": "5",
        "geomaps_attribution": "",
        "geomaps_max_zoom": "0",
        "geomaps_tile_provider": "OpenStreetMap.Mapnik",
        "geomaps_tile_url": "",
        "ha_failover_delay": "1m",
        "history_period": "24h",
        "iframe_sandboxing_enabled": "1",
        "iframe_sandboxing_exceptions": "",
        "item_test_timeout": "60s",
        "login_attempts": "5",
        "login_block": "30s",
        "max_in_table": "50",
        "max_overview_table_size": "50",
        "max_period": "2y",
        "media_type_test_timeout": "65s",
        "ok_ack_color": "009900",
        "ok_ack_style": "1",
        "ok_period": "5m",
        "ok_unack_color": "009900",
        "ok_unack_style": "1",
        "period_default": "1h",
        "problem_ack_color": "CC0000",
        "problem_ack_style": "1",
        "problem_unack_color": "CC0000",
        "problem_unack_style": "1",
        "proxy_secrets_provider": "0",
        "report_test_timeout": "60s",
        "script_timeout": "60s",
        "search_limit": "1000",
        "server_check_interval": "10",
        "serverid": "019f4707-1a3e-70e6-aca2-286a2b696358",
        "severity_color_0": "97AAB3",
        "severity_color_1": "7499FF",
        "severity_color_2": "FFC859",
        "severity_color_3": "FFA059",
        "severity_color_4": "E97659",
        "severity_color_5": "E45959",
        "severity_name_0": "No clasificado",
        "severity_name_1": "Información",
        "severity_name_2": "Advertencia",
        "severity_name_3": "Promedio",
        "severity_name_4": "Alto",
        "severity_name_5": "Desastre",
        "show_technical_errors": "0",
        "snmptrap_logging": "1",
        "socket_timeout": "3s",
        "timeout_browser": "60s",
        "timeout_db_monitor": "3s",
        "timeout_external_check": "3s",
        "timeout_http_agent": "3s",
        "timeout_script": "3s",
```

```
        "timeout_simple_check": "3s",
        "timeout_snmp_agent": "3s",
        "timeout_ssh_agent": "3s",
        "timeout_telemetry_query": "3s",
        "timeout_telnet_agent": "3s",
        "timeout_zabbix_agent": "3s",
        "uri_valid_schemes": "http,https,ftp,file,mailto,tel,ssh",
        "url": "",
        "validate_uri_schemes": "1",
        "vault_provider": "0",
        "work_period": "1-5,09:00-18:00",
        "x_frame_options": "SAMEORIGIN"
    },
    "id": 1
}
```


Fuente

CSettings::get() en *ui/include/classes/api/services/CSettings.php*.

settings.update

Descripción

`object settings.update(object settings)`

Este método le permite actualizar la configuración común existente.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración del rol de usuario. Consulte User roles para obtener más información.

Parámetros

`(object)` Propiedades de configuración que se actualizarán.

Valores de retorno

`(array)` Devuelve un array con los nombres de los parámetros actualizados.

Ejemplos

Solicitud:


```
{
    "jsonrpc": "2.0",
    "method": "settings.update",
    "params": {
        "login_attempts": "1",
        "login_block": "1m"
    },
    "id": 1
}
```


Respuesta:


```
{
    "jsonrpc": "2.0",
    "result": [
        "login_attempts",
        "login_block"
    ],
    "id": 1
}
```


Fuente

CSettings::update() en *ui/include/classes/api/services/CSettings.php*.

Configuración

Esta clase está diseñada para exportar e importar datos de configuración de Zabbix.

Métodos disponibles:

- configuration.export - exportar datos de configuración
- configuration.import - importar datos de configuración
- configuration.importcompare - comparar el archivo de importación con los elementos actuales del sistema

configuration.export

Descripción

`string configuration.export(object parameters)`

Este método permite exportar datos de configuración como una cadena serializada.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object)` Parámetros que definen los objetos que se exportarán y el formato a utilizar.

Parameter	Type	Description
format	string	Formato en el que se deben exportar los datos. Valores posibles: `yaml` - YAML; `xml` - XML; `json` - JSON; `raw` - matriz PHP sin procesar. Comportamiento del parámetro: - *required*
prettyprint	boolean	Hace que la salida sea más legible para las personas al añadir sangría. Valores posibles: `true` - añadir sangría; `false` - *(default)* no añadir sangría.

Parameter	Type	Description
options	object	Objetos que se exportarán. El objeto `options` tiene los siguientes parámetros: `host_groups` - `(array)` IDs de grupos de host para exportar; `hosts` - `(array)` IDs de host para exportar; `images` - `(array)` IDs de imágenes para exportar; `maps` - `(array)` IDs de mapas para exportar; `mediaTypes` - `(array)` IDs de tipos de medios para exportar; `template_groups` - `(array)` IDs de grupos de template para exportar; `templates` - `(array)` IDs de templates para exportar; `dashboards` - `(array)` IDs de paneles para exportar; `global_regexes` - `(array)` IDs de expresiones regulares globales para exportar. Los usuarios de tipo *Admin* y *User* pueden exportar solo aquellos objetos para los que tienen permiso de solo lectura o lectura y escritura, así como imágenes, pero no tipos de medios. Comportamiento del parámetro: - *required*.

Valores de retorno

`(string)` Devuelve una cadena serializada que contiene los datos de configuración solicitados.

Ejemplos

Exportando una plantilla

Exportar la configuración de la plantilla "10571" como una cadena XML.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "configuration.export",
    "params": {
        "options": {
            "templates": [
                "10571"
            ]
        },
        "format": "xml"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": "<?xml version=\"1.0\" encoding=\"UTF-8\"?>\n<zabbix_export><version>8.0</version><template_
```

```
    "id": 1
}
```

Fuente

CConfiguration::export() en *ui/include/classes/api/services/CConfiguration.php*.

configuration.import

Descripción

`boolean configuration.import(object parameters)`

Este método permite importar datos de configuración desde una cadena serializada.

Note:
Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object)` Parámetros que contienen los datos a importar y las reglas sobre cómo deben tratarse los datos.

Parameter	Type	Description
format	string	Formato de la cadena serializada. Valores posibles: `yaml` - YAML; `xml` - XML; `json` - JSON. Comportamiento del parámetro: - *required*
source	string	Cadena serializada que contiene los datos de configuración. Comportamiento del parámetro: - *required*
rules	object	Reglas sobre cómo deben importarse los objetos nuevos y existentes. Los usuarios de tipo *Admin* solo pueden importar aquellos objetos para los que tengan permiso de lectura y escritura, así como los mapas. Por ejemplo, un host y sus entidades (items, triggers, gráficos, etc.) solo pueden importarse si el grupo de usuarios del usuario tiene permiso sobre el grupo de hosts al que pertenecerá el host importado. Las imágenes y los tipos de medios no pueden ser importados por usuarios de tipo *Admin*. El parámetro `rules` se describe en detalle en la tabla siguiente. Comportamiento del parámetro: - *required*
returnMissingObjects	boolean	Reglas sobre si el objeto de respuesta debe contener información acerca de los objetos faltantes; valor predeterminado: `false`.

Attention:
Si no se proporcionan reglas, la configuración no se actualizará.

El objeto `rules` admite los siguientes parámetros.

Parameter	Type	Description
dashboards	object	Reglas sobre cómo importar dashboards. Parámetros admitidos: `createMissing` - `(boolean)` si se establece en `true`, se crearán nuevos dashboards; valor predeterminado: `false`; `updateExisting` - `(boolean)` si se establece en `true` y un dashboard importable concreto está presente en el sistema, se actualizará por completo. Las páginas y los widgets se reescribirán por completo con los nuevos; valor predeterminado: `false`.
discoveryRules	object	Reglas sobre cómo importar reglas de LLD. Parámetros admitidos: `createMissing` - `(boolean)` si se establece en `true`, se crearán nuevas reglas de LLD; valor predeterminado: `false`; `updateExisting` - `(boolean)` si se establece en `true`, las reglas de LLD existentes se actualizarán; valor predeterminado: `false`; `deleteMissing` - `(boolean)` si se establece en `true`, las reglas de LLD que no estén presentes en los datos importados se eliminarán de la base de datos; valor predeterminado: `false`.
global_regexes	object	Reglas sobre cómo importar expresiones regulares globales. Parámetros admitidos: `createMissing` - `(boolean)` si se establece en `true`, se crearán nuevas expresiones regulares globales; valor predeterminado: `false`; `updateExisting` - `(boolean)` si se establece en `true`, las expresiones regulares globales existentes se actualizarán, sobrescribiendo por completo sus subexpresiones; valor predeterminado: `false`.
graphs	object	Reglas sobre cómo importar gráficos. Parámetros admitidos: `createMissing` - `(boolean)` si se establece en `true`, se crearán nuevos gráficos; valor predeterminado: `false`; `updateExisting` - `(boolean)` si se establece en `true`, los gráficos existentes se actualizarán; valor predeterminado: `false`; `deleteMissing` - `(boolean)` si se establece en `true`, los gráficos que no estén presentes en los datos importados se eliminarán de la base de datos; valor predeterminado: `false`.
host_groups	object	Reglas sobre cómo importar grupos de hosts. Parámetros admitidos: `createMissing` - `(boolean)` si se establece en `true`, se crearán nuevos grupos de hosts; valor predeterminado: `false`; `updateExisting` - `(boolean)` si se establece en `true`, los grupos de hosts existentes se actualizarán; valor predeterminado: `false`.
hosts	object	Reglas sobre cómo importar hosts. Parámetros admitidos: `createMissing` - `(boolean)` si se establece en `true`, se crearán nuevos hosts; valor predeterminado: `false`; `updateExisting` - `(boolean)` si se establece en `true`, los hosts existentes se actualizarán; valor predeterminado: `false`.

Parameter	Type	Description
httptests	object	Reglas sobre cómo importar escenarios web. Parámetros admitidos: `createMissing` - `(boolean)` si se establece en `true`, se crearán nuevos escenarios web; valor predeterminado: `false`; `updateExisting` - `(boolean)` si se establece en `true`, los escenarios web existentes se actualizarán; valor predeterminado: `false`; `deleteMissing` - `(boolean)` si se establece en `true`, los escenarios web que no estén presentes en los datos importados se eliminarán de la base de datos; valor predeterminado: `false`.
images	object	Reglas sobre cómo importar imágenes. Parámetros admitidos: `createMissing` - `(boolean)` si se establece en `true`, se crearán nuevas imágenes; valor predeterminado: `false`; `updateExisting` - `(boolean)` si se establece en `true`, las imágenes existentes se actualizarán; valor predeterminado: `false`.
items	object	Reglas sobre cómo importar items. Parámetros admitidos: `createMissing` - `(boolean)` si se establece en `true`, se crearán nuevos items; valor predeterminado: `false`; `updateExisting` - `(boolean)` si se establece en `true`, los items existentes se actualizarán; valor predeterminado: `false`; `deleteMissing` - `(boolean)` si se establece en `true`, los items que no estén presentes en los datos importados se eliminarán de la base de datos; valor predeterminado: `false`.
maps	object	Reglas sobre cómo importar mapas. Parámetros admitidos: `createMissing` - `(boolean)` si se establece en `true`, se crearán nuevos mapas; valor predeterminado: `false`; `updateExisting` - `(boolean)` si se establece en `true`, los mapas existentes se actualizarán; valor predeterminado: `false`.
mediaTypes	object	Reglas sobre cómo importar tipos de medios. Parámetros admitidos: `createMissing` - `(boolean)` si se establece en `true`, se crearán nuevos tipos de medios; valor predeterminado: `false`; `updateExisting` - `(boolean)` si se establece en `true`, los tipos de medios existentes se actualizarán; valor predeterminado: `false`.
templateDashboards	object	Reglas sobre cómo importar dashboards de template. Parámetros admitidos: `createMissing` - `(boolean)` si se establece en `true`, se crearán nuevos dashboards de template; valor predeterminado: `false`; `updateExisting` - `(boolean)` si se establece en `true`, los dashboards de template existentes se actualizarán; valor predeterminado: `false`; `deleteMissing` - `(boolean)` si se establece en `true`, los dashboards de template que no estén presentes en los datos importados se eliminarán de la base de datos; valor predeterminado: `false`.
template_groups	object	Reglas sobre cómo importar grupos de template. Parámetros admitidos: `createMissing` - `(boolean)` si se establece en `true`, se crearán nuevos grupos de template; valor predeterminado: `false`; `updateExisting` - `(boolean)` si se establece en `true`, los grupos de template existentes se actualizarán; valor predeterminado: `false`.

Parameter	Type	Description
templateLinkage	object	Reglas sobre cómo importar enlaces de template. Parámetros admitidos: `createMissing` - `(boolean)` si se establece en `true`, los templates que no estén enlazados al host o template que se está importando, pero que estén presentes en los datos importados, se enlazarán; valor predeterminado: `false`; `deleteMissing` - `(boolean)` si se establece en `true`, los templates que estén enlazados al host o template que se está importando, pero que no estén presentes en los datos importados, se desvincularán sin eliminar las entidades (items, triggers, etc.) heredadas de los templates desvinculados; valor predeterminado: `false`.
templates	object	Reglas sobre cómo importar templates. Parámetros admitidos: `createMissing` - `(boolean)` si se establece en `true`, se crearán nuevos templates; valor predeterminado: `false`; `updateExisting` - `(boolean)` si se establece en `true`, los templates existentes se actualizarán; valor predeterminado: `false`.
triggers	object	Reglas sobre cómo importar triggers. Parámetros admitidos: `createMissing` - `(boolean)` si se establece en `true`, se crearán nuevos triggers; valor predeterminado: `false`; `updateExisting` - `(boolean)` si se establece en `true`, los triggers existentes se actualizarán; valor predeterminado: `false`; `deleteMissing` - `(boolean)` si se establece en `true`, los triggers que no estén presentes en los datos importados se eliminarán de la base de datos; valor predeterminado: `false`.
valueMaps	object	Reglas sobre cómo importar mapas de valores de host o template. Parámetros admitidos: `createMissing` - `(boolean)` si se establece en `true`, se crearán nuevos mapas de valores; valor predeterminado: `false`; `updateExisting` - `(boolean)` si se establece en `true`, los mapas de valores existentes se actualizarán; valor predeterminado: `false`; `deleteMissing` - `(boolean)` si se establece en `true`, los mapas de valores que no estén presentes en los datos importados se eliminarán de la base de datos; valor predeterminado: `false`.

Valores de retorno

`(boolean)` Devuelve `true` si la importación se ha realizado correctamente.

Ejemplos

Importando una plantilla

Importa la configuración de la plantilla contenida en la cadena XML. Si faltan métricas o disparadores en la cadena XML, se eliminarán de la base de datos y todo lo demás permanecerá sin cambios.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "configuration.import",
    "params": {
        "format": "xml",
        "rules": {
            "templates": {
                "createMissing": true,
                "updateExisting": true
            },
            "items": {
```

```
                "createMissing": true,
                "updateExisting": true,
                "deleteMissing": true
            },
            "triggers": {
                "createMissing": true,
                "updateExisting": true,
                "deleteMissing": true
            },
            "valueMaps": {
                "createMissing": true,
                "updateExisting": false
            }
        },
        "source": "<?xml version=\"1.0\" encoding=\"UTF-8\"?>\n<zabbix_export><version>8.0</version><templ
    },
    "id": 1
}
```


Respuesta:


```
{
    "jsonrpc": "2.0",
    "result": true,
    "id": 1
}
```


Fuente

CConfiguration::import() en *ui/include/classes/api/services/CConfiguration.php*.

configuration.importcompare

Descripción

`array configuration.importcompare(object parameters)`

Este método permite comparar el archivo de importación con los elementos actuales del sistema y muestra qué se cambiará si este archivo de importación se importa.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración del rol de usuario. Consulte User roles para obtener más información.

Parámetros

`(object)` Parámetros que contienen los posibles datos a importar y las reglas sobre cómo deben tratarse los datos.

Parameter	Type	Description
format	string	Formato de la cadena serializada. Valores posibles: `yaml` - YAML; `xml` - XML; `json` - JSON. Comportamiento del parámetro: - *required*
source	string	Cadena serializada que contiene los datos de configuración. Comportamiento del parámetro: - *required*

Parameter	Type	Description
rules	object	Reglas sobre cómo deben compararse los objetos nuevos y existentes. Los usuarios de tipo *Admin* y *User* pueden comparar objetos nuevos solo con los existentes para los que tengan permiso de *solo lectura* o *lectura y escritura* permission. El parámetro `rules` se describe en detalle en la tabla siguiente. Comportamiento del parámetro: - *required*
returnMissingObjects	boolean	Reglas sobre si el objeto de respuesta debe contener información acerca de los objetos faltantes; valor predeterminado: `false`.

Attention:
Si no se proporcionan reglas, no habrá nada que actualizar y el resultado estará vacío.

Note:
La comparación se realizará solo para grupos de host y templates. Los triggers y los gráficos se compararán solo para los templates importados; cualquier otro se considerará como ”nuevo”.

El objeto `rules` admite los siguientes parámetros.

Parameter	Type	Description
dashboards	object	Reglas sobre cómo importar dashboards. Parámetros admitidos: `createMissing` - `(boolean)` si se establece en `true`, se crearán nuevos dashboards; valor predeterminado: `false`; `updateExisting` - `(boolean)` si se establece en `true` y un dashboard importable concreto está presente en el sistema, se actualizará por completo. Las páginas y los widgets se sobrescribirán por completo con los nuevos; valor predeterminado: `false`.
discoveryRules	object	Reglas sobre cómo importar reglas LLD. Parámetros admitidos: `createMissing` - `(boolean)` si se establece en `true`, se crearán nuevas reglas LLD; valor predeterminado: `false`; `updateExisting` - `(boolean)` si se establece en `true`, se actualizarán las reglas LLD existentes; valor predeterminado: `false`; `deleteMissing` - `(boolean)` si se establece en `true`, las reglas LLD que no estén presentes en los datos importados se eliminarán de la base de datos; valor predeterminado: `false`.
global_regexes	object	Reglas sobre cómo importar expresiones regulares globales. Parámetros admitidos: `createMissing` - `(boolean)` si se establece en `true`, se crearán nuevas expresiones regulares globales; valor predeterminado: `false`; `updateExisting` - `(boolean)` si se establece en `true`, se actualizarán las expresiones regulares globales existentes, sobrescribiendo por completo sus subexpresiones; valor predeterminado: `false`.

Parameter	Type	Description
graphs	object	Reglas sobre cómo importar gráficos. Parámetros admitidos: `createMissing` - `(boolean)` si se establece en `true`, se crearán nuevos gráficos; valor predeterminado: `false`; `updateExisting` - `(boolean)` si se establece en `true`, se actualizarán los gráficos existentes; valor predeterminado: `false`; `deleteMissing` - `(boolean)` si se establece en `true`, los gráficos que no estén presentes en los datos importados se eliminarán de la base de datos; valor predeterminado: `false`.
host_groups	object	Reglas sobre cómo importar grupos de host. Parámetros admitidos: `createMissing` - `(boolean)` si se establece en `true`, se crearán nuevos grupos de host; valor predeterminado: `false`; `updateExisting` - `(boolean)` si se establece en `true`, se actualizarán los grupos de host existentes; valor predeterminado: `false`.
httptests	object	Reglas sobre cómo importar escenarios web. Parámetros admitidos: `createMissing` - `(boolean)` si se establece en `true`, se crearán nuevos escenarios web; valor predeterminado: `false`; `updateExisting` - `(boolean)` si se establece en `true`, se actualizarán los escenarios web existentes; valor predeterminado: `false`; `deleteMissing` - `(boolean)` si se establece en `true`, los escenarios web que no estén presentes en los datos importados se eliminarán de la base de datos; valor predeterminado: `false`.
images	object	Reglas sobre cómo importar imágenes. Parámetros admitidos: `createMissing` - `(boolean)` si se establece en `true`, se crearán nuevas imágenes; valor predeterminado: `false`; `updateExisting` - `(boolean)` si se establece en `true`, se actualizarán las imágenes existentes; valor predeterminado: `false`. Este parámetro no hará ninguna diferencia en la salida. Solo se permite por coherencia con `configuration.import`.
items	object	Reglas sobre cómo importar items. Parámetros admitidos: `createMissing` - `(boolean)` si se establece en `true`, se crearán nuevos items; valor predeterminado: `false`; `updateExisting` - `(boolean)` si se establece en `true`, se actualizarán los items existentes; valor predeterminado: `false`; `deleteMissing` - `(boolean)` si se establece en `true`, los items que no estén presentes en los datos importados se eliminarán de la base de datos; valor predeterminado: `false`.
maps	object	Reglas sobre cómo importar mapas. Parámetros admitidos: `createMissing` - `(boolean)` si se establece en `true`, se crearán nuevos mapas; valor predeterminado: `false`; `updateExisting` - `(boolean)` si se establece en `true`, se actualizarán los mapas existentes; valor predeterminado: `false`. Este parámetro no hará ninguna diferencia en la salida. Solo se permite por coherencia con `configuration.import`.

Parameter	Type	Description
mediaTypes	object	Reglas sobre cómo importar tipos de medios. Parámetros admitidos: `createMissing` - `(boolean)` si se establece en `true`, se crearán nuevos tipos de medios; valor predeterminado: `false`; `updateExisting` - `(boolean)` si se establece en `true`, se actualizarán los tipos de medios existentes; valor predeterminado: `false`. Este parámetro no hará ninguna diferencia en la salida. Solo se permite por coherencia con `configuration.import`.
templateDashboards	object	Reglas sobre cómo importar dashboards de template. Parámetros admitidos: `createMissing` - `(boolean)` si se establece en `true`, se crearán nuevos dashboards de template; valor predeterminado: `false`; `updateExisting` - `(boolean)` si se establece en `true`, se actualizarán los dashboards de template existentes; valor predeterminado: `false`; `deleteMissing` - `(boolean)` si se establece en `true`, los dashboards de template que no estén presentes en los datos importados se eliminarán de la base de datos; valor predeterminado: `false`.
template_groups	object	Reglas sobre cómo importar grupos de template. Parámetros admitidos: `createMissing` - `(boolean)` si se establece en `true`, se crearán nuevos grupos de template; valor predeterminado: `false`; `updateExisting` - `(boolean)` si se establece en `true`, se actualizarán los grupos de template existentes; valor predeterminado: `false`.
templateLinkage	object	Reglas sobre cómo importar enlaces de template. Parámetros admitidos: `createMissing` - `(boolean)` si se establece en `true`, los templates que no estén enlazados al host o template que se está importando, pero que estén presentes en los datos importados, se enlazarán; valor predeterminado: `false`; `deleteMissing` - `(boolean)` si se establece en `true`, los templates que estén enlazados al host o template que se está importando, pero que no estén presentes en los datos importados, se desvincularán sin eliminar las entidades (items, triggers, etc.) heredadas de los templates desvinculados; valor predeterminado: `false`.
templates	object	Reglas sobre cómo importar templates. Parámetros admitidos: `createMissing` - `(boolean)` si se establece en `true`, se crearán nuevos templates; valor predeterminado: `false`; `updateExisting` - `(boolean)` si se establece en `true`, se actualizarán los templates existentes; valor predeterminado: `false`.
triggers	object	Reglas sobre cómo importar triggers. Parámetros admitidos: `createMissing` - `(boolean)` si se establece en `true`, se crearán nuevos triggers; valor predeterminado: `false`; `updateExisting` - `(boolean)` si se establece en `true`, se actualizarán los triggers existentes; valor predeterminado: `false`; `deleteMissing` - `(boolean)` si se establece en `true`, los triggers que no estén presentes en los datos importados se eliminarán de la base de datos; valor predeterminado: `false`.

Parameter	Type	Description
valueMaps	object	Reglas sobre cómo importar mapas de valores de host o template. Parámetros admitidos: `createMissing` - `(boolean)` si se establece en `true`, se crearán nuevos mapas de valores; valor predeterminado: `false`; `updateExisting` - `(boolean)` si se establece en `true`, se actualizarán los mapas de valores existentes; valor predeterminado: `false`; `deleteMissing` - `(boolean)` si se establece en `true`, los mapas de valores que no estén presentes en los datos importados se eliminarán de la base de datos; valor predeterminado: `false`.

Valores de retorno

`(array)` Devuelve un array con los cambios en la configuración que se realizarán.

Ejemplos

Comparar la importación de una template

Compara la template contenida en la cadena XML con los elementos actuales del sistema y muestra lo que se cambiará si se importa esta template.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "configuration.importcompare",
    "params": {
        "format": "xml",
        "rules": {
            "discoveryRules": {
                "createMissing": true,
                "updateExisting": true,
                "deleteMissing": true
            },
            "graphs": {
                "createMissing": true,
                "updateExisting": true,
                "deleteMissing": true
            },
            "host_groups": {
                "createMissing": true,
                "updateExisting": true
            },
            "template_groups": {
                "createMissing": true,
                "updateExisting": true
            },
            "httptests": {
                "createMissing": true,
                "updateExisting": true,
                "deleteMissing": true
            },
            "items": {
                "createMissing": true,
                "updateExisting": true,
                "deleteMissing": true
            },
            "templateLinkage": {
                "createMissing": true,
                "deleteMissing": true
            },
            "templates": {
```

```
                "createMissing": true,
                "updateExisting": true
            },
            "templateDashboards": {
                "createMissing": true,
                "updateExisting": true,
                "deleteMissing": true
            },
            "triggers": {
                "createMissing": true,
                "updateExisting": true,
                "deleteMissing": true
            },
            "valueMaps": {
                "createMissing": true,
                "updateExisting": true,
                "deleteMissing": true
            }
        },
        "source": "<?xml version=\"1.0\" encoding=\"UTF-8\"?>\n<zabbix_export><version>8.0</version><templ
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templates": {
            "updated": [
                {
                    "before": {
                        "uuid": "5aef0444a82a4d8cb7a95dc4c0c85330",
                        "template": "New template",
                        "name": "New template",
                        "groups": [
                            {
                                "name": "Templates"
                            }
                        ]
                    },
                    "after": {
                        "uuid": "5aef0444a82a4d8cb7a95dc4c0c85330",
                        "template": "New template",
                        "name": "New template",
                        "groups": [
                            {
                                "name": "Templates"
                            }
                        ]
                    },
                    "items": {
                        "added": [
                            {
                                "after": {
                                    "uuid": "648006da5971424ead0c47ddbbf1ea2e",
                                    "name": "CPU utilization",
                                    "key": "system.cpu.util",
                                    "value_type": "FLOAT",
                                    "units": "%"
                                },
                                "triggers": {
```

```
                                "added": [
                                    {
                                        "after": {
                                            "uuid": "736225012c534ec480c2a66a91322ce0",
                                            "expression": "avg(/New template/system.cpu.util,3m)>70",
                                            "name": "CPU utilization too high on 'New host' for 3 minu
                                            "priority": "WARNING"
                                        }
                                    }
                                ]
                            }
                        }
                    ],
                    "removed": [
                        {
                            "before": {
                                "uuid": "6805d4c39a624a8bab2cc8ab63df1ab3",
                                "name": "CPU load",
                                "key": "system.cpu.load",
                                "value_type": "FLOAT"
                            },
                            "triggers": {
                                "removed": [
                                    {
                                        "before": {
                                            "uuid": "ab4c2526c2bc42e48a633082255ebcb3",
                                            "expression": "avg(/New template/system.cpu.load,3m)>2",
                                            "name": "CPU load too high on 'New host' for 3 minutes",
                                            "priority": "WARNING"
                                        }
                                    }
                                ]
                            }
                        }
                    ],
                    "updated": [
                        {
                            "before": {
                                "uuid": "7f1e6f1e48aa4a128e5b6a958a5d11c3",
                                "name": "Zabbix agent ping",
                                "key": "agent.ping"
                            },
                            "after": {
                                "uuid": "7f1e6f1e48aa4a128e5b6a958a5d11c3",
                                "name": "Zabbix agent ping",
                                "key": "agent.ping",
                                "delay": "3m"
                            }
                        }
                    ]
                }
            }
        ]
    }
},
"id": 1
}
```


Fuente

CConfiguration::importcompare() en *ui/include/classes/api/services/CConfiguration.php*.

Correlación

Esta clase está diseñada para trabajar con correlaciones.

Referencias de objetos:

- Correlación
- Operación de correlación
- Filtro de correlación
  - Condición de filtro de correlación

Métodos disponibles:

- correlation.create - crear nuevas correlaciones
- correlation.delete - eliminar correlaciones
- correlation.get - obtener correlaciones
- correlation.update - actualizar correlaciones

Objeto correlación

Los siguientes objetos están directamente relacionados con la API de `correlación`.

Correlación

El objeto correlación tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
correlationid	ID	ID de la correlación. Comportamiento de la propiedad: - *sólo lectura* - *requerido* para operaciones de actualización
name	string	Nombre de la correlación. Comportamiento de la propiedad: - *requerido* para operaciones de creación
description	string	Descripción de la correlación.
status	integer	Indica si la correlación está habilitada o deshabilitada. Valores posibles: 0 - *(por defecto)* habilitada; 1 - deshabilitada.

Operación de correlación

El objeto de operación de correlación define una operación que se realizará cuando se ejecute una correlación. Tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
type	integer	Tipo de operación. Valores posibles: 0 - cerrar eventos antiguos; 1 - cerrar nuevo evento. Comportamiento de la propiedad: - *obligatorio*

Filtro de correlación

El objeto de filtro de correlación define un conjunto de condiciones que deben cumplirse para realizar las operaciones de correlación configuradas. Tiene las siguientes propiedades.

Property	Type	Description
conditions	array	Conjunto de condiciones de filtro que se usarán para filtrar resultados. Las condiciones se ordenarán según su posición en la fórmula. Property behavior: - *required*
evaltype	integer	Método de evaluación de las condiciones del filtro. Valores posibles: 0 - And/Or; 1 - And; 2 - Or; 3 - Expresión personalizada. Property behavior: - *required*
eval_formula	string	Expresión generada que se usará para evaluar las condiciones del filtro. La expresión contiene IDs que hacen referencia a condiciones de filtro específicas mediante su `formulaid`. El valor de `eval_formula` es igual al valor de `formula` para filtros con una expresión personalizada. Property behavior: - *read-only*
formula	string	Expresión definida por el usuario que se usará para evaluar las condiciones de los filtros con una expresión personalizada. La expresión debe contener IDs que hagan referencia a condiciones de filtro específicas mediante su `formulaid`. Los IDs usados en la expresión deben coincidir exactamente con los definidos en las condiciones del filtro: no puede quedar ninguna condición sin usar ni omitirse ninguna. Property behavior: - *required* si `evaltype` está establecido en ”custom expression”

Condición de filtro de correlación

El objeto de condición de filtro de correlación define una condición específica que debe comprobarse antes de ejecutar las operaciones de correlación.

Propiedad	Tipo	Descripción
type	integer	Tipo de condición. Valores posibles: 0 - etiqueta de evento anterior; 1 - etiqueta de evento nueva; 2 - grupo de hosts del evento nuevo; 3 - par de etiquetas de evento; 4 - valor de la etiqueta de evento anterior; 5 - valor de la etiqueta de evento nuevo. Comportamiento de la propiedad: - *obligatorio*
tag	string	Etiqueta de evento (anterior o nueva). Comportamiento de la propiedad: - *obligatorio* si `type` está establecido en ”etiqueta de evento anterior”, ”etiqueta de evento nueva”, ”valor de la etiqueta de evento anterior” o ”valor de la etiqueta de evento nuevo”

Propiedad	Tipo	Descripción
groupid	ID	ID del grupo de hosts. Comportamiento de la propiedad: - *obligatorio* si `type` está establecido en ”grupo de hosts del evento nuevo”
oldtag	string	Etiqueta de evento anterior. Comportamiento de la propiedad: - *obligatorio* si `type` está establecido en ”par de etiquetas de evento”
newtag	string	Etiqueta de evento anterior. Comportamiento de la propiedad: - *obligatorio* si `type` está establecido en ”par de etiquetas de evento”
value	string	Valor de la etiqueta de evento (anterior o nueva). Comportamiento de la propiedad: - *obligatorio* si `type` está establecido en ”valor de la etiqueta de evento anterior” o ”valor de la etiqueta de evento nuevo”
formulaid	string	ID único arbitrario que se utiliza para hacer referencia a la condición desde una expresión personalizada. Solo puede contener letras mayúsculas. El ID debe ser definido por el usuario al modificar las condiciones del filtro, pero se generará de nuevo al solicitarlas posteriormente.
operator	integer	Operador de condición. Comportamiento de la propiedad: - *obligatorio* si `type` está establecido en ”grupo de hosts del evento nuevo”, ”valor de la etiqueta de evento anterior” o ”valor de la etiqueta de evento nuevo”

Note:
Para comprender mejor cómo utilizar los filtros con varios tipos de expresiones, consulte los ejemplos en las páginas de los métodos correlation.get y correlation.create.

Los siguientes operadores y valores son compatibles para cada tipo de condición.

Condición	Nombre de la condición	Operadores soportados	Valor esperado
2	Grupo de hosts	=, <>	ID del grupo de hosts.
4	Valor de la etiqueta de evento anterior	=, <>, like, not like	string
5	Valor de la etiqueta de evento nuevo	=, <>, like, not like	string

correlation.create

Descripción

`object correlation.create(object/array correlations)`

Este método permite crear nuevas correlaciones.

Note:
Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object/array)` Correlaciones a crear.

Además de las propiedades estándar de correlación, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
operations	array	Operaciones de correlación a crear para la correlación. Comportamiento del parámetro: - *obligatorio*
filter	object	Objeto filtro de correlación para la correlación. Comportamiento del parámetro: - *obligatorio*

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las correlaciones creadas bajo la propiedad `correlationids`. El orden de los IDs devueltos coincide con el orden de las correlaciones enviadas.

Ejemplos

Crear una nueva correlación de etiquetas de evento

Cree una correlación utilizando el método de evaluación `AND/OR` con una condición y una operación. Por defecto, la correlación estará habilitada.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "correlation.create",
    "params": {
        "name": "new event tag correlation",
        "filter": {
            "evaltype": 0,
            "conditions": [
                {
                    "type": 1,
                    "tag": "ok"
                }
            ]
        },
        "operations": [
            {
                "type": 0
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "correlationids": [
            "1"
        ]
    },
    "id": 1
}
```

Usar un filtro de expresión personalizada

Cree una correlación que utilice una condición de filtro personalizada. Los IDs de fórmula ”A” o ”B” se han elegido arbitrariamente. El tipo de condición será ”Grupo de equipos” con el operador ”<>”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "correlation.create",
    "params": {
        "name": "nueva correlación de grupo de equipos",
        "description": "una descripción personalizada",
        "status": 0,
        "filter": {
            "evaltype": 3,
            "formula": "A or B",
            "conditions": [
                {
                    "type": 2,
                    "operator": 1,
                    "formulaid": "A"
                },
                {
                    "type": 2,
                    "operator": 1,
                    "formulaid": "B"
                }
            ]
        },
        "operations": [
            {
                "type": 1
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "correlationids": [
            "2"
        ]
    },
    "id": 1
}
```

Ver también

- Filtro de correlación
- Operación de correlación

Fuente

CCorrelation::create() en *ui/include/classes/api/services/CCorrelation.php*.

correlation.delete

Descripción

`object correlation.delete(array correlationids)`

Este método le permite eliminar correlaciones.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración del rol de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(array)` IDs de las correlaciones a eliminar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las correlaciones eliminadas bajo la propiedad `correlationids`.

Ejemplo

Eliminar múltiples correlaciones

Eliminar dos correlaciones.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "correlation.delete",
    "params": [
        "1",
        "2"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "correlationids": [
            "1",
            "2"
        ]
    },
    "id": 1
}
```

Fuente

CCorrelation::delete() en *ui/include/classes/api/services/CCorrelation.php*.

correlation.get

Descripción

`integer/array correlation.get(object parameters)`

El método permite recuperar correlaciones según los parámetros proporcionados.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(objeto)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
correlationids	ID/array	Devolver solo las correlaciones con los IDs dados.
selectFilter	query	Devolver una propiedad `filter` con las condiciones de correlación.
selectOperations	query	Devolver una propiedad `operations` con las operaciones de correlación.
sortfield	string/array	Ordenar el resultado por las propiedades dadas. Valores posibles: `correlationid`, `name`, `status`.

Parámetro	Tipo	Descripción
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperar correlaciones

Recupera todas las correlaciones configuradas junto con las condiciones y operaciones de correlación. El filtro utiliza el tipo de evaluación "and/or", por lo que la propiedad `formula` está vacía y `eval_formula` se genera automáticamente.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "correlation.get",
    "params": {
        "output": "extend",
        "selectOperations": "extend",
        "selectFilter": "extend"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "correlationid": "1",
            "name": "Correlation 1",
            "description": "",
            "status": "0",
            "filter": {
                "evaltype": "0",
                "formula": "",
                "conditions": [
                    {
                        "type": "3",
                        "oldtag": "error",
                        "newtag": "ok",
                        "formulaid": "A"
                    }
                ],
                "eval_formula": "A"
            },
            "operations": [
                {
```

```
                    "type": "0"
                }
            ]
        }
    ],
    "id": 1
}
```

Ver también

- Filtro de correlación
- Operación de correlación

Fuente

CCorrelation::get() en *ui/include/classes/api/services/CCorrelation.php*.

correlation.update

Descripción

`object correlation.update(object/array correlations)`

Este método le permite actualizar correlaciones existentes.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración del rol de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(objeto/array)` Propiedades de correlación que se actualizarán.

La propiedad `correlationid` debe definirse para cada correlación, todas las demás propiedades son opcionales. Solo se actualizarán las propiedades pasadas, todas las demás permanecerán sin cambios.

Además de las propiedades estándar de correlación, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
filter	objeto	Objeto Correlation filter para reemplazar el filtro actual.
operations	array	Correlation operations para reemplazar las operaciones existentes.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las correlaciones actualizadas bajo la propiedad `correlationids`.

Ejemplos

Deshabilitar correlación

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "correlation.update",
    "params": {
        "correlationid": "1",
        "status": "1"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "correlationids": [
```

```
            "1"
        ]
    },
    "id": 1
}
```

Reemplazar condiciones, pero mantener el método de evaluación

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "correlation.update",
    "params": {
        "correlationid": "1",
        "filter": {
            "conditions": [
                {
                    "type": 3,
                    "oldtag": "error",
                    "newtag": "ok"
                }
            ]
        }
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "correlationids": [
            "1"
        ]
    },
    "id": 1
}
```

Ver también

- Filtro de correlación
- Operación de correlación

Fuente

CCorrelation::update() en *ui/include/classes/api/services/CCorrelation.php*.

Dashboard

Esta clase está diseñada para trabajar con dashboards.

Referencias de objetos:

- Dashboard
- Página de dashboard
  - Widget de dashboard
    - Campo de widget de dashboard
- Grupo de usuarios de dashboard
- Usuario de dashboard

Métodos disponibles:

- dashboard.create - crear nuevos dashboards
- dashboard.delete - eliminar dashboards
- dashboard.get - obtener dashboards
- dashboard.update - actualizar dashboards

Objeto dashboard

Los siguientes objetos están directamente relacionados con la API de `dashboard`.

Dashboard

El objeto dashboard tiene las siguientes propiedades.

Property	Type	Description
dashboardid	ID	ID del dashboard. Property behavior: - *solo lectura* - *obligatorio* para operaciones de actualización
name	string	Nombre del dashboard. Property behavior: - *obligatorio* para operaciones de creación
userid	ID	ID del usuario que es el propietario del dashboard.
private	integer	Tipo de compartición del dashboard. Valores posibles: 0 - dashboard público; 1 - *(predeterminado)* dashboard privado.
display_period	integer	Intervalo predeterminado de la presentación de diapositivas (en segundos). Valores posibles: 10, 30, 60, 120, 600, 1800, 3600. Predeterminado: 30.
auto_start	integer	Inicio automático de la presentación de diapositivas. Valores posibles: 0 - no iniciar automáticamente la presentación de diapositivas; 1 - *(predeterminado)* iniciar automáticamente la presentación de diapositivas.

Página de dashboard

El objeto de página de dashboard tiene las siguientes propiedades.

Property	Type	Description
dashboard_pageid	ID	ID de la página de dashboard. Property behavior: - *read-only*
name	string	Nombre de la página de dashboard. Default: cadena vacía.
display_period	integer	Intervalo de la presentación de diapositivas del dashboard (en segundos). Valores posibles: 0, 10, 30, 60, 120, 600, 1800, 3600. Default: 0 (usará el intervalo de presentación de diapositivas predeterminado).
widgets	array	Matriz de objetos dashboard widget.

Widget de dashboard

El objeto widget de dashboard tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
widgetid	ID	ID del widget de dashboard. Comportamiento de la propiedad: - *solo lectura*
type	string	Tipo de widget de dashboard. Valores posibles: actionlog - Registro de acciones; clock - Reloj; discovery - Estado del descubrimiento; favgraphs - Gráficas favoritas; favmaps - Mapas favoritos; gauge - Medidor; geomap - Geomapa; graph - Gráfica (clásica); graphprototype - Prototipo de gráfica; honeycomb - Panal; hostavail - Disponibilidad de host; hostcard - Tarjeta de host; hostnavigator - Navegador de host; itemcard - Tarjeta de item; itemhistory - Historial de item; itemnavigator - Navegador de item; item - Valor de item; map - Mapa; navtree - Árbol de navegación de mapas; piechart - Gráfica circular; problemhosts - Hosts con problemas; problems - Problemas; problemsbysv - Problemas por severidad; scatterplot - Diagrama de dispersión; slareport - Informe de SLA; svggraph - Gráfica; systeminfo - Información del sistema; tophosts - Hosts principales; topitems - Items principales; toptriggers - Triggers principales; trigover - Resumen de triggers; url - URL; web - Monitorización web. Comportamiento de la propiedad: - *requerido*
name	string	Nombre personalizado del widget.
x	integer	Posición horizontal desde el lado izquierdo del dashboard. Los valores posibles van de 0 a 71.
y	integer	Posición vertical desde la parte superior del dashboard. Los valores posibles van de 0 a 63.
width	integer	El ancho del widget. Los valores posibles van de 1 a 72.
height	integer	La altura del widget. Los valores posibles van de 1 a 64.
view_mode	integer	El modo de visualización del widget. Valores posibles: 0 - *(por defecto)* vista por defecto del widget; 1 - con cabecera oculta;

Propiedad	Tipo	Descripción
fields	array	Array de objetos campo de widget de dashboard. Comportamiento de la propiedad: - ver widgets individuales en Campos de widget de dashboard

Campo de widget de dashboard

El objeto campo de widget de dashboard tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
type	integer	Tipo de campo de widget. Valores posibles: 0 - Integer; 1 - String; 2 - Grupo de hosts; 3 - Host; 4 - Item; 5 - Prototipo de item; 6 - Gráfico; 7 - Prototipo de gráfico; 8 - Mapa; 9 - Servicio; 10 - SLA; 11 - Usuario; 12 - Acción; 13 - Tipo de medio. Comportamiento de la propiedad: - *requerido*
name	string	Nombre del campo de widget. Valores posibles: ver Campos de widget de dashboard. Comportamiento de la propiedad: - *requerido*
value	mixed	Valor del campo de widget dependiendo del tipo. Valores posibles: ver Campos de widget de dashboard. Comportamiento de la propiedad: - *requerido*

Grupo de usuarios del dashboard

Lista de permisos del dashboard basados en grupos de usuarios. Tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
usrgrpid	ID	ID del grupo de usuarios. Comportamiento de la propiedad: - *requerido*

Propiedad	Tipo	Descripción
permission	integer	Tipo de nivel de permiso. Valores posibles: 2 - solo lectura; 3 - lectura-escritura. Comportamiento de la propiedad: - *requerido*

Usuario del dashboard

Lista de permisos del dashboard basados en usuarios. Tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
userid	ID	ID del usuario. Comportamiento de la propiedad: - *requerido*
permission	integer	Tipo de nivel de permiso. Valores posibles: 2 - solo lectura; 3 - lectura-escritura. Comportamiento de la propiedad: - *requerido*

dashboard.create

Descripción

`object dashboard.create(object/array dashboards)`

Este método le permite crear nuevos dashboards.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object/array)` Dashboards a crear.

Además de las propiedades estándar del dashboard, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
pages	array	Páginas del dashboard que se crearán para el dashboard. Las páginas del dashboard se ordenarán en el mismo orden que se especifique. Comportamiento del parámetro: - *obligatorio*
users	array	Usuarios del dashboard que se compartirán en el dashboard.
userGroups	array	Grupos de usuarios del dashboard que se compartirán en el dashboard.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los paneles creados bajo la propiedad `dashboardids`. El orden de los IDs devueltos coincide con el orden de los paneles enviados.

Ejemplos

Creando un panel

Cree un panel llamado "Mi panel" con un widget de Problemas con etiquetas y utilizando dos tipos de compartición (grupo de usuarios y usuario) en una sola página de panel.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "problems",
                        "x": 0,
                        "y": 0,
                        "width": 36,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 1,
                                "name": "tags.0.tag",
                                "value": "service"
                            },
                            {
                                "type": 0,
                                "name": "tags.0.operator",
                                "value": 1
                            },
                            {
                                "type": 1,
                                "name": "tags.0.value",
                                "value": "zabbix_server"
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": "7",
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": "4",
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
```

```
    "result": {
        "dashboardids": [
            "2"
        ]
    },
    "id": 1
}
```

Ver también

- Página de dashboard
- Widget de dashboard
- Campo de widget de dashboard
- Usuario de dashboard
- Grupo de usuarios de dashboard

Fuente

CDashboard::create() en *ui/include/classes/api/services/CDashboard.php*.

dashboard.delete

Descripción

`object dashboard.delete(array dashboardids)`

Este método le permite eliminar dashboards.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(array)` IDs de los paneles a eliminar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los paneles eliminados bajo la propiedad `dashboardids`.

Ejemplos

Eliminando múltiples paneles

Eliminar dos paneles.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.delete",
    "params": [
        "2",
        "3"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "2",
            "3"
        ]
    },
```

```
    "id": 1
}
```

Fuente

CDashboard::delete() en *ui/include/classes/api/services/CDashboard.php*.

dashboard.get

Descripción

`integer/array dashboard.get(object parameters)`

El método permite recuperar dashboards según los parámetros proporcionados.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
dashboardids	ID/array	Devolver solo los paneles con los IDs dados.
selectPages	query	Devolver una propiedad `pages` con las páginas del panel, correctamente ordenadas.
selectUsers	query	Devolver una propiedad `users` con los usuarios con los que se comparte el panel.
selectUserGroups	query	Devolver una propiedad `userGroups` con los grupos de usuarios con los que se comparte el panel.
sortfield	string/array	Ordenar el resultado por las propiedades dadas. Valores posibles: `dashboardid`.
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperar un panel por ID

Recuperar todos los datos sobre los paneles "1" y "2".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.get",
    "params": {
        "output": "extend",
        "selectPages": "extend",
        "selectUsers": "extend",
        "selectUserGroups": "extend",
        "dashboardids": [
            "1",
            "2"
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "dashboardid": "1",
            "name": "Dashboard",
            "userid": "1",
            "private": "0",
            "display_period": "30",
            "auto_start": "1",
            "users": [],
            "userGroups": [],
            "pages": [
                {
                    "dashboard_pageid": "1",
                    "name": "",
                    "display_period": "0",
                    "widgets": [
                        {
                            "widgetid": "9",
                            "type": "systeminfo",
                            "name": "",
                            "x": "12",
                            "y": "8",
                            "width": "12",
                            "height": "5",
                            "view_mode": "0",
                            "fields": []
                        },
                        {
                            "widgetid": "8",
                            "type": "problemsbysv",
                            "name": "",
                            "x": "12",
                            "y": "4",
                            "width": "12",
                            "height": "4",
                            "view_mode": "0",
                            "fields": []
                        },
                        {
                            "widgetid": "7",
                            "type": "problemhosts",
                            "name": "",
                            "x": "12",
```

```
                "y": "0",
                "width": "12",
                "height": "4",
                "view_mode": "0",
                "fields": []
            },
            {
                "widgetid": "6",
                "type": "discovery",
                "name": "",
                "x": "6",
                "y": "9",
                "width": "18",
                "height": "4",
                "view_mode": "0",
                "fields": []
            },
            {
                "widgetid": "5",
                "type": "web",
                "name": "",
                "x": "0",
                "y": "9",
                "width": "18",
                "height": "4",
                "view_mode": "0",
                "fields": []
            },
            {
                "widgetid": "4",
                "type": "problems",
                "name": "",
                "x": "0",
                "y": "3",
                "width": "12",
                "height": "6",
                "view_mode": "0",
                "fields": []
            },
            {
                "widgetid": "3",
                "type": "favmaps",
                "name": "",
                "x": "8",
                "y": "0",
                "width": "12",
                "height": "3",
                "view_mode": "0",
                "fields": []
            },
            {
                "widgetid": "1",
                "type": "favgraphs",
                "name": "",
                "x": "0",
                "y": "0",
                "width": "12",
                "height": "3",
                "view_mode": "0",
                "fields": []
            }
        ]
```

```
            },
            {
                "dashboard_pageid": "2",
                "name": "",
                "display_period": "0",
                "widgets": []
            },
            {
                "dashboard_pageid": "3",
                "name": "Custom page name",
                "display_period": "60",
                "widgets": []
            }
        ]
    },
    {
        "dashboardid": "2",
        "name": "My dashboard",
        "userid": "1",
        "private": "1",
        "display_period": "60",
        "auto_start": "1",
        "users": [
            {
                "userid": "4",
                "permission": "3"
            }
        ],
        "userGroups": [
            {
                "usrgrpid": "7",
                "permission": "2"
            }
        ],
        "pages": [
            {
                "dashboard_pageid": "4",
                "name": "",
                "display_period": "0",
                "widgets": [
                    {
                        "widgetid": "10",
                        "type": "problems",
                        "name": "",
                        "x": "0",
                        "y": "0",
                        "width": "12",
                        "height": "5",
                        "view_mode": "0",
                        "fields": [
                            {
                                "type": "2",
                                "name": "groupids",
                                "value": "4"
                            }
                        ]
                    }
                ]
            }
        ]
    }
],
```

```
    "id": 1
}
```

Ver también

- Página de dashboard
- Widget de dashboard
- Campo de widget de dashboard
- Usuario de dashboard
- Grupo de usuarios de dashboard

Fuente

CDashboard::get() en *ui/include/classes/api/services/CDashboard.php*.

dashboard.update

Descripción

`object dashboard.update(object/array dashboards)`

Este método permite actualizar dashboards existentes.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(objeto/array)` Propiedades del dashboard que se actualizarán.

La propiedad `dashboardid` debe definirse para cada dashboard, todas las demás propiedades son opcionales. Sólo se actualizarán las propiedades pasadas, todas las demás permanecerán sin cambios.

Además de las propiedades estándar del dashboard, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
pages	array	Páginas del dashboard para reemplazar las páginas existentes. Las páginas del dashboard se actualizan por la propiedad `dashboard_pageid`. Se crearán nuevas páginas para los objetos sin la propiedad `dashboard_pageid` y las páginas existentes se eliminarán si no se reutilizan. Las páginas del dashboard se ordenarán en el mismo orden que se especifique. Sólo se actualizarán las propiedades especificadas de las páginas del dashboard.
users	array	Usuarios del dashboard para reemplazar los elementos existentes.
userGroups	array	Grupos de usuarios del dashboard para reemplazar los elementos existentes.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los tableros actualizados bajo la propiedad `dashboardids`.

Ejemplos

Renombrar un panel

Renombrar un panel a "Estado del servidor SQL".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.update",
    "params": {
        "dashboardid": "2",
        "name": "Estado del servidor SQL"
```

```
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "2"
        ]
    },
    "id": 1
}
```

Actualización de páginas del panel

Cambie el nombre de la primera página del panel, reemplace los widgets en la segunda página del panel y agregue una nueva página como la tercera. Elimine todas las demás páginas del panel.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.update",
    "params": {
        "dashboardid": "2",
        "pages": [
            {
                "dashboard_pageid": 1,
                "name": "Página renombrada"
            },
            {
                "dashboard_pageid": 2,
                "widgets": [
                    {
                        "type": "clock",
                        "x": 0,
                        "y": 0,
                        "width": 12,
                        "height": 3
                    }
                ]
            },
            {
                "display_period": 60
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "2"
        ]
    },
    "id": 1
}
```

Cambiar propietario del panel

Disponible solo para administradores y superadministradores.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.update",
    "params": {
        "dashboardid": "2",
        "userid": "1"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "2"
        ]
    },
    "id": 1
}
```

Ver también

- Página de dashboard
- Widget de dashboard
- Campo de widget de dashboard
- Usuario de dashboard
- Grupo de usuarios de dashboard

Fuente

CDashboard::update() en *ui/include/classes/api/services/CDashboard.php*.

Campos de los widgets del panel

Esta página contiene enlaces de navegación para los parámetros de los widgets del dashboard y los posibles valores de propiedad para los respectivos objetos dashboard widget field.

Para ver los parámetros y valores de propiedad para cada widget, utilice la barra lateral para navegar a la página correspondiente del widget.

1 Registro de acciones

Descripción

Estos parámetros y los posibles valores de propiedad para los objetos de campo del widget del dashboard correspondiente permiten configurar el widget *Registro de acciones* en los métodos `dashboard.create` y `dashboard.update`.

> **Attention:**
> Las propiedades `fields` del widget no se validan durante la creación o actualización de un dashboard. Esto permite a los usuarios modificar widgets integrados y crear widgets personalizados, pero también introduce el riesgo de crear o actualizar widgets de forma incorrecta. Para garantizar la creación o actualización correctas del widget *Registro de acciones*, consulte el comportamiento de los parámetros descrito en las tablas siguientes.

Parámetros

Los siguientes parámetros son compatibles con el widget *Registro de acciones*.

Parámetro	tipo	nombre	valor
Intervalo de actualización	0	rf_rate	0 - Sin actualización; 10 - 10 segundos; 30 - 30 segundos; 60 - *(predeterminado)* 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - 15 minutos.
Destinatarios	11	userids.0	ID del usuario. Nota: Para configurar varios usuarios, cree un objeto de campo del widget del dashboard para cada usuario, con un número incrementado en el nombre de la propiedad.
Acciones	12	actionids.0	ID de la acción. Nota: Para configurar varias acciones, cree un objeto de campo del widget del dashboard para cada acción, con un número incrementado en el nombre de la propiedad.
Tipos de medios	13	mediatypeids.0	ID del tipo de medio. Nota: Para configurar varios tipos de medios, cree un objeto de campo del widget del dashboard para cada tipo de medio, con un número incrementado en el nombre de la propiedad.
Estado	0	statuses.0	0 - En progreso; 1 - Enviado/Ejecutado; 2 - Error. Nota: Para configurar varios valores, cree un objeto de campo del widget del dashboard para cada valor, con un número incrementado en el nombre de la propiedad.
Cadena de búsqueda	1	message	Cualquier valor de cadena.
Período de tiempo	1	time_period._referenc	`DASHBOARD._timeperiod` - establece el selector de período de tiempo del dashboard como fuente de datos; `ABCDE._timeperiod` - establece un widget compatible (con su parámetro *Referencia* establecido en "`ABCDE`") como fuente de datos. Predeterminado: `DASHBOARD._timeperiod` Como alternativa, puede establecer el período de tiempo únicamente en los parámetros *Desde* y *Hasta*.
Desde	1	time_period.from	Cadena de tiempo válida en sintaxis absoluta (`YYYY-MM-DD hh:mm:ss`) o relativa (`now`, `now/d`, `now/w-1w`, etc.). Comportamiento del parámetro: - *compatible* si no se establece *Período de tiempo* - *obligatorio* si se establece `time_period.to`
Hasta	1	time_period.to	Cadena de tiempo válida en sintaxis absoluta (`YYYY-MM-DD hh:mm:ss`) o relativa (`now`, `now/d`, `now/w-1w`, etc.). Comportamiento del parámetro: - *compatible* si no se establece *Período de tiempo* - *obligatorio* si se establece `time_period.from`
Ordenar entradas por	0	sort_triggers	3 - Hora (ascendente); 4 - *(predeterminado)* Hora (descendente); 5 - Tipo (ascendente); 6 - Tipo (descendente); 7 - Estado (ascendente); 8 - Estado (descendente); 11 - Destinatario (ascendente); 12 - Destinatario (descendente).

Parámetro	tipo	nombre	valor
Mostrar líneas	0	show_lines	Los valores posibles van de 1 a 100. Predeterminado: 25.

Ejemplos

Los siguientes ejemplos tienen como objetivo describir únicamente la configuración de los objetos de campo del widget de panel para el widget *Registro de acciones*. Para obtener más información sobre la configuración de un panel, consulte dashboard.create.

Configuración de un widget *Registro de acciones*

Configure un widget *Registro de acciones* que muestre 10 entradas de detalles de operaciones de acciones, ordenadas por hora (en orden ascendente). Además, muestre detalles solo de aquellas operaciones de acciones que intentaron enviar un correo electrónico al usuario "1", pero no tuvieron éxito.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "Mi panel",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "actionlog",
                        "name": "Registro de acciones",
                        "x": 0,
                        "y": 0,
                        "width": 36,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 0,
                                "name": "show_lines",
                                "value": 10
                            },
                            {
                                "type": 0,
                                "name": "sort_triggers",
                                "value": 3
                            },
                            {
                                "type": 11,
                                "name": "userids.0",
                                "value": 1
                            },
                            {
                                "type": 13,
                                "name": "mediatypeids.0",
                                "value": 1
                            },
                            {
                                "type": 0,
                                "name": "statuses.0",
                                "value": 2
                            }
                        ]
```

```
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Véase también

- Campo de widget de panel
- `dashboard.create`
- `dashboard.update`

2 Reloj

Descripción

Estos parámetros y los posibles valores de propiedad para los objetos de campo correspondientes del widget del dashboard permiten configurar el widget *Reloj* en los métodos `dashboard.create` y `dashboard.update`.

Attention:
Las propiedades `fields` del widget no se validan durante la creación o actualización de un dashboard. Esto permite a los usuarios modificar widgets integrados y crear widgets personalizados, pero también introduce el riesgo de crear o actualizar widgets de forma incorrecta. Para garantizar la creación o actualización correcta del widget *Reloj*, consulte el comportamiento de los parámetros descrito en las tablas siguientes.

Parámetros

Los siguientes parámetros son compatibles para el widget *Reloj*.

Parámetro	tipo	nombre	valor
Intervalo de actualización	0	rf_rate	0 - Sin actualización; 10 - 10 segundos; 30 - 30 segundos; 60 - 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - *(por defecto)* 15 minutos.
Tipo de hora	0	time_type	0 - *(por defecto)* Hora local; 1 - Hora del servidor; 2 - Hora del equipo.
Tipo de reloj	0	clock_type	0 - *(por defecto)* Analógico; 1 - Digital.

Los siguientes parámetros son compatibles si *Tipo de hora* está configurado como ”Hora del equipo”.

Parámetro	tipo	nombre	valor
Métrica	4	itemid.0	ID de la métrica. Comportamiento del parámetro: - *obligatorio*

Los siguientes parámetros son compatibles si *Tipo de reloj* está configurado como ”Digital”.

Parámetro	tipo	nombre	valor
Mostrar	0	show.0	1 - Fecha; 2 - *(por defecto)* Hora; 3 - Zona horaria. Nota: Para configurar múltiples valores, cree un objeto de campo de widget de panel para cada valor con un número incrementado en el nombre de la propiedad.

Configuración avanzada

Se admiten los siguientes parámetros de configuración avanzada si el *Tipo de reloj* está establecido en ”Digital”.

Parámetro	tipo	nombre	valor
Color de fondo	1	bg_color	Código de color hexadecimal (por ejemplo, FF0000). Por defecto: "" (vacío).

Fecha

Se admiten los siguientes parámetros de configuración avanzada si el *Tipo de reloj* está configurado en ”Digital” y *Mostrar* está configurado en ”Fecha”.

Parámetro	tipo	nombre	valor
Negrita	0	date_bold	0 - *(por defecto)* Deshabilitado; 1 - Habilitado.
Color	1	date_color	Código de color hexadecimal (por ejemplo, FF0000). Por defecto: "" (vacío).

Hora

Se admiten los siguientes parámetros de configuración avanzada si *Tipo de reloj* está configurado como ”Digital” y *Mostrar* está configurado como ”Hora”.

Parámetro	tipo	nombre	valor
Negrita	0	time_bold	0 - *(por defecto)* Deshabilitado; 1 - Habilitado.
Color	1	time_color	Código de color hexadecimal (por ejemplo, `FF0000`). Por defecto: `""` (vacío).
Segundos	0	time_sec	0 - Deshabilitado; 1 - *(por defecto)* Habilitado.
Formato	0	time_format	0 - *(por defecto)* 24 horas; 1 - 12 horas.

Zona horaria

Se admiten los siguientes parámetros de configuración avanzada si *Tipo de reloj* está establecido en "Digital" y *Mostrar* está establecido en "Zona horaria".

Parámetro	tipo	nombre	valor
Negrita	0	tzone_bold	0 - *(por defecto)* Deshabilitado; 1 - Habilitado.
Color	1	tzone_color	Código de color hexadecimal (por ejemplo, `FF0000`). Por defecto: `""` (vacío).
Zona horaria	1	tzone_timezone	Cadena de zona horaria válida (por ejemplo, `Europe/Riga`, `system`, `UTC`, etc.). Para la lista completa de zonas horarias admitidas, consulte la documentación de PHP. Por defecto: `local`. Comportamiento del parámetro: - *admitido* si *Tipo de hora* está establecido en "Hora local" o "Hora del servidor"
Formato	0	tzone_format	0 - *(por defecto)* Corto; 1 - Completo. Comportamiento del parámetro: - *admitido* si *Tipo de hora* está establecido en "Hora local" o "Hora del servidor"

Ejemplos

Los siguientes ejemplos tienen como objetivo describir únicamente la configuración de los objetos de campo del widget de panel para el widget *Reloj*. Para obtener más información sobre la configuración de un panel, consulte `dashboard.create`.

Configuración de un widget *Reloj*

Configure un widget *Reloj* que muestre la fecha, hora y zona horaria local en un reloj digital personalizado.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "Mi panel",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "clock",
                        "name": "Reloj",
                        "x": 0,
```

```
                    "y": 0,
                    "width": 12,
                    "height": 3,
                    "view_mode": 0,
                    "fields": [
                        {
                            "type": 0,
                            "name": "clock_type",
                            "value": 1
                        },
                        {
                            "type": 0,
                            "name": "show.0",
                            "value": 1
                        },
                        {
                            "type": 0,
                            "name": "show.1",
                            "value": 2
                        },
                        {
                            "type": 0,
                            "name": "show.2",
                            "value": 3
                        },
                        {
                            "type": 1,
                            "name": "date_color",
                            "value": "E1E1E1"
                        },
                        {
                            "type": 0,
                            "name": "time_bold",
                            "value": 1
                        },
                        {
                            "type": 1,
                            "name": "tzone_color",
                            "value": "E1E1E1"
                        },
                        {
                            "type": 1,
                            "name": "tzone_timezone",
                            "value": "Europe/Riga"
                        },
                        {
                            "type": 0,
                            "name": "tzone_format",
                            "value": 1
                        }
                    ]
                }
            ]
        }
    ],
    "userGroups": [
        {
            "usrgrpid": 7,
            "permission": 2
        }
    ],
    "users": [
```

```
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Véase también

- Campo de widget de panel
- `dashboard.create`
- `dashboard.update`

3 Estado de descubrimiento

Descripción

Estos parámetros y los posibles valores de propiedad para los objetos de campo del widget del dashboard correspondiente permiten configurar el widget *Estado del descubrimiento* en los métodos `dashboard.create` y `dashboard.update`.

Parámetros

Los siguientes parámetros son compatibles con el widget *Estado de descubrimiento*.

Parámetro	tipo	nombre	valor
Intervalo de actualización	0	rf_rate	0 - Sin actualización; 10 - 10 segundos; 30 - 30 segundos; 60 - *(por defecto)* 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - 15 minutos.

Ejemplos

Los siguientes ejemplos tienen como objetivo describir únicamente la configuración de los objetos de campo del widget del panel para el widget *Estado de descubrimiento*. Para obtener más información sobre la configuración de un panel, consulte `dashboard.create`.

Configuración del widget *Estado de descubrimiento*

Configure un widget *Estado de descubrimiento* con el intervalo de actualización establecido en 15 minutos.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "Mi panel",
        "display_period": 30,
```

```
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "discovery",
                        "name": "Estado de descubrimiento",
                        "x": 0,
                        "y": 0,
                        "width": 18,
                        "height": 3,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 0,
                                "name": "rf_rate",
                                "value": 900
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Véase también

- Campo de widget de panel
- `dashboard.create`
- `dashboard.update`

4 Gráficas favoritas

Descripción

Estos parámetros y los posibles valores de propiedad de los objetos de campo del widget del dashboard correspondiente permiten configurar el widget *Gráficos favoritos* en los métodos `dashboard.create` y `dashboard.update`.

Parámetros

Los siguientes parámetros son compatibles con el widget *Gráficas favoritas*.

Parámetro	tipo	nombre	valor
Intervalo de actualización	0	rf_rate	0 - Sin actualización; 10 - 10 segundos; 30 - 30 segundos; 60 - 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - *(por defecto)* 15 minutos.

Ejemplos

Los siguientes ejemplos tienen como objetivo describir únicamente la configuración de los objetos de campo del widget de panel para el widget *Gráficas favoritas*. Para obtener más información sobre la configuración de un panel, consulte `dashboard.create`.

Configuración de un widget *Gráficas favoritas*

Configure un widget *Gráficas favoritas* con el intervalo de actualización establecido en 10 minutos.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "Mi panel",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "favgraphs",
                        "name": "Gráficas favoritas",
                        "x": 0,
                        "y": 0,
                        "width": 12,
                        "height": 3,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 0,
                                "name": "rf_rate",
                                "value": 600
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
```

```
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Véase también

- Campo de widget de panel
- `dashboard.create`
- `dashboard.update`

5 Mapas favoritos

Descripción

Estos parámetros y los posibles valores de propiedad para los objetos de campo correspondientes del widget del dashboard permiten configurar el widget *Mapas favoritos* en los métodos `dashboard.create` y `dashboard.update`.

Parámetros

Los siguientes parámetros son compatibles con el widget *Mapas favoritos*.

Parámetro	tipo	nombre	valor
Intervalo de actualización	0	rf_rate	0 - Sin actualización; 10 - 10 segundos; 30 - 30 segundos; 60 - 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - *(por defecto)* 15 minutos.

Ejemplos

Los siguientes ejemplos tienen como objetivo describir únicamente la configuración de los objetos de campo del widget de panel para el widget *Mapas favoritos*. Para obtener más información sobre la configuración de un panel, consulte `dashboard.create`.

Configuración de un widget *Mapas favoritos*

Configure un widget *Mapas favoritos* con el intervalo de actualización establecido en 10 minutos.

Petición:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "favmaps",
```

```
                        "name": "Favorite maps",
                        "x": 0,
                        "y": 0,
                        "width": 12,
                        "height": 3,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 0,
                                "name": "rf_rate",
                                "value": 600
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Véase también

- Campo de widget de panel
- `dashboard.create`
- `dashboard.update`

6 Medidor

Descripción

Estos parámetros y los posibles valores de propiedad para los objetos de campo correspondientes del widget del dashboard permiten configurar el widget *Gauge* en los métodos `dashboard.create` y `dashboard.update`.

Attention:
Las propiedades `fields` del widget no se validan durante la creación o actualización de un dashboard. Esto permite a los usuarios modificar widgets integrados y crear widgets personalizados, pero también introduce el riesgo de crear o actualizar widgets de forma incorrecta. Para garantizar la creación o actualización correctas del widget *Gauge*, consulte el comportamiento de los parámetros descrito en las tablas siguientes.

Parámetros

Se admiten los siguientes parámetros para el widget *Gauge*.

Parámetro	tipo	nombre	valor
Intervalo de actualización	0	rf_rate	0 - Sin actualización; 10 - 10 segundos; 30 - 30 segundos; 60 - *(por defecto)* 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - 15 minutos.
Elemento	4	itemid.0	ID del Elemento. Comportamiento del parámetro: - *obligatorio* si *Elemento (Widget)* no está establecido
Elemento (Widget)	1	itemid._reference	En lugar del ID del Elemento: `ABCDE._itemid` - establezca un widget compatible (con su parámetro *Referencia* configurado en "ABCDE") como fuente de datos para los elementos. Comportamiento del parámetro: - *obligatorio* si *Elemento* no está establecido
Mín	1	min	Cualquier valor numérico. Se admiten Sufijos (por ejemplo, "1d", "2w", "4K", "8G"). Por defecto: "0".
Máx	1	max	Cualquier valor numérico. Se admiten Sufijos (por ejemplo, "1d", "2w", "4K", "8G"). Por defecto: "100".
Arco de valor	1	value_arc_color	Código de color hexadecimal (por ejemplo, `FF0000`). Por defecto: `""` (vacío).
Fondo del arco	1	empty_color	Código de color hexadecimal (por ejemplo, `FF0000`). Por defecto: `""` (vacío).
Fondo	1	bg_color	Código de color hexadecimal (por ejemplo, `FF0000`). Por defecto: `""` (vacío).

Parámetro	tipo	nombre	valor
Mostrar	0	show.0	1 - Descripción; 2 - Valor; 3 - Aguja; 4 - Escala; 5 - Arco de valor. Nota: Para configurar varios valores, cree un objeto de campo de widget de panel para cada valor con un número incrementado en el nombre de la propiedad. Por defecto: 1, 2, 4, 5. Los valores ”Aguja” y ”Escala” no son compatibles si ambos: - no se establece ningún objeto de campo de widget de panel para *Mostrar* con el valor ”Arco de valor”; - el parámetro de configuración avanzada *Mostrar arco* está configurado en ”Deshabilitado”. Los parámetros de configuración avanzada para las opciones de *Mostrar* no son compatibles si no se establecen objetos de campo de widget de panel con los valores respectivos.
Anular equipo	1	override_hostid._referencia	`ABCDE._hostid` - establezca un widget compatible (con su parámetro *Referencia* configurado en ”ABCDE”) como fuente de datos para los equipos; `DASHBOARD._hostid` - establezca el selector de equipos del panel como fuente de datos para los equipos. Este parámetro no es compatible si se configura el widget en un panel de plantilla.

Configuración avanzada

Los siguientes parámetros de configuración avanzada son compatibles con el widget *Gauge*.

Note:
El número en el nombre de la propiedad *Thresholds* (por ejemplo, thresholds.0.color) hace referencia a la posición del umbral en una lista ordenada de forma ascendente. Sin embargo, si los umbrales se configuran en un orden diferente, los valores se ordenarán de forma ascendente después de actualizar la configuración del widget en el frontend de Zabbix (por ejemplo, `"thresholds.0.threshold":"5"` → `"thresholds.0.threshold":"1"`; `"thresholds.1.threshold":"1"` → `"thresholds.1.threshold": "5"`).

Parámetro	type	name	value
Ángulo	0	angle	Valores posibles: 180 *(predeterminado)* o 270.
Descripción			
Descripción	1	description	Cualquier valor de cadena, incluidas las macros. Macros compatibles: `{HOST.*}`, `{ITEM.*}`, `{INVENTORY.*}`, macros de usuario. Predeterminado: `{ITEM.NAME}`.
Tamaño	0	desc_size	Los valores posibles van de 1 a 100. Predeterminado: 15.
Posición vertical	0	desc_v_pos	0 - Arriba; 1 - *(predeterminado)* Abajo.
Negrita	0	desc_bold	0 - *(predeterminado)* Deshabilitado; 1 - Habilitado.
Color	1	desc_color	Código de color hexadecimal (por ejemplo, `FF0000`). Predeterminado: `""` (vacío).
Valor			

Parámetro	type	name	value
Posiciones decimales	0	decimal_places	Los valores posibles van de 1 a 10. Predeterminado: 2.
Tamaño	0	value_size	Los valores posibles van de 1 a 100. Predeterminado: 25.
Negrita	0	value_bold	0 - *(predeterminado)* Deshabilitado; 1 - Habilitado.
Color	1	value_color	Código de color hexadecimal (por ejemplo, FF0000). Predeterminado: "" (vacío).
Unidades			
Unidades (casilla de verificación)	0	units_show	0 - Deshabilitado; 1 - *(predeterminado)* Habilitado.
Unidades (valor)	1	units	Cualquier valor de cadena. Comportamiento del parámetro: - *compatible* si *Unidades* (casilla de verificación) está configurado como ”Habilitado”
Tamaño	0	units_size	Los valores posibles van de 1 a 100. Predeterminado: 25. Comportamiento del parámetro: - *compatible* si *Unidades* (casilla de verificación) está configurado como ”Habilitado”
Negrita	0	units_bold	0 - *(predeterminado)* Deshabilitado; 1 - Habilitado. Comportamiento del parámetro: - *compatible* si *Unidades* (casilla de verificación) está configurado como ”Habilitado”
Posición	0	units_pos	0 - Antes del valor; 1 - Encima del valor; 2 - *(predeterminado)* Después del valor; 3 - Debajo del valor. Comportamiento del parámetro: - *compatible* si *Unidades* (casilla de verificación) está configurado como ”Habilitado” Este parámetro se ignora si se configura con una de las siguientes unidades relacionadas con el tiempo: unixtime, uptime, s.
Color	1	units_color	Código de color hexadecimal (por ejemplo, FF0000). Predeterminado: "" (vacío).
Arco de valor			
Tamaño del arco	0	value_arc_size	Los valores posibles van de 1 a 100. Predeterminado: 20.
Aguja			
Color	1	needle_color	Código de color hexadecimal (por ejemplo, FF0000). Predeterminado: "" (vacío). Comportamiento del parámetro: - *compatible* si se configura un objeto de campo del widget del dashboard para *Mostrar* con el valor ”Arco de valor”, o si *Mostrar arco* está configurado como ”Habilitado”
Escala			

Parámetro	type	name	value
Mostrar unidades	0	scale_show_units	0 - Deshabilitado; 1 - *(predeterminado)* Habilitado. Comportamiento del parámetro: - *compatible* si *Unidades* (casilla de verificación) está configurado como ”Habilitado” y se configura un objeto de campo del widget del dashboard para *Mostrar* con el valor ”Arco de valor”, o si *Mostrar arco* está configurado como ”Habilitado”
Tamaño	0	scale_size	Los valores posibles van de 1 a 100. Predeterminado: 15. Comportamiento del parámetro: - *compatible* si se configura un objeto de campo del widget del dashboard para *Mostrar* con el valor ”Arco de valor”, o si *Mostrar arco* está configurado como ”Habilitado”
Posiciones decimales	0	scale_decimal_places	Los valores posibles van de 1 a 10. Predeterminado: 0. Comportamiento del parámetro: - *compatible* si se configura un objeto de campo del widget del dashboard para *Mostrar* con el valor ”Arco de valor”, o si *Mostrar arco* está configurado como ”Habilitado”
Umbrales			
Color	1	thresholds.0.color	Código de color hexadecimal (por ejemplo, FF0000).
Umbral	1	thresholds.0.threshold	Cualquier valor numérico. Se admiten sufijos (por ejemplo, ”1d”, ”2w”, ”4K”, ”8G”).
Mostrar etiquetas	0	th_show_labels	0 - *(predeterminado)* Deshabilitado; 1 - Habilitado. Comportamiento del parámetro: - *compatible* si se han configurado *Umbrales* y se configura un objeto de campo del widget del dashboard para *Mostrar* con el valor ”Arco de valor”, o si *Mostrar arco* está configurado como ”Habilitado”
Mostrar arco	0	th_show_arc	0 - *(predeterminado)* Deshabilitado; 1 - Habilitado. Comportamiento del parámetro: - *compatible* si se han configurado *Umbrales*
Tamaño del arco	0	th_arc_size	Los valores posibles van de 1 a 100. Predeterminado: 5. Comportamiento del parámetro: - *compatible* si *Mostrar arco* está configurado como ”Habilitado”

Ejemplos

Los siguientes ejemplos tienen como objetivo describir únicamente la configuración de los objetos de campo del widget de panel para el widget *Indicador*. Para obtener más información sobre la configuración de un panel, consulte `dashboard.create`.

Configuración de un widget *Gauge*

Configure un widget *Gauge* que muestre el valor de la métrica ”44474” (Interfaz enp0s3: Bits enviados). Además, ajuste visualmente el widget con múltiples opciones avanzadas, incluidos los umbrales.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
```

```
        "name": "Mi panel",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "gauge",
                        "name": "Gauge",
                        "x": 0,
                        "y": 0,
                        "width": 18,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 4,
                                "name": "itemid.0",
                                "value": 44474
                            },
                            {
                                "type": 1,
                                "name": "min",
                                "value": "100000"
                            },
                            {
                                "type": 1,
                                "name": "max",
                                "value": "1000000"
                            },
                            {
                                "type": 0,
                                "name": "show.0",
                                "value": 1
                            },
                            {
                                "type": 0,
                                "name": "show.1",
                                "value": 2
                            },
                            {
                                "type": 0,
                                "name": "show.2",
                                "value": 3
                            },
                            {
                                "type": 0,
                                "name": "show.4",
                                "value": 4
                            },
                            {
                                "type": 0,
                                "name": "show.5",
                                "value": 5
                            },
                            {
                                "type": 0,
                                "name": "angle",
                                "value": 270
                            },
                            {
                                "type": 0,
```

```
            "name": "desc_size",
            "value": 10
        },
        {
            "type": 0,
            "name": "desc_bold",
            "value": 1
        },
        {
            "type": 0,
            "name": "decimal_places",
            "value": 0
        },
        {
            "type": 0,
            "name": "value_bold",
            "value": 1
        },
        {
            "type": 0,
            "name": "units_size",
            "value": 15
        },
        {
            "type": 0,
            "name": "units_pos",
            "value": 3
        },
        {
            "type": 1,
            "name": "needle_color",
            "value": "3C3C3C"
        },
        {
            "type": 1,
            "name": "thresholds.0.color",
            "value": "FF465C"
        },
        {
            "type": 1,
            "name": "thresholds.0.threshold",
            "value": "700000"
        },
        {
            "type": 1,
            "name": "thresholds.1.color",
            "value": "FFD54F"
        },
        {
            "type": 1,
            "name": "thresholds.1.threshold",
            "value": "500000"
        },
        {
            "type": 1,
            "name": "thresholds.2.color",
            "value": "0EC9AC"
        },
        {
            "type": 1,
            "name": "thresholds.2.threshold",
            "value": "100000"
```

```
                            },
                            {
                                "type": 0,
                                "name": "th_show_labels",
                                "value": 1
                            },
                            {
                                "type": 0,
                                "name": "th_show_arc",
                                "value": 1
                            },
                            {
                                "type": 0,
                                "name": "th_arc_size",
                                "value": 15
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Véase también

- Campo de widget de panel
- `dashboard.create`
- `dashboard.update`

7 Geomapa

Descripción

Estos parámetros y los posibles valores de las propiedades de los objetos de campo del widget del dashboard correspondiente permiten configurar el widget *Geomap* en los métodos `dashboard.create` y `dashboard.update`.

Attention:
Las propiedades `fields` del widget no se validan durante la creación o actualización de un dashboard. Esto permite a los usuarios modificar widgets integrados y crear widgets personalizados, pero también introduce el riesgo de crear o actualizar widgets de forma incorrecta. Para garantizar la creación o actualización correcta del widget *Geomap*, consulte el comportamiento de los parámetros descrito en las tablas siguientes.

Parámetros

Los siguientes parámetros son compatibles con el widget *Geomap*.

Parámetro	tipo	nombre	valor
Intervalo de actualización	0	rf_rate	0 - Sin actualización; 10 - 10 segundos; 30 - 30 segundos; 60 - *(predeterminado)* 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - 15 minutos.
Grupos de hosts	2	groupids.0	ID del grupo de hosts. Nota: Para configurar varios grupos de hosts, cree un objeto de campo del widget del dashboard para cada grupo de hosts, con un número incrementado en el nombre de la propiedad. Este parámetro no es compatible si se configura el widget en un dashboard de template.
Grupos de hosts (widget)	1	groupids._reference	En lugar del ID del grupo de hosts: `ABCDE._hostgroupids` - establezca un widget compatible (con su parámetro *Reference* establecido en "`ABCDE`") como fuente de datos para los grupos de hosts. Este parámetro no es compatible si se configura el widget en un dashboard de template.
Hosts	3	hostids.0	ID del host. Nota: Para configurar varios hosts, cree un objeto de campo del widget del dashboard para cada host, con un número incrementado en el nombre de la propiedad. Para varios hosts, el parámetro *Grupos de hosts* debe no estar configurado o estar configurado con al menos un grupo de hosts al que pertenezcan los hosts configurados. Este parámetro no es compatible si se configura el widget en un dashboard de template.
Hosts (widget/dashboard)	1	hostids._reference	En lugar del ID del host: `DASHBOARD.hostids` - establezca el selector de hosts del dashboard como fuente de datos para los hosts; `ABCDE._hostids` - establezca un widget compatible (con su parámetro *Reference* establecido en "`ABCDE`") como fuente de datos para los hosts. Este parámetro no es compatible si se configura el widget en un dashboard de template.
Etiquetas			
Tipo de evaluación	0	evaltype	0 - *(predeterminado)* Y/O; 2 - O. Este parámetro no es compatible si se configura el widget en un dashboard de template.

Parámetro	tipo	nombre	valor
Nombre de etiqueta	1	tags.0.tag	Cualquier valor de cadena. Nota: El número del nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* al configurar *Etiquetas* Este parámetro no es compatible si se configura el widget en un dashboard de template.
Operador	0	tags.0.operator	0 - Contiene; 1 - Es igual a; 2 - No contiene; 3 - No es igual a; 4 - Existe; 5 - No existe. Nota: El número del nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* al configurar *Etiquetas* Este parámetro no es compatible si se configura el widget en un dashboard de template.
Valor de etiqueta	1	tags.0.value	Cualquier valor de cadena. Nota: El número del nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* al configurar *Etiquetas* Este parámetro no es compatible si se configura el widget en un dashboard de template.
Vista inicial	1	default_view	*latitud*, *longitud*, *nivel de zoom* separados por comas (*opcional*, los valores posibles van de 0 a 30). Ejemplo: `40.6892494,-74.0466891,10`.
Agrupación			
Modo de agrupación	0	clustering_mode	0 - *(predeterminado)* Automático; 1 - Nivel de zoom.
Nivel de zoom de agrupación	0	clustering_zoom_level	Los valores posibles van de 0 a 30. Predeterminado: 0. Comportamiento del parámetro: - *compatible* si *Modo de agrupación* está establecido en "Nivel de zoom"
Referencia	1	reference	Cualquier valor de cadena compuesto por 5 caracteres (por ejemplo, `ABCDE` o `JBPNL`). Este valor debe ser único dentro del dashboard al que pertenece el widget. Comportamiento del parámetro: - *obligatorio*

Ejemplos

Los siguientes ejemplos tienen como objetivo describir únicamente la configuración de los objetos de campo del widget de panel para el widget *Geomap*. Para obtener más información sobre la configuración de un panel, consulte `dashboard.create`.

Configuración de un widget *Geomap*

Configure un widget *Geomap* que muestre equipos de los grupos de equipos ”2” y ”22” según la siguiente configuración de etiquetas: la etiqueta con el nombre ”component” contiene el valor ”node”, y la etiqueta con el nombre ”location” es igual al valor ”New York”. Además, establezca la vista inicial del mapa en las coordenadas ”40.6892494” (latitud), ”-74.0466891” (longitud) con el nivel de zoom ”10”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "geomap",
                        "name": "Geomap",
                        "x": 0,
                        "y": 0,
                        "width": 36,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 2,
                                "name": "groupids.0",
                                "value": 22
                            },
                            {
                                "type": 2,
                                "name": "groupids.1",
                                "value": 2
                            },
                            {
                                "type": 1,
                                "name": "default_view",
                                "value": "40.6892494,-74.0466891,10"
                            },
                            {
                                "type": 0,
                                "name": "evaltype",
                                "value": 2
                            },
                            {
                                "type": 1,
                                "name": "tags.0.tag",
                                "value": "component"
                            },
                            {
                                "type": 0,
                                "name": "tags.0.operator",
                                "value": 0
                            },
                            {
                                "type": 1,
                                "name": "tags.0.value",
                                "value": "node"
                            },
                            {
                                "type": 1,
                                "name": "tags.1.tag",
```

```
                                "value": "location"
                            },
                            {
                                "type": 0,
                                "name": "tags.1.operator",
                                "value": 1
                            },
                            {
                                "type": 1,
                                "name": "tags.1.value",
                                "value": "New York"
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```


Respuesta:


```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```


Véase también

- Campo de widget de panel
- `dashboard.create`
- `dashboard.update`

8 Gráfico

Descripción

Estos parámetros y los posibles valores de propiedad para los objetos de campo correspondientes del widget del dashboard permiten configurar el widget *Graph* en los métodos `dashboard.create` y `dashboard.update`.

Attention:
Las propiedades `fields` del widget no se validan durante la creación o actualización de un dashboard. Esto permite a los usuarios modificar widgets integrados y crear widgets personalizados, pero también introduce el riesgo de crear o actualizar widgets de forma incorrecta. Para garantizar la creación o actualización correctas del widget *Graph*, consulte el comportamiento de los parámetros descrito en las tablas siguientes.

Parámetros

Los siguientes parámetros son compatibles para el *widget de Gráfico*.

Parámetro	tipo	nombre	valor
Intervalo de actualización	0	rf_rate	0 - Sin actualización; 10 - 10 segundos; 30 - 30 segundos; 60 - *(por defecto)* 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - 15 minutos.
Referencia	1	reference	Cualquier valor de cadena que consista en 5 caracteres (por ejemplo, `ABCDE` o `JBPNL`). Este valor debe ser único dentro del panel al que pertenece el widget. Comportamiento del parámetro: - *obligatorio*

Conjunto de datos

Los siguientes parámetros son compatibles con la configuración de un *Conjunto de datos*.

Note:
El primer número del nombre de la propiedad (por ejemplo, ds.0.hosts.0, ds.0.items.0) representa el conjunto de datos correspondiente, mientras que el segundo número, si está presente, representa el host o el item configurado.

Parámetro	type	name	value
Tipo de conjunto de datos	0	ds.0.dataset_type	0 - Lista de items; 1 - Patrones de items *(predeterminado)*.
Items	4	ds.0.itemids.0	ID del item. Al configurar el widget en un dashboard de template, solo deben establecerse los items configurados en el template. Nota: Para configurar varios items, cree un objeto de campo de widget del dashboard para cada item, incrementando el número en el nombre de la propiedad. Comportamiento del parámetro: - *obligatorio* si *Tipo de conjunto de datos* está establecido en "Lista de items" y *Items (Widget)* no está establecido

Parámetro	type	name	value
Items (Widget)	1	ds.0.itemids.0._reference	En lugar del ID del item: `ABCDE._itemid` - establezca un widget compatible (con su parámetro *Reference* establecido en "`ABCDE`") como fuente de datos para los items. Nota: Para configurar varios widgets, cree un objeto de campo de widget del dashboard para cada widget, incrementando el número en el nombre de la propiedad. Comportamiento del parámetro: - *obligatorio* si *Tipo de conjunto de datos* está establecido en "Lista de items" y *Items* no está establecido
Color	1	ds.0.color.0	Código de color hexadecimal (por ejemplo, `FF0000`). Comportamiento del parámetro: - *obligatorio* si *Tipo de conjunto de datos* está establecido en "Lista de items"
Patrones de hosts	1	ds.0.hosts.0	Nombre o patrón de host (por ejemplo, "Zabbix*"). Comportamiento del parámetro: - *obligatorio* si *Tipo de conjunto de datos* está establecido en "Patrones de items" Este parámetro no es compatible al configurar el widget en un dashboard de template.
Patrones de items	1	ds.0.items.0	Nombre o patrón de item (por ejemplo, "*: Número de *valores procesados por segundo"). Al configurar el widget en un dashboard de template, solo deben establecerse los patrones de los items configurados en el template. Comportamiento del parámetro: - *obligatorio* si *Tipo de conjunto de datos* está establecido en "Patrones de items"
Color	1	ds.0.color	Código de color hexadecimal (por ejemplo, `FF0000`). Comportamiento del parámetro: - *compatible* si *Tipo de conjunto de datos* está establecido en "Patrones de items" y *Paleta de colores* no está establecida
Paleta de colores	0	ds.0.color_palette	Índice de la paleta de colores. Los valores posibles van de 0 a 11. Predeterminado: 0. Comportamiento del parámetro: - *compatible* si *Tipo de conjunto de datos* está establecido en "Patrones de items" y *Color* no está establecido
Dibujar	0	ds.0.type	0 - Línea *(predeterminado)*; 1 - Puntos; 2 - Escalones; 3 - Barras.
Apilado	0	ds.0.stacked	0 - Deshabilitado *(predeterminado)*; 1 - Habilitado. Comportamiento del parámetro: - *compatible* si *Dibujar* está establecido en "Línea", "Escalones" o "Barras"

Parámetro	type	name	value
Ancho	0	ds.0.width	Los valores posibles van de 1 a 10. Predeterminado: 1. Comportamiento del parámetro: - *compatible* si *Dibujar* está establecido en ”Línea” o ”Escalones”
Tamaño de punto	0	ds.0.pointsize	Los valores posibles van de 1 a 10. Predeterminado: 3. Comportamiento del parámetro: - *compatible* si *Dibujar* está establecido en ”Puntos”
Transparencia	0	ds.0.transparency	Los valores posibles van de 1 a 10. Predeterminado: 5.
Relleno	0	ds.0.fill	Los valores posibles van de 1 a 10. Predeterminado: 3. Comportamiento del parámetro: - *compatible* si *Dibujar* está establecido en ”Línea” o ”Escalones”
Datos faltantes	0	ds.0.missingdatafunc	0 - Ninguno *(predeterminado)*; 1 - Conectados; 2 - Tratar como 0; 3 - Último conocido. Comportamiento del parámetro: - *compatible* si *Dibujar* está establecido en ”Línea” o ”Escalones”
Anular host	1	ds.0.override_hostid	`ABCDE` - establezca un widget compatible (con su parámetro ”Reference” establecido en ”`ABCDE`”) como fuente de datos para los hosts; `DASHBOARD._hostid` - establezca el selector de hosts del dashboard como fuente de datos para los hosts. Este parámetro no es compatible al configurar el widget en un dashboard de template.
Eje Y	0	ds.0.axisy	0 - Izquierdo *(predeterminado)*; 1 - Derecho.
Invertir valores	0	ds.0.invert_values	0 - Deshabilitado (predeterminado); 1 - Habilitado.
Desplazamiento temporal	1	ds.0.timeshift	Cadena de tiempo válida (por ejemplo, `3600`, `1h`, etc.). Puede utilizar sufijos de tiempo. También se permiten valores negativos. Predeterminado: `""` (vacío).
Función de agregación	0	ds.0.aggregate_function	0 - no utilizada *(predeterminado)*; 1 - min; 2 - max; 3 - avg; 4 - count; 5 - sum; 6 - first; 7 - last.
Intervalo de agregación	1	ds.0.aggregate_interval	Cadena de tiempo válida (por ejemplo, `3600`, `1h`, etc.). Puede utilizar sufijos de tiempo. Predeterminado: `1h`.

Parámetro	type	name	value
Agregar	0	ds.0.aggregate_grouping	0 - Cada item *(predeterminado)*; 1 - Conjunto de datos. Comportamiento del parámetro: - *compatible* si *Función de agregación* está establecida en ”min”, ”max”, ”avg”, ”count”, ”sum”, ”first” o ”last”
Aproximación	0	ds.0.approximation	1 - min; 2 - avg *(predeterminado)*; 4 - max; 7 - todos.
Etiqueta del conjunto de datos	1	ds.0.data_set_label	Cualquier valor de cadena. Predeterminado: "" (vacío).
Etiquetas de items			
Tipo de evaluación	0	ds.0.item_tags_evaltype	0 - Y/O *(predeterminado)*; 2 - O.
Nombre de etiqueta	1	ds.0.item_tags.0.tag	Cualquier valor de cadena. Nota: El número del nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* al configurar *Etiquetas de items*
Operador	0	ds.0.item_tags.0.operator	0 - Contiene; 1 - Es igual a; 2 - No contiene; 3 - No es igual a; 4 - Existe; 5 - No existe. Nota: El número del nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* al configurar *Etiquetas de items*
Valor de etiqueta	1	ds.0.item_tags.0.value	Cualquier valor de cadena. Nota: El número del nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* al configurar *Etiquetas de items*

Opciones de visualización

Los siguientes parámetros son compatibles con la configuración de *Opciones de visualización*.

Parámetro	type	name	value
Selección de datos históricos	0	source	0 - *(predeterminado)* Automático; 1 - Historial; 2 - Tendencias.

Parámetro	type	name	value
Triggers simples	0	simple_triggers	0 - *(predeterminado)* Deshabilitado; 1 - Habilitado.
Tiempo de trabajo	0	working_time	0 - *(predeterminado)* Deshabilitado; 1 - Habilitado.
Nombres de los hosts en las etiquetas	0	show_hostnames	0 - *(predeterminado)* Automático; 1 - Mostrar; 2 - Ocultar. Este parámetro no es compatible si se configura el widget en un dashboard de template.
Línea de percentil (izquierda)			
Estado	0	percentile_left	0 - *(predeterminado)* Deshabilitado; 1 - Habilitado. Comportamiento del parámetro: - *compatible* si *Eje Y* (en la configuración de *Conjunto de datos*) está establecido en ”Izquierda”
Valor	0	percentile_left_value	Los valores posibles van de 1 a 100. Comportamiento del parámetro: - *compatible* si *Eje Y* (en la configuración de *Conjunto de datos*) está establecido en ”Izquierda”
Línea de percentil (derecha)			
Estado	0	percentile_right	0 - *(predeterminado)* Deshabilitado; 1 - Habilitado. Comportamiento del parámetro: - *compatible* si *Eje Y* (en la configuración de *Conjunto de datos*) está establecido en ”Derecha”
Valor	0	percentile_right_value	Los valores posibles van de 1 a 100. Comportamiento del parámetro: - *compatible* si *Eje Y* (en la configuración de *Conjunto de datos*) está establecido en ”Derecha”

Periodo de tiempo

Se admiten los siguientes parámetros para configurar el *Periodo de tiempo*.

Parámetro	tipo	nombre	valor
Periodo de tiempo	1	time_period._referencia	`DASHBOARD._timeperiod` - establece el selector de periodo de tiempo del dashboard como fuente de datos; `ABCDE._timeperiod` - establece un widget compatible (con su parámetro *Referencia* configurado en "`ABCDE`") como fuente de datos. Por defecto: `DASHBOARD._timeperiod` Alternativamente, puede establecer el periodo de tiempo solo en los parámetros *Desde* y *Hasta*.
Desde	1	time_period.from	Cadena de tiempo válida en sintaxis absoluta (`YYYY-MM-DD hh:mm:ss`) o relativa (`now`, `now/d`, `now/w-1w`, etc.). Comportamiento del parámetro: - *admitido* si *Periodo de tiempo* no está configurado - *obligatorio* si se establece `time_period.to`
Hasta	1	time_period.to	Cadena de tiempo válida en sintaxis absoluta (`YYYY-MM-DD hh:mm:ss`) o relativa (`now`, `now/d`, `now/w-1w`, etc.). Comportamiento del parámetro: - *admitido* si *Periodo de tiempo* no está configurado - *obligatorio* si se establece `time_period.from`

Ejes

Se admiten los siguientes parámetros para configurar los *Ejes*.

Parámetro	tipo	nombre	valor
Y izquierdo	0	lefty	0 - Deshabilitado; 1 - *(por defecto)* Habilitado. Comportamiento del parámetro: - *soportado* si el *Eje Y* (en la configuración del *Conjunto de datos*) está establecido en "Izquierda"
Y derecho	0	righty	0 - *(por defecto)* Deshabilitado; 1 - Habilitado. Comportamiento del parámetro: - *soportado* si el *Eje Y* (en la configuración del *Conjunto de datos*) está establecido en "Derecha"
Escala	0	lefty_scale righty_scale	0 - *(por defecto)* Lineal; 1 - Logarítmica.
Mín	1	lefty_min righty_min	Cualquier valor numérico. Por defecto: "" (vacío).
Máx	1	lefty_max righty_max	Cualquier valor numérico. Por defecto: "" (vacío).
Unidades (tipo)	0	lefty_units righty_units	0 - *(por defecto)* Automático; 1 - Estático.
Unidades (valor)	1	lefty_static_units righty_static_units	Cualquier valor de cadena. Por defecto: "" (vacío).
Eje X	0	xaxis	0 - Deshabilitado; 1 - *(por defecto)* Habilitado.

Leyenda

Los siguientes parámetros son compatibles para configurar la *Leyenda*.

Parámetro	tipo	nombre	valor
Mostrar leyenda	0	legend	0 - Deshabilitada; 1 - *(predeterminado)* Habilitada.
Mostrar mínimo/promedio/máximo	0	legend_statistic	0 - *(predeterminado)* Deshabilitado; 1 - Habilitado. Comportamiento del parámetro: - *compatible* si *Mostrar leyenda* está establecido en ”Habilitada”
Mostrar función de agregación	0	legend_aggregation	0 - *(predeterminado)* Deshabilitada; 1 - Habilitada. Comportamiento del parámetro: - *compatible* si *Mostrar leyenda* está establecido en ”Habilitada”
Filas	0	legend_lines_mode	0 - *(predeterminado)* Fijas; 1 - Variables. Comportamiento del parámetro: - *compatible* si *Mostrar leyenda* está establecido en ”Habilitada”
Número de filas/ Número máximo de filas	0	legend_lines	Los valores posibles van de 1 a 10. Predeterminado: 1. Comportamiento del parámetro: - *compatible* si *Mostrar leyenda* está establecido en ”Habilitada”
Número de columnas	0	legend_columns	Los valores posibles van de 1 a 4. Predeterminado: 4. Comportamiento del parámetro: - *compatible* si *Mostrar leyenda* está establecido en ”Habilitada” y *Mostrar mínimo/promedio/máximo* está establecido en ”Deshabilitado”

Problemas

Los siguientes parámetros son compatibles para configurar *Problemas*.

Parámetro	tipo	nombre	valor
Mostrar problemas	0	show_problems	0 - *(por defecto)* Deshabilitado; 1 - Habilitado.
Solo métricas seleccionadas	0	graph_item_problems	0 - Deshabilitado; 1 - *(por defecto)* Habilitado.
Equipos con problemas	1	problemhosts.0	Nombre del equipo. Nota: El número en el nombre de la propiedad hace referencia al equipo configurado. Para configurar varios equipos, cree un objeto de campo de widget de panel para cada equipo con un número incrementado en el nombre de la propiedad. Este parámetro no es compatible si se configura el widget en un panel de plantilla.

Parámetro	tipo	nombre	valor
Severidad	0	severities.0	0 - No clasificado; 1 - Información; 2 - Advertencia; 3 - Media; 4 - Alta; 5 - Desastre. Por defecto: vacío (todas habilitadas). Nota: Para configurar varios valores, cree un objeto de campo de widget de panel para cada valor con un número incrementado en el nombre de la propiedad.
Problema	1	problem_name	Nombre del evento del problema (no distingue mayúsculas/minúsculas, nombre completo o parte de él).
Etiquetas de problema			
Tipo de evaluación	0	evaltype	0 - *(por defecto)* Y/O; 2 - O.
Nombre de etiqueta	1	tags.0.tag	Cualquier valor de cadena. Nota: El número en el nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* si se configuran *Etiquetas de problema*
Operador	0	tags.0.operator	0 - Contiene; 1 - Igual a; 2 - No contiene; 3 - No igual a; 4 - Existe; 5 - No existe. Nota: El número en el nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* si se configuran *Etiquetas de problema*
Valor de etiqueta	1	tags.0.value	Cualquier valor de cadena. Nota: El número en el nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* si se configuran *Etiquetas de problema*

Anulaciones

Los siguientes parámetros son compatibles con la configuración de las *Anulaciones*.

Note:
El primer número del nombre de la propiedad (por ejemplo, or.0.hosts.0, or.0.items.0) representa el conjunto de datos correspondiente, mientras que el segundo número, si está presente, representa el host o el item configurado.

Parámetro	type	name	value
Patrones de host	1	or.0.hosts.0	Nombre o patrón de host (por ejemplo, `Zabbix*`). Este parámetro no es compatible al configurar el widget en un dashboard de template. Comportamiento del parámetro: - *obligatorio* al configurar *Anulaciones*
Patrones de item	1	or.0.items.0	Nombre o patrón de item (por ejemplo, `*: Number of processed *values per second`). Al configurar el widget en un dashboard de template, solo deben establecerse los patrones de los items configurados en el template. Comportamiento del parámetro: - *obligatorio* al configurar *Anulaciones*
Color base	1	or.0.color	Código de color hexadecimal (por ejemplo, FF0000).
Ancho	0	or.0.width	Los valores posibles van de 1 a 10.
Dibujar	0	or.0.type	0 - Línea; 1 - Puntos; 2 - Escalones; 3 - Barra.
Transparencia	0	or.0.transparency	Los valores posibles van de 1 a 10.
Relleno	0	or.0.fill	Los valores posibles van de 1 a 10.
Tamaño del punto	0	or.0.pointsize	Los valores posibles van de 1 a 10.
Datos faltantes	0	or.0.missingdatafunc	0 - Ninguno; 1 - Conectado; 2 - Tratar como 0; 3 - Último conocido.
Eje Y	0	or.0.axisy	0 - Izquierdo; 1 - Derecho.
Invertir valores	0	or.0.invert_values	0 - (predeterminado) Deshabilitado; 1 - Habilitado.
Desplazamiento temporal	1	or.0.timeshift	Cadena de tiempo válida (por ejemplo, 3600, 1h, etc.). Puede utilizar sufijos de tiempo. Se permiten valores negativos.

Ejemplos

Los siguientes ejemplos tienen como objetivo describir únicamente la configuración de los objetos de campo del widget de panel para el widget *Gráfico*. Para obtener más información sobre la configuración de un panel, consulte `dashboard.create`.

Configuración de un widget *Gráfico*

Configure un widget *Gráfico* de la siguiente manera:

- 2 conjuntos de datos para un total de 9 items en 1 host.
- El primer conjunto de datos es de tipo ”Lista de items” y consta de 3 items que se representan mediante líneas de diferente color, pero con el mismo ancho, transparencia y relleno.
- El segundo conjunto de datos es de tipo ”Patrones de items”, consta de 6 items, tiene una agregación configurada y se representa mediante una línea con un color, ancho, transparencia y relleno personalizados.
- El segundo conjunto de datos también tiene una etiqueta de conjunto de datos personalizada.
- Los datos en el gráfico se muestran para un período de tiempo de las últimas 3 horas.
- Los problemas en el gráfico se muestran solo para los items configurados.
- El gráfico tiene dos ejes Y, de los cuales el eje Y derecho muestra valores solo para el segundo conjunto de datos.
- La leyenda del gráfico muestra los items configurados en 4 filas, así como los valores mínimo, máximo y promedio de los conjuntos de datos.

Request:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": 30,
        "auto_start": 1,
```

```
        "pages": [
            {
                "widgets": [
                    {
                        "type": "svggraph",
                        "name": "Graph",
                        "x": 0,
                        "y": 0,
                        "width": 36,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 0,
                                "name": "ds.0.dataset_type",
                                "value": 0
                            },
                            {
                                "type": 4,
                                "name": "ds.0.itemids.0",
                                "value": 23264
                            },
                            {
                                "type": 1,
                                "name": "ds.0.color.0",
                                "value": "FF0000"
                            },
                            {
                                "type": 4,
                                "name": "ds.0.itemids.1",
                                "value": 23269
                            },
                            {
                                "type": 1,
                                "name": "ds.0.color.1",
                                "value": "BF00FF"
                            },
                            {
                                "type": 4,
                                "name": "ds.0.itemids.2",
                                "value": 23257
                            },
                            {
                                "type": 1,
                                "name": "ds.0.color.2",
                                "value": "0040FF"
                            },
                            {
                                "type": 0,
                                "name": "ds.0.width",
                                "value": 3
                            },
                            {
                                "type": 0,
                                "name": "ds.0.transparency",
                                "value": 3
                            },
                            {
                                "type": 0,
                                "name": "ds.0.fill",
                                "value": 1
                            },
```

```
                {
                    "type": 1,
                    "name": "ds.1.hosts.0",
                    "value": "Zabbix server"
                },
                {
                    "type": 1,
                    "name": "ds.1.items.0",
                    "value": "*: Number of processed *values per second"
                },
                {
                    "type": 1,
                    "name": "ds.1.color",
                    "value": "000000"
                },
                {
                    "type": 0,
                    "name": "ds.1.transparency",
                    "value": 0
                },
                {
                    "type": 0,
                    "name": "ds.1.fill",
                    "value": 0
                },
                {
                    "type": 0,
                    "name": "ds.1.axisy",
                    "value": 1
                },
                {
                    "type": 0,
                    "name": "ds.1.aggregate_function",
                    "value": 3
                },
                {
                    "type": 1,
                    "name": "ds.1.aggregate_interval",
                    "value": "1m"
                },
                {
                    "type": 0,
                    "name": "ds.1.aggregate_grouping",
                    "value": 1
                },
                {
                    "type": 1,
                    "name": "ds.1.data_set_label",
                    "value": "Number of processed values per second"
                },
                {
                    "type": 0,
                    "name": "graph_time",
                    "value": 1
                },
                {
                    "type": 1,
                    "name": "time_period.from",
                    "value": "now-3h"
                },
                {
                    "type": 1,
```

```
                                "name": "time_period.to",
                                "value": "now"
                            },
                            {
                                "type": 0,
                                "name": "legend_statistic",
                                "value": 1
                            },
                            {
                                "type": 0,
                                "name": "legend_lines",
                                "value": 4
                            },
                            {
                                "type": 0,
                                "name": "show_problems",
                                "value": 1
                            },
                            {
                                "type": 1,
                                "name": "reference",
                                "value": "YZABC"
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Véase también

- Campo de widget de panel
- dashboard.create
- dashboard.update

9 Gráfico (clásico)

Descripción

Estos parámetros y los posibles valores de propiedad para los objetos de campo del widget del dashboard correspondiente permiten configurar el widget *Graph (classic)* en los métodos `dashboard.create` y `dashboard.update`.

Attention:
Las propiedades `fields` del widget no se validan durante la creación o actualización de un dashboard. Esto permite a los usuarios modificar widgets integrados y crear widgets personalizados, pero también introduce el riesgo de crear o actualizar widgets de forma incorrecta. Para garantizar la creación o actualización correctas del widget *Graph (classic)*, consulte el comportamiento de los parámetros descrito en las tablas siguientes.

Parámetros

Los siguientes parámetros son compatibles con el widget *Gráfico (clásico)*.

Parámetro	tipo	nombre	valor
Intervalo de actualización	0	rf_rate	0 - Sin actualización; 10 - 10 segundos; 30 - 30 segundos; 60 - *(por defecto)* 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - 15 minutos.
Origen	0	source_type	0 - *(por defecto)* Gráfico; 1 - Gráfico simple.
Gráfico	6	graphid.0	ID de Gráfico. Comportamiento del parámetro: - *requerido* si *Origen* está configurado como "Gráfico"
Gráfico (Widget)	1	graphid._reference	En lugar del ID de Gráfico: `ABCDE._graphid` - establezca un widget compatible (con su parámetro *Referencia* configurado como "`ABCDE`") como fuente de datos para los gráficos. Comportamiento del parámetro: - *requerido* si *Origen* está configurado como "Gráfico simple" y *Gráfico* no está configurado
Item	4	itemid.0	ID de Item. Comportamiento del parámetro: - *requerido* si *Origen* está configurado como "Gráfico simple" y *Item (Widget)* no está configurado
Item (Widget)	1	itemid._reference	En lugar del ID de Item: `ABCDE._itemid` - establezca un widget compatible (con su parámetro *Referencia* configurado como "`ABCDE`") como fuente de datos para los items. Comportamiento del parámetro: - *requerido* si *Origen* está configurado como "Gráfico simple" y *Item* no está configurado
Período de tiempo	1	time_period._reference	`DASHBOARD._timeperiod` - establezca el selector de período de tiempo del dashboard como fuente de datos; `ABCDE._timeperiod` - establezca un widget compatible (con su parámetro *Referencia* configurado como "`ABCDE`") como fuente de datos. Por defecto: `DASHBOARD._timeperiod` Alternativamente, puede establecer el período de tiempo solo en los parámetros *Desde* y *Hasta*.

Parámetro	tipo	nombre	valor
Desde	1	time_period.from	Cadena de tiempo válida en sintaxis absoluta (`YYYY-MM-DD hh:mm:ss`) o relativa (`now`, `now/d`, `now/w-1w`, etc.). Comportamiento del parámetro: - *compatible* si *Período de tiempo* no está configurado - *requerido* si `time_period.to` está configurado
Hasta	1	time_period.to	Cadena de tiempo válida en sintaxis absoluta (`YYYY-MM-DD hh:mm:ss`) o relativa (`now`, `now/d`, `now/w-1w`, etc.). Comportamiento del parámetro: - *compatible* si *Período de tiempo* no está configurado - *requerido* si `time_period.from` está configurado
Mostrar leyenda	0	show_legend	0 - Deshabilitado; 1 - *(por defecto)* Habilitado.
Sobrescribir host	1	override_hostid._reference	`ABCDE._hostid` - establezca un widget compatible (con su parámetro *Referencia* configurado como "`ABCDE`") como fuente de datos para los hosts; `DASHBOARD._hostid` - establezca el selector de host del dashboard como fuente de datos para los hosts. Este parámetro no es compatible si se configura el widget en un dashboard de template.
Referencia	1	reference	Cualquier valor de cadena que consista en 5 caracteres (por ejemplo, `ABCDE` o `JBPNL`). Este valor debe ser único dentro del dashboard al que pertenece el widget. Comportamiento del parámetro: - *requerido*

Ejemplos

Los siguientes ejemplos tienen como objetivo describir únicamente la configuración de los objetos de campo del widget de panel para el widget *Gráfico (clásico)*. Para obtener más información sobre la configuración de un panel, consulte `dashboard.create`.

Configuración de un widget *Gráfico (clásico)*

Configure un widget *Gráfico (clásico)* que muestre un gráfico simple para la métrica "42269".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "graph",
                        "name": "Graph (classic)",
                        "x": 0,
                        "y": 0,
                        "width": 36,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 0,
                                "name": "source_type",
                                "value": 1
```

```
                            },
                            {
                                "type": 4,
                                "name": "itemid.0",
                                "value": 42269
                            },
                            {
                                "type": 1,
                                "name": "reference",
                                "value": "RSTUV"
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Véase también

- Campo de widget de panel
- `dashboard.create`
- `dashboard.update`

10 Prototipo de gráfico

Descripción

Estos parámetros y los posibles valores de propiedad para los objetos de campo del widget de dashboard correspondiente permiten configurar el widget *Prototipo de gráfico* en los métodos `dashboard.create` y `dashboard.update`.

Attention:
Las propiedades `fields` del widget no se validan durante la creación o actualización de un dashboard. Esto permite a los usuarios modificar widgets integrados y crear widgets personalizados, pero también introduce el riesgo de crear o actualizar widgets de forma incorrecta. Para garantizar la creación o actualización correcta del widget *Prototipo de gráfico*, consulte el comportamiento de los parámetros descrito en las tablas siguientes.

Parámetros

Los siguientes parámetros son compatibles con el widget *Prototipo de gráfico*.

Parámetro	tipo	nombre	valor
Intervalo de actualización	0	rf_rate	0 - Sin actualización; 10 - 10 segundos; 30 - 30 segundos; 60 - *(predeterminado)* 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - 15 minutos.
Origen	0	source_type	2 - *(predeterminado)* Prototipo de gráfico; 3 - Prototipo de gráfico simple.
Prototipo de gráfico	7	graphid.0	ID del prototipo de gráfico. Comportamiento del parámetro: - *obligatorio* si *Origen* está establecido en ”Prototipo de gráfico”
Prototipo de item	5	itemid.0	ID del prototipo de item. Comportamiento del parámetro: - *obligatorio* si *Origen* está establecido en ”Prototipo de gráfico simple”
Período de tiempo	1	time_period._referenc	`DASHBOARD._timeperiod` - establece el selector de período de tiempo del dashboard como origen de datos; `ABCDE._timeperiod` - establece un widget compatible (con su parámetro *Referencia* establecido en ”ABCDE”) como origen de datos. Predeterminado: `DASHBOARD._timeperiod` Como alternativa, puede establecer el período de tiempo únicamente en los parámetros *Desde* y *Hasta*.
Desde	1	time_period.from	Cadena de tiempo válida en sintaxis absoluta (`YYYY-MM-DD hh:mm:ss`) o relativa (`now`, `now/d`, `now/w-1w`, etc.). Comportamiento del parámetro: - *compatible* si no se establece *Período de tiempo* - *obligatorio* si se establece `time_period.to`
Hasta	1	time_period.to	Cadena de tiempo válida en sintaxis absoluta (`YYYY-MM-DD hh:mm:ss`) o relativa (`now`, `now/d`, `now/w-1w`, etc.). Comportamiento del parámetro: - *compatible* si no se establece *Período de tiempo* - *obligatorio* si se establece `time_period.from`
Mostrar leyenda	0	show_legend	0 - Deshabilitado; 1 - *(predeterminado)* Habilitado.
Anular host	1	override_hostid._refere	`ABCDE._hostid` - establece un widget compatible (con su parámetro *Referencia* establecido en ”ABCDE”) como origen de datos para los hosts; `DASHBOARD._hostid` - establece el selector de host del dashboard como origen de datos para los hosts. Este parámetro no es compatible al configurar el widget en un dashboard de template.
Columnas	0	columns	Los valores posibles van de 1 a 24. Predeterminado: 2.
Filas	0	rows	Los valores posibles van de 1 a 16. Predeterminado: 1.

Parámetro	tipo	nombre	valor
Referencia	1	reference	Cualquier valor de cadena compuesto por 5 caracteres (por ejemplo, `ABCDE` o `JBPNL`). Este valor debe ser único dentro del dashboard al que pertenece el widget. Comportamiento del parámetro: - *obligatorio*

Ejemplos

Los siguientes ejemplos tienen como objetivo describir únicamente la configuración de los objetos de campo del widget de panel para el widget *Prototipo de gráfico*. Para obtener más información sobre la configuración de un panel, consulte `dashboard.create`.

Configuración de un widget *Prototipo de gráfico*

Configure un widget *Prototipo de gráfico* que muestre una cuadrícula de 3 gráficos (3 columnas, 1 fila) creados a partir de una métrica prototipo (ID: ”42316”) mediante descubrimiento de bajo nivel.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "Mi panel",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "graphprototype",
                        "name": "Prototipo de gráfico",
                        "x": 0,
                        "y": 0,
                        "width": 48,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 0,
                                "name": "source_type",
                                "value": 3
                            },
                            {
                                "type": 5,
                                "name": "itemid.0",
                                "value": 42316
                            },
                            {
                                "type": 0,
                                "name": "columns",
                                "value": 3
                            },
                            {
                                "type": 1,
                                "name": "reference",
                                "value": "OPQWX"
                            }
                        ]
                    }
                ]
            }
```

```
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Véase también

- Campo de widget de panel
- `dashboard.create`
- `dashboard.update`

11 Honeycomb

Descripción

Estos parámetros y los posibles valores de las propiedades de los objetos de campo del widget del dashboard correspondiente permiten configurar el widget *Honeycomb* en los métodos `dashboard.create` y `dashboard.update`.

Attention:
Las propiedades `fields` del widget no se validan durante la creación o actualización de un dashboard. Esto permite a los usuarios modificar widgets integrados y crear widgets personalizados, pero también introduce el riesgo de crear o actualizar widgets incorrectamente. Para garantizar la creación o actualización correctas del widget *Honeycomb*, consulte el comportamiento de los parámetros descrito en las tablas siguientes.

Parámetros

Se admiten los siguientes parámetros para el widget *Honeycomb*.

Parámetro	tipo	nombre	valor
Intervalo de actualización	0	rf_rate	0 - Sin actualización; 10 - 10 segundos; 30 - 30 segundos; 60 - *(por defecto)* 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - 15 minutos.

Parámetro	tipo	nombre	valor
Grupos de hosts	2	groupids.0	ID de Grupo de hosts. Nota: Para configurar varios grupos de hosts, cree un objeto de campo de widget de panel para cada grupo de hosts con un número incrementado en el nombre de la propiedad. Este parámetro no es compatible si se configura el widget en un panel de plantilla.
Grupos de hosts (Widget)	1	groupids._reference	En lugar del ID de Grupo de hosts: `ABCDE._hostgroupids` - establezca un widget compatible (con su parámetro *Referencia* configurado en "`ABCDE`") como fuente de datos para los grupos de hosts. Este parámetro no es compatible si se configura el widget en un panel de plantilla.
Hosts	3	hostids.0	ID de Host. Nota: Para configurar varios hosts, cree un objeto de campo de widget de panel para cada host con un número incrementado en el nombre de la propiedad. Para varios hosts, el parámetro *Grupos de hosts* debe no estar configurado en absoluto o configurado con al menos un grupo de hosts al que pertenezcan los hosts configurados. Este parámetro no es compatible si se configura el widget en un panel de plantilla.
Hosts (Widget/Panel)	1	hostids._reference	En lugar del ID de Host: `DASHBOARD.hostids` - establezca el selector de hosts del panel como fuente de datos para los hosts; `ABCDE._hostids` - establezca un widget compatible (con su parámetro *Referencia* configurado en "`ABCDE`") como fuente de datos para los hosts. Este parámetro no es compatible si se configura el widget en un panel de plantilla.
Etiquetas de host			
Tipo de evaluación	0	evaltype_host	0 - *(por defecto)* Y/O; 2 - O. Este parámetro no es compatible si se configura el widget en un panel de plantilla.
Nombre de etiqueta	1	host_tags.0.tag	Cualquier valor de cadena. Nota: El número en el nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* si se configuran *Etiquetas de host* Este parámetro no es compatible si se configura el widget en un panel de plantilla.

Parámetro	tipo	nombre	valor
Operador	0	host_tags.0.operator	0 - Contiene; 1 - Igual a; 2 - No contiene; 3 - No igual a; 4 - Existe; 5 - No existe. Nota: El número en el nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* si se configuran *Etiquetas de host* Este parámetro no es compatible si se configura el widget en un panel de plantilla.
Valor de etiqueta	1	host_tags.0.value	Cualquier valor de cadena. Nota: El número en el nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* si se configuran *Etiquetas de host* Este parámetro no es compatible si se configura el widget en un panel de plantilla.
Patrones de ítem	1	items.0	Nombre o patrón de Ítem. Nota: Para configurar varios patrones de ítem, cree un objeto de campo de widget de panel para cada patrón de ítem con un número incrementado en el nombre de la propiedad. Comportamiento del parámetro: - *obligatorio*
Etiquetas de ítem			
Tipo de evaluación	0	evaltype_item	0 - *(por defecto)* Y/O; 2 - O.
Nombre de etiqueta	1	item_tags.0.tag	Cualquier valor de cadena. Nota: El número en el nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* si se configuran *Etiquetas de ítem*
Operador	0	item_tags.0.operator	0 - Contiene; 1 - Igual a; 2 - No contiene; 3 - No igual a; 4 - Existe; 5 - No existe. Nota: El número en el nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* si se configuran *Etiquetas de ítem*

Parámetro	tipo	nombre	valor
Valor de etiqueta	1	item_tags.0.value	Cualquier valor de cadena. Nota: El número en el nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* si se configuran *Etiquetas de ítem*
Mostrar hosts en mantenimiento	0	maintenance	0 - *(por defecto)* Deshabilitado; 1 - Habilitado.
Mostrar	0	show.0	1 - Etiqueta principal; 2 - Etiqueta secundaria. Nota: Para configurar varios valores, cree un objeto de campo de widget de panel para cada valor con un número incrementado en el nombre de la propiedad. Por defecto: 1, 2.
Referencia	1	reference	Cualquier valor de cadena que consista en 5 caracteres (por ejemplo, `ABCDE` o `JBPNL`). Este valor debe ser único dentro del panel al que pertenece el widget. Comportamiento del parámetro: - *obligatorio*

Configuración avanzada

Los siguientes parámetros de configuración avanzada son compatibles con el widget *Honeycomb*.

Note:
El número en el nombre de la propiedad *Thresholds* (por ejemplo, thresholds.0.color) hace referencia a la posición del umbral en una lista ordenada de forma ascendente. Sin embargo, si los umbrales se configuran en un orden diferente, los valores se ordenarán de forma ascendente después de actualizar la configuración del widget en el frontend de Zabbix (por ejemplo, `"thresholds.0.threshold":"5"` → `"thresholds.0.threshold":"1"`; `"thresholds.1.threshold":"1"` → `"thresholds.1.threshold": "5"`).

Parámetro	type	name	value
Etiqueta principal			
Tipo	0	primary_label_type	0 - *(predeterminado)* Texto; 1 - Valor.
Texto	1	primary_label	Cualquier valor de cadena, incluidas macros. Macros compatibles: `{HOST.*}`, `{ITEM.*}`, `{INVENTORY.*}`, macros de usuario. Predeterminado: `{HOST.NAME}` Comportamiento del parámetro: - *compatible* si *Tipo* está establecido en "Texto"
Posiciones decimales	0	primary_label_decimal_places	Los valores posibles van de 0 a 6. Predeterminado: 2. Comportamiento del parámetro: - *compatible* si *Tipo* está establecido en "Valor"
Tamaño (tipo)	0	primary_label_size_type	0 - *(predeterminado)* Automático; 1 - Personalizado.

Parámetro	type	name	value
Tamaño	0	primary_label_size	Los valores posibles van de 1 a 100. Predeterminado: 20. Comportamiento del parámetro: - *compatible* si *Tamaño* (tipo) está establecido en ”Personalizado”
Negrita	0	primary_label_bold	0 - *(predeterminado)* Deshabilitado; 1 - Habilitado.
Color	1	primary_label_color	Código de color hexadecimal (por ejemplo, `FF0000`). Predeterminado: basado en `theme` del objeto Settings y del objeto User: `1F2C33` para ”blue-theme”, ”blue-classic-theme” o ”hc-light”; `EEEEEE` para ”dark-theme”, ”dark-classic-theme” o ”hc-dark”; `E5F1FF` para ”dark-blue-theme”.
Unidades (casilla de verificación)	0	primary_label_units_show	0 - Deshabilitado; 1 - *(predeterminado)* Habilitado. Comportamiento del parámetro: - *compatible* si *Tipo* está establecido en ”Valor”
Unidades (valor)	1	primary_label_units	Cualquier valor de cadena. "" (vacío) Comportamiento del parámetro: - *compatible* si *Tipo* está establecido en ”Valor” y *Unidades* (casilla de verificación) está establecido en ”Habilitado”
Posición	0	primary_label_units_pos	0 - Antes del valor; 1 - *(predeterminado)* Después del valor. Comportamiento del parámetro: - *compatible* si *Tipo* está establecido en ”Valor” y *Unidades* (casilla de verificación) está establecido en ”Habilitado” Este parámetro se ignora si se establece en una de las siguientes unidades relacionadas con el tiempo: unixtime, uptime, s.
Etiqueta secundaria			
Tipo	0	secondary_label_type	0 - Texto; 1 - *(predeterminado)* Valor.
Texto	1	secondary_label	Cualquier valor de cadena, incluidas macros. Macros compatibles: `{HOST.*}`, `{ITEM.*}`, `{INVENTORY.*}`, macros de usuario. Predeterminado: `{{ITEM.LASTVALUE}.fmtnum(2)}` Comportamiento del parámetro: - *compatible* si *Tipo* está establecido en ”Texto”
Posiciones decimales	0	secondary_label_decimals	Los valores posibles van de 0 a 6. Predeterminado: 2. Comportamiento del parámetro: - *compatible* si *Tipo* está establecido en ”Valor”
Tamaño (tipo)	0	secondary_label_size_type	0 - *(predeterminado)* Automático; 1 - Personalizado.
Tamaño	0	secondary_label_size	Los valores posibles van de 1 a 100. Predeterminado: 30. Comportamiento del parámetro: - *compatible* si *Tamaño* (tipo) está establecido en ”Personalizado”

Parámetro	type	name	value
Negrita	0	secondary_label_bold	0 - Deshabilitado; 1 - *(predeterminado)* Habilitado.
Color	1	secondary_label_color	Código de color hexadecimal (por ejemplo, FF0000). Predeterminado: basado en theme del objeto Settings y del objeto User: 1F2C33 para "blue-theme", "blue-classic-theme" o "hc-light"; EEEEEE para "dark-theme", "dark-classic-theme" o "hc-dark"; E5F1FF para "dark-blue-theme".
Unidades (casilla de verificación)	0	secondary_label_units_show	0 - Deshabilitado; 1 - *(predeterminado)* Habilitado. Comportamiento del parámetro: - *compatible* si *Tipo* está establecido en "Valor"
Unidades (valor)	1	secondary_label_units	Cualquier valor de cadena. "" (vacío) Comportamiento del parámetro: - *compatible* si *Tipo* está establecido en "Valor" y *Unidades* (casilla de verificación) está establecido en "Habilitado"
Posición	0	secondary_label_position	0 - Antes del valor; 1 - *(predeterminado)* Después del valor. Comportamiento del parámetro: - *compatible* si *Tipo* está establecido en "Valor" y *Unidades* (casilla de verificación) está establecido en "Habilitado" Este parámetro se ignora si se establece en una de las siguientes unidades relacionadas con el tiempo: unixtime, uptime, s.
Color de fondo			
Color de fondo	1	bg_color	Código de color hexadecimal (por ejemplo, FF0000). Predeterminado: basado en theme del objeto Settings y del objeto User: D9E7ED para "blue-theme" y "blue-classic-theme"; 3D5059 para "dark-theme" y "dark-classic-theme"; 004FA8 para "dark-blue-theme"; AAD7E9 para "hc-light"; 335463 para "hc-dark".
Umbrales			
Interpolación de color	0	interpolation	0 - Deshabilitada; 1 - *(predeterminada)* Habilitada.
Color	1	thresholds.0.color	Código de color hexadecimal (por ejemplo, FF0000).
Umbral	1	thresholds.0.threshold	Cualquier valor numérico. Se admiten sufijos (por ejemplo, "1d", "2w", "4K", "8G").

Ejemplos

Los siguientes ejemplos tienen como objetivo describir únicamente la configuración de los objetos de campo del widget del panel para el widget *Honeycomb*. Para obtener más información sobre la configuración de un panel, consulte dashboard.create.

Configuración de un widget *Honeycomb*

Configure un widget *Honeycomb* que muestre la utilización de los procesos del servidor Zabbix. Además, cambie la etiqueta principal de las celdas del honeycomb y ajuste visualmente el widget con umbrales.

Petición:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "Mi panel",
        "display_period": "30",
```

```
        "auto_start": "1",
        "pages": [
            {
                "widgets": [
                    {
                        "type": "honeycomb",
                        "name": "Honeycomb",
                        "x": "0",
                        "y": "0",
                        "width": "24",
                        "height": "5",
                        "view_mode": "0",
                        "fields": [
                            {
                                "type": 2,
                                "name": "groupids.0",
                                "value": 4
                            },
                            {
                                "type": 3,
                                "name": "hostids.0",
                                "value": 10084
                            },
                            {
                                "type": 1,
                                "name": "items.0",
                                "value": "Zabbix server: Utilization*"
                            },
                            {
                                "type": 1,
                                "name": "primary_label",
                                "value": "{ITEM.NAME}"
                            },
                            {
                                "type": 1,
                                "name": "thresholds.0.color",
                                "value": "0EC9AC"
                            },
                            {
                                "type": 1,
                                "name": "thresholds.0.threshold",
                                "value": "0"
                            },
                            {
                                "type": 1,
                                "name": "thresholds.1.color",
                                "value": "FFD54F"
                            },
                            {
                                "type": 1,
                                "name": "thresholds.1.threshold",
                                "value": "70"
                            },
                            {
                                "type": 1,
                                "name": "thresholds.2.color",
                                "value": "FF465C"
                            },
                            {
                                "type": 1,
                                "name": "thresholds.2.threshold",
                                "value": "90"
```

```
                            },
                            {
                                "type": 1,
                                "name": "reference",
                                "value": "KSTMQ"
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Véase también

- Campo de widget de panel
- `dashboard.create`
- `dashboard.update`

12 Disponibilidad del equipo

Descripción

Estos parámetros y los posibles valores de propiedad para los objetos de campo correspondientes del widget del dashboard permiten configurar el widget *Disponibilidad del host* en los métodos `dashboard.create` y `dashboard.update`.

Parámetros

Los siguientes parámetros son compatibles con el widget *Disponibilidad de equipos*.

Parámetro	tipo	nombre	valor
Intervalo de actualización	0	rf_rate	0 - Sin actualización; 10 - 10 segundos; 30 - 30 segundos; 60 - 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - *(por defecto)* 15 minutos.
Grupos de equipos	2	groupids.0	ID del Grupo de equipos. Nota: Para configurar varios grupos de equipos, cree un objeto de campo de widget de panel para cada grupo de equipos con un número incrementado en el nombre de la propiedad. Este parámetro no es compatible si se configura el widget en un panel de plantilla.
Grupos de equipos (Widget)	1	groupids._reference	En lugar del ID del Grupo de equipos: `ABCDE._hostgroupids` - establezca un widget compatible (con su parámetro *Referencia* configurado en "`ABCDE`") como fuente de datos para los grupos de equipos. Este parámetro no es compatible si se configura el widget en un panel de plantilla.
Tipo de interfaz	0	interface_type.0	0 - Ninguno; 1 - Agente Zabbix (comprobaciones pasivas); 2 - SNMP; 3 - IPMI; 4 - JMX; 5 - Agente Zabbix (comprobaciones activas). Por defecto: 1, 2, 3, 4, 5 (todas habilitadas). Nota: Para configurar varios valores, cree un objeto de campo de widget de panel para cada valor con un número incrementado en el nombre de la propiedad.
Disposición	0	layout	0 - *(por defecto)* Horizontal; 1 - Vertical.
Mostrar equipos en mantenimiento	0	maintenance	0 - *(por defecto)* Deshabilitado; 1 - Habilitado.
Mostrar solo totales	0	only_totals	0 - *(por defecto)* Deshabilitado; 1 - Habilitado.

Ejemplos

Los siguientes ejemplos tienen como objetivo describir únicamente la configuración de los objetos de campo del widget del panel para el widget *Disponibilidad de equipos*. Para obtener más información sobre la configuración de un panel, consulte `dashboard.create`.

Configuración de un widget *Disponibilidad de equipos*

Configure un widget *Disponibilidad de equipos* que muestre información de disponibilidad (en un diseño vertical) para los equipos en el grupo de equipos "4" con interfaces "Zabbix agent" y "SNMP" configuradas.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
```

```
    "params": {
        "name": "My dashboard",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "hostavail",
                        "name": "Host availability",
                        "x": 0,
                        "y": 0,
                        "width": 18,
                        "height": 3,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 2,
                                "name": "groupids.0",
                                "value": 4
                            },
                            {
                                "type": 0,
                                "name": "interface_type",
                                "value": 1
                            },
                            {
                                "type": 0,
                                "name": "interface_type",
                                "value": 2
                            },
                            {
                                "type": 0,
                                "name": "layout",
                                "value": 1
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
```

```
            "3"
        ]
    },
    "id": 1
}
```

Véase también

- Campo de widget de panel
- `dashboard.create`
- `dashboard.update`

13 Tarjeta de equipo

Descripción

Estos parámetros y los posibles valores de propiedad de los objetos de campo del widget del dashboard correspondiente permiten configurar el widget *Host card* en los métodos `dashboard.create` y `dashboard.update`.

Attention:
Las propiedades `fields` del widget no se validan durante la creación o actualización de un dashboard. Esto permite a los usuarios modificar widgets integrados y crear widgets personalizados, pero también introduce el riesgo de crear o actualizar widgets de forma incorrecta. Para garantizar la creación o actualización correctas del widget *Host card*, consulte el comportamiento de los parámetros descrito en las tablas siguientes.

Parámetros

Los siguientes parámetros son compatibles para el widget *Tarjeta de equipo*.

Parámetro	tipo	nombre	valor
Intervalo de actualización	0	rf_rate	0 - Sin actualización; 10 - 10 segundos; 30 - 30 segundos; 60 - *(por defecto)* 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - 15 minutos.
Equipo	3	hostid.0	ID de Equipo. Comportamiento del parámetro: - *obligatorio* si *Equipo (Widget/Panel)* no está establecido Este parámetro no es compatible si se configura el widget en un panel de plantilla.
Equipo (Widget/Panel)	1	hostid._reference	En lugar del ID de Equipo: `DASHBOARD.hostid` - establece el selector de equipo del panel como la fuente de datos para el equipo; `ABCDE._hostid` - establece un widget compatible (con su parámetro *Referencia* configurado en "`ABCDE`") como la fuente de datos para el equipo. Comportamiento del parámetro: - *obligatorio* si *Equipo* no está establecido Este parámetro no es compatible si se configura el widget en un panel de plantilla.

Parámetro	tipo	nombre	valor
Mostrar problemas suprimidos	0	show_suppressed	0 - *(por defecto)* Deshabilitado; 1 - Habilitado.
Mostrar	0	sections.0	0 - Grupos de equipos; 1 - Descripción; 2 - Monitorización; 3 - Disponibilidad; 4 - Monitorizado por; 5 - Plantillas; 6 - Inventario; 7 - Etiquetas. Nota: El número en el nombre de la propiedad hace referencia al orden de la sección en la lista de secciones. Para configurar varias secciones, cree un objeto de campo de widget de panel para cada sección con un número incrementado en el nombre de la propiedad.

Los siguientes parámetros son compatibles si *Mostrar* está configurado en ”Inventario”.

Parámetro	tipo	nombre	valor
Campos de inventario	0	inventory.0	ID de Inventario. Nota: Para configurar varios campos de inventario, cree un objeto de campo de widget de panel para cada campo de inventario con un número incrementado en el nombre de la propiedad.

Ejemplos

Los siguientes ejemplos tienen como objetivo describir únicamente la configuración de los objetos de campo del widget de panel para el widget *Tarjeta de equipo*. Para obtener más información sobre la configuración de un panel, consulte `dashboard.create`.

Configuración de un widget *Tarjeta de equipo*

Configure un widget *Tarjeta de equipo* que muestre estas secciones: ”Supervisión”, ”Disponibilidad”, ”Supervisado por”, ”Inventario” y ”Etiquetas”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "Mi panel",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "hostcard",
                        "name": "Tarjeta de equipo",
                        "x": 0,
                        "y": 0,
                        "width": 14,
                        "height": 7,
                        "view_mode": 0,
                        "fields": [
                            {
```

```
                            "type": 3,
                            "name": "hostid.0",
                            "value": 10084
                        },
                        {
                            "type": 0,
                            "name": "show_suppressed",
                            "value": 1
                        },
                        {
                            "type": 0,
                            "name": "sections.0",
                            "value": 2
                        },
                        {
                            "type": 0,
                            "name": "sections.1",
                            "value": 3
                        },
                        {
                            "type": 0,
                            "name": "sections.2",
                            "value": 4
                        },
                        {
                            "type": 0,
                            "name": "sections.3",
                            "value": 6
                        },
                        {
                            "type": 0,
                            "name": "sections.4",
                            "value": 7
                        },
                        {
                            "type": 0,
                            "name": "inventory.0",
                            "value": 25
                        },
                        {
                            "type": 0,
                            "name": "inventory.1",
                            "value": 26
                        }
                    ]
                }
            ]
        }
    ],
    "userGroups": [
        {
            "usrgrpid": 7,
            "permission": 2
        }
    ],
    "users": [
        {
            "userid": 1,
            "permission": 3
        }
    ]
},
```

```
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Véase también

- Campo de widget de panel
- `dashboard.create`
- `dashboard.update`

14 Navegador de equipos

Descripción

Estos parámetros y los posibles valores de propiedad para los objetos de campo correspondientes del widget del dashboard permiten configurar el widget *Host navigator* en los métodos `dashboard.create` y `dashboard.update`.

Attention:
Las propiedades `fields` del widget no se validan durante la creación o actualización de un dashboard. Esto permite a los usuarios modificar widgets integrados y crear widgets personalizados, pero también introduce el riesgo de crear o actualizar widgets de forma incorrecta. Para garantizar la creación o actualización correctas del widget *Host navigator*, consulte el comportamiento de los parámetros descrito en las tablas siguientes.

Parámetros

Los siguientes parámetros son compatibles con el widget *Navegador de hosts*.

Parámetro	type	name	value
Intervalo de actualización	0	rf_rate	0 - Sin actualización; 10 - 10 segundos; 30 - 30 segundos; 60 - *(predeterminado)* 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - 15 minutos.
Grupos de hosts	2	groupids.0	ID del grupo de hosts. Nota: Para configurar varios grupos de hosts, cree un objeto de campo de widget de dashboard para cada grupo de hosts con un número incrementado en el nombre de la propiedad. Este parámetro no es compatible si se configura el widget en un dashboard de template.
Grupos de hosts (widget)	1	groupids._reference	En lugar del ID del grupo de hosts: `ABCDE._hostgroupids` - establezca un widget compatible (con su parámetro *Referencia* establecido en "ABCDE") como fuente de datos para los grupos de hosts. Este parámetro no es compatible si se configura el widget en un dashboard de template.

Parámetro	type	name	value
Patrones de hosts	1	hosts.0	Nombre o patrón de host. Nota: Para configurar varios patrones de hosts, cree un objeto de campo de widget de dashboard para cada patrón de host con un número incrementado en el nombre de la propiedad. Para varios patrones de hosts, el parámetro *Grupos de hosts* debe no estar configurado o estar configurado con al menos un grupo de hosts al que pertenezcan los hosts que coincidan con los patrones configurados. Este parámetro no es compatible si se configura el widget en un dashboard de template.
Estado del host	0	status	-1 - *(predeterminado)* Cualquiera; 0 - Habilitado; 1 - Deshabilitado. Este parámetro no es compatible si se configura el widget en un dashboard de template.
Etiquetas de host			
Tipo de evaluación	0	host_tags_evaltype	0 - *(predeterminado)* Y/O; 2 - O. Este parámetro no es compatible si se configura el widget en un dashboard de template.
Nombre de etiqueta	1	host_tags.0.tag	Cualquier valor de cadena. Nota: El número del nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* si se configura *Etiquetas de host* Este parámetro no es compatible si se configura el widget en un dashboard de template.
Operador	0	host_tags.0.operator	0 - Contiene; 1 - Es igual a; 2 - No contiene; 3 - No es igual a; 4 - Existe; 5 - No existe. Nota: El número del nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* si se configura *Etiquetas de host* Este parámetro no es compatible si se configura el widget en un dashboard de template.
Valor de etiqueta	1	host_tags.0.value	Cualquier valor de cadena. Nota: El número del nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* si se configura *Etiquetas de host* Este parámetro no es compatible si se configura el widget en un dashboard de template.

Parámetro	type	name	value
Gravedad	0	severities.0	0 - No clasificado; 1 - Información; 2 - Advertencia; 3 - Promedio; 4 - Alto; 5 - Desastre. Predeterminado: vacío (todos habilitados). Nota: Para configurar varios valores, cree un objeto de campo de widget de dashboard para cada valor con un número incrementado en el nombre de la propiedad.
Mostrar hosts en mantenimiento	0	maintenance	0 - *(predeterminado)* Deshabilitado; 1 - Habilitado.
Mostrar problemas	0	show_problems	0 - Todos; 1 - *(predeterminado)* No suprimidos; 2 - Ninguno.
Agrupar por			
Atributo	0	group_by.0.attribute	0 - Grupo de hosts; 1 - Valor de etiqueta; 2 - Gravedad. Nota: El número del nombre de la propiedad hace referencia al orden del atributo en la lista de atributos de agrupación. Comportamiento del parámetro: - *obligatorio* si se configura *Agrupar por*
Valor	1	group_by.0.tag_name	Cualquier valor de cadena. Nota: El número del nombre de la propiedad hace referencia al conjunto de atributos de agrupación establecido en el parámetro *Atributo*. Comportamiento del parámetro: - *obligatorio* si se configura *Agrupar por* y *Atributo* está establecido en "Valor de etiqueta"
Límite de hosts	0	show_lines	Los valores posibles van de 1 a 9999. Predeterminado: 100. Este parámetro no es compatible si se configura el widget en un dashboard de template.
Referencia	1	reference	Cualquier valor de cadena compuesto por 5 caracteres (por ejemplo, `ABCDE` o `JBPNL`). Este valor debe ser único dentro del dashboard al que pertenece el widget. Comportamiento del parámetro: - *obligatorio*

Ejemplos

Los siguientes ejemplos tienen como objetivo describir únicamente la configuración de los objetos de campo del widget del panel para el widget *Navegador de equipos*. Para obtener más información sobre la configuración de un panel, consulte `dashboard.create`.

Configuración de un widget *Navegador de equipos*

Configure un widget *Navegador de equipos* que muestre los equipos agrupados por su grupo de equipos y, luego, por el valor de la etiqueta "city".

Petición:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "Mi panel",
        "display_period": "30",
        "auto_start": "1",
        "pages": [
            {
                "widgets": [
                    {
                        "type": "hostnavigator",
                        "name": "Navegador de equipos",
                        "x": "0",
                        "y": "0",
                        "width": "12",
                        "height": "5",
                        "view_mode": "0",
                        "fields": [
                            {
                                "type": 2,
                                "name": "groupids.0",
                                "value": 2
                            },
                            {
                                "type": 2,
                                "name": "groupids.1",
                                "value": 4
                            },
                            {
                                "type": 0,
                                "name": "group_by.0.attribute",
                                "value": 0
                            },
                            {
                                "type": 0,
                                "name": "group_by.1.attribute",
                                "value": 1
                            },
                            {
                                "type": 1,
                                "name": "group_by.1.tag_name",
                                "value": "city"
                            },
                            {
                                "type": 1,
                                "name": "reference",
                                "value": "SWKLB"
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
```

```
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Véase también

- Campo de widget de panel
- `dashboard.create`
- `dashboard.update`

15 Tarjeta de métrica

Descripción

Estos parámetros y los posibles valores de propiedad para los objetos de campo del widget del dashboard correspondiente permiten configurar el widget *Item card* en los métodos `dashboard.create` y `dashboard.update`.

Attention:
Las propiedades `fields` del widget no se validan durante la creación o actualización de un dashboard. Esto permite a los usuarios modificar widgets integrados y crear widgets personalizados, pero también introduce el riesgo de crear o actualizar widgets de forma incorrecta. Para garantizar la creación o actualización correcta del widget *Item card*, consulte el comportamiento de los parámetros descrito en las tablas siguientes.

Parámetros

Se admiten los siguientes parámetros para el widget *Tarjeta de elemento*.

Parámetro	tipo	nombre	valor
Intervalo de actualización	0	rf_rate	0 - Sin actualización; 10 - 10 segundos; 30 - 30 segundos; 60 - *(por defecto)* 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - 15 minutos.
Elemento	4	itemid.0	ID de Elemento. Comportamiento del parámetro: - *obligatorio* si *Elemento (Widget)* no está establecido

Parámetro	tipo	nombre	valor
Elemento (Widget)	1	itemid._reference	En lugar del ID de Elemento: `ABCDE._itemid` - establezca un widget compatible (con su parámetro *Referencia* configurado en "`ABCDE`") como fuente de datos para el elemento. Comportamiento del parámetro: - *obligatorio* si *Elemento* no está establecido
Mostrar	0	sections.0	0 - Descripción; 1 - Texto de error; 2 - Métricas; 3 - Últimos datos; 4 - Tipo de información; 5 - Disparadores; 6 - Interfaz del host; 7 - Tipo; 8 - Inventario del host; 9 - Etiquetas. Nota: El número en el nombre de la propiedad hace referencia al orden de la sección en la lista de secciones. Para configurar varias secciones, cree un objeto de campo de widget de panel para cada sección con un número incrementado en el nombre de la propiedad.
Sobrescribir host	1	override_hostid._reference	`ABCDE._hostid` - establezca un widget compatible (con su parámetro *Referencia* configurado en "`ABCDE`") como fuente de datos para los hosts; `DASHBOARD._hostid` - establezca el selector de host del panel como fuente de datos para los hosts. Por defecto: `""` (vacío) Este parámetro no es compatible si se configura el widget en un panel de plantilla.

Sparkline

Los siguientes parámetros son compatibles si *Mostrar* está establecido en "Últimos datos".

Parámetro	tipo	nombre	valor
Ancho	0	sparkline.width	Los valores posibles van de 0 a 10. Predeterminado: 1.
Relleno	0	sparkline.fill	Los valores posibles van de 0 a 10. Predeterminado: 3.
Color	1	sparkline.color	Código de color hexadecimal (por ejemplo, `FF0000`). Predeterminado: `42A5F5`.
Período de tiempo	1	sparkline.time_period.reference	`DASHBOARD._timeperiod`: establece el selector de período de tiempo del dashboard como fuente de datos; `ABCDE._timeperiod`: establece un widget compatible (con su parámetro `reference` igual a `ABCDE`) como fuente de datos. Predeterminado: `""` (vacío) Como alternativa, puede establecer el período de tiempo únicamente en los parámetros *Desde* y *Hasta*.

Parámetro	tipo	nombre	valor
Desde	1	sparkline.time_period.from	Cadena de tiempo válida en formato absoluto (`YYYY-MM-DD hh:mm:ss`) o en sintaxis de tiempo relativa (`now`, `now/d`, `now/w-1w`, etc.). Predeterminado: now-1h. Comportamiento del parámetro: - *compatible* si no se establece *Período de tiempo* - *obligatorio* si se establece `sparkline.time_period.to`
Hasta	1	sparkline.time_period.to	Cadena de tiempo válida en formato absoluto (`YYYY-MM-DD hh:mm:ss`) o en sintaxis de tiempo relativa (`now`, `now/d`, `now/w-1w`, etc.). Predeterminado: now. Comportamiento del parámetro: - *compatible* si no se establece *Período de tiempo* - *obligatorio* si se establece `time_period.to`
Datos históricos	0	sparkline.history	0 - *(predeterminado)* Automático; 1 - Historial; 2 - Tendencias.

Ejemplos

Los siguientes ejemplos tienen como objetivo describir únicamente la configuración de los objetos de campo del widget de panel para el widget *Tarjeta de métrica*. Para obtener más información sobre la configuración de un panel, consulte `dashboard.create`.

Configuración de un widget *Tarjeta de métrica*

Configure un widget *Tarjeta de métrica* que muestre estas secciones: ”Descripción”, ”Últimos datos”, ”Disparadores” y ”Etiquetas”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "Mi panel",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "itemcard",
                        "name": "Tarjeta de métrica",
                        "x": 0,
                        "y": 0,
                        "width": 14,
                        "height": 7,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 4,
                                "name": "itemid.0",
                                "value": 42257
                            },
                            {
                                "type": 0,
                                "name": "sections.0",
                                "value": 0
                            },
                            {
```

```
                                "type": 0,
                                "name": "sections.1",
                                "value": 3
                            },
                            {
                                "type": 0,
                                "name": "sections.2",
                                "value": 5
                            },
                            {
                                "type": 0,
                                "name": "sections.3",
                                "value": 9
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Véase también

- Campo de widget de panel
- `dashboard.create`
- `dashboard.update`

16 Historial de métricas

Descripción

Estos parámetros y los posibles valores de propiedad para los objetos de campo correspondientes del widget del dashboard permiten configurar el widget *Historial de item* en los métodos `dashboard.create` y `dashboard.update`.

Attention:
Las propiedades `fields` del widget no se validan durante la creación o actualización de un dashboard. Esto permite a los usuarios modificar widgets integrados y crear widgets personalizados, pero también introduce el riesgo de crear o actualizar widgets de forma incorrecta. Para garantizar la creación o actualización correctas del widget *Historial de item*, consulte el comportamiento de los parámetros descrito en las tablas siguientes.

Parámetros

Los siguientes parámetros son compatibles con el widget *Historial de item*.

Parámetro	tipo	nombre	valor
Intervalo de actualización	0	rf_rate	0 - Sin actualización; 10 - 10 segundos; 30 - 30 segundos; 60 - *(predeterminado)* 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - 15 minutos.
Diseño	0	layout	0 - *(predeterminado)* Horizontal; 1 - Vertical.
Columnas (consulte más abajo)			
Mostrar líneas	0	show_lines	Los valores posibles van de 1 a 100. Predeterminado: 25.
Anular host	1	override_hostid._reference	`ABCDE._hostid` - establece un widget compatible (con su parámetro *Referencia* establecido en "`ABCDE`") como fuente de datos para los hosts; `DASHBOARD._hostid` - establece el selector de host del dashboard como fuente de datos para los hosts. Este parámetro no es compatible al configurar el widget en un dashboard de template.
Configuración avanzada (consulte más abajo)			
Referencia	1	reference	Cualquier valor de cadena compuesto por 5 caracteres (por ejemplo, `ABCDE` o `JBPNL`). Este valor debe ser único dentro del dashboard al que pertenece el widget. Comportamiento del parámetro: - *obligatorio*

Columnas

Las columnas tienen parámetros comunes y parámetros adicionales dependiendo de la configuración del parámetro *Métrica*.

Note:
Para todos los parámetros relacionados con columnas, el número en el nombre de la propiedad (por ejemplo, columns.0.name) hace referencia a una columna para la cual se configura el parámetro.

Los siguientes parámetros son compatibles con todas las columnas.

Parámetro	tipo	nombre	valor
Nombre	1	columns.0.name	Cualquier valor de cadena. Comportamiento del parámetro: - *obligatorio*

Parámetro	tipo	nombre	valor
Métrica	4	columns.0.itemid	ID de la métrica. Al configurar el widget en un panel de plantilla, solo deben establecerse las métricas configuradas en la plantilla. Comportamiento del parámetro: - *obligatorio*
Color base	1	columns.0.base_color	Código de color hexadecimal (por ejemplo, FF0000). Por defecto: "" (vacío).

Se admiten los siguientes parámetros de columna si la *Métrica* configurada es una métrica de tipo numérico.

Parámetro	tipo	nombre	valor
Mostrar	0	columns.0.display	1 - *(por defecto)* Como está; 2 - Barra; 3 - Indicadores.
Mín	1	columns.0.min	Cualquier valor numérico. Comportamiento del parámetro: - *admitido* si *Mostrar* está configurado como ”Barra” o ”Indicadores”
Máx	1	columns.0.max	Cualquier valor numérico. Comportamiento del parámetro: - *admitido* si *Mostrar* está configurado como ”Barra” o ”Indicadores”
Umbrales			
Color	1	columns.0.thresholds.color	Código de color hexadecimal (por ejemplo, FF0000).
Umbral	1	columns.0.thresholds.threshold	Cualquier valor numérico. Se admiten Sufijos (por ejemplo, ”1d”, ”2w”, ”4K”, ”8G”).
Datos históricos	0	columns.0.history	0 - *(por defecto)* Automático; 1 - Historial; 2 - Tendencias.

Los siguientes parámetros de columna son compatibles si el *Item* configurado es un item de tipo carácter, texto o registro.

Parameter	type	name	value
Highlights			
Highlight	1	columns.0.highlights.color	Código de color hexadecimal (por ejemplo, FF0000).
Threshold	1	columns.0.highlights.pattern	Cualquier expresión regular.
Display	0	columns.0.display	1 - *(predeterminado)* Tal cual; 4 - HTML; 5 - Una sola línea.
Single line	0	columns.0.max_length	Los valores posibles van de 1 a 500. Predeterminado: 100. Comportamiento del parámetro: - *compatible* si *Display* está establecido en ”Una sola línea”
Use monospace font	0	columns.0.monospace_font	0 - *(predeterminado)* Usar la fuente predeterminada; 1 - Usar fuente monoespaciada.
Display local time	0	columns.0.local_time	0 - *(predeterminado)* Mostrar la marca de tiempo; 1 - Mostrar la hora local. Comportamiento del parámetro: - *compatible* si *Item* está establecido en un item de tipo registro y *Show timestamp* está establecido en ”Habilitado”

Se admiten los siguientes parámetros de columna si la *Métrica* configurada es de tipo binario.

Parámetro	tipo	nombre	valor
Mostrar miniatura	1	columns.0.show_thumbnail	0 - *(por defecto)* Deshabilitado; 1 - Habilitado.

Configuración avanzada

Los siguientes parámetros de configuración avanzada son compatibles con el widget *Historial de item*.

Parámetro	tipo	nombre	valor
Nuevos valores	0	sortorder	0 - *(por defecto)* Arriba; 1 - Abajo.
Mostrar marca de tiempo	0	show_timestamp	0 - *(por defecto)* Deshabilitado; 1 - Habilitado.
Mostrar encabezado de columna	0	show_column_header	0 - Apagado; 1 - Horizontal; 2 - *(por defecto)* Vertical.
Período de tiempo	1	time_period._reference	`DASHBOARD._timeperiod` - establece el selector de período de tiempo del dashboard como fuente de datos; `ABCDE._timeperiod` - establece un widget compatible (con su parámetro *Referencia* configurado en "`ABCDE`") como fuente de datos. Por defecto: `DASHBOARD._timeperiod` Alternativamente, puede establecer el período de tiempo solo en los parámetros *Desde* y *Hasta*.
Desde	1	time_period.from	Cadena de tiempo válida en sintaxis absoluta (`YYYY-MM-DD hh:mm:ss`) o relativa (`now`, `now/d`, `now/w-1w`, etc.). Comportamiento del parámetro: - *soportado* si *Período de tiempo* no está configurado - *requerido* si `time_period.to` está configurado
Hasta	1	time_period.to	Cadena de tiempo válida en sintaxis absoluta (`YYYY-MM-DD hh:mm:ss`) o relativa (`now`, `now/d`, `now/w-1w`, etc.). Comportamiento del parámetro: - *soportado* si *Período de tiempo* no está configurado - *requerido* si `time_period.from` está configurado

Ejemplos

Los siguientes ejemplos tienen como objetivo describir únicamente la configuración de los objetos de campo del widget de panel para el widget *Histórico de métricas*. Para obtener más información sobre la configuración de un panel, consulte `dashboard.create`.

Configuración de un widget *Historial de métricas*

Configure un widget *Historial de métricas* que muestre los últimos datos de dos métricas numéricas "42269" y "42270". Además, configure las columnas de métricas para que se muestren verticalmente, con los nombres de las columnas mostrados horizontalmente; limite la visualización a 15 líneas de datos e incluya una columna de marca de tiempo por separado.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "Mi panel",
```

```
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "itemhistory",
                        "name": "Historial de métricas",
                        "x": "0",
                        "y": "0",
                        "width": "18",
                        "height": "6",
                        "view_mode": "0",
                        "fields": [
                            {
                                "type": "0",
                                "name": "layout",
                                "value": "1"
                            },
                            {
                                "type": "1",
                                "name": "columns.0.name",
                                "value": "Utilización de CPU"
                            },
                            {
                                "type": "4",
                                "name": "columns.0.itemid",
                                "value": "42269"
                            },
                            {
                                "type": "1",
                                "name": "columns.1.name",
                                "value": "Utilización de memoria"
                            },
                            {
                                "type": "4",
                                "name": "columns.1.itemid",
                                "value": "42270"
                            },
                            {
                                "type": "0",
                                "name": "show_lines",
                                "value": "15"
                            },
                            {
                                "type": "0",
                                "name": "show_timestamp",
                                "value": "1"
                            },
                            {
                                "type": "0",
                                "name": "show_column_header",
                                "value": "1"
                            },
                            {
                                "type": "1",
                                "name": "reference",
                                "value": "KIVKD"
                            }
                        ]
                    }
                ]
```

```
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Véase también

- Campo de widget de panel
- `dashboard.create`
- `dashboard.update`

17 Navegador de métricas

Descripción

Estos parámetros y los posibles valores de propiedad para los objetos de campo de widget correspondientes permiten configurar el widget *Navegador de items* en los métodos `dashboard.create` y `dashboard.update`.

Attention:
Las propiedades `fields` del widget no se validan durante la creación o actualización de un dashboard. Esto permite a los usuarios modificar widgets integrados y crear widgets personalizados, pero también introduce el riesgo de crear o actualizar widgets de forma incorrecta. Para garantizar la creación o actualización correctas del widget *Navegador de items*, consulte el comportamiento de los parámetros descrito en las tablas siguientes.

Parámetros

Los siguientes parámetros son compatibles con el widget *Navegador de items*.

Parámetro	tipo	nombre	valor
Intervalo de actualización	0	rf_rate	0 - Sin actualización; 10 - 10 segundos; 30 - 30 segundos; 60 - *(predeterminado)* 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - 15 minutos.

Parámetro	tipo	nombre	valor
Grupos de hosts	2	groupids.0	ID del grupo de hosts. Nota: Para configurar varios grupos de hosts, cree un objeto de campo del widget del dashboard para cada grupo de hosts, con un número incrementado en el nombre de la propiedad. Este parámetro no es compatible si se configura el widget en un dashboard de template.
Grupos de hosts (widget)	1	groupids._reference	En lugar del ID del grupo de hosts: `ABCDE._hostgroupids` - establezca un widget compatible (con su parámetro *Referencia* establecido en "`ABCDE`") como fuente de datos para los grupos de hosts. Este parámetro no es compatible si se configura el widget en un dashboard de template.
Hosts	3	hostids.0	ID del host. Nota: Para configurar varios hosts, cree un objeto de campo del widget del dashboard para cada host, con un número incrementado en el nombre de la propiedad. Para varios hosts, el parámetro *Grupos de hosts* debe no estar configurado o estar configurado con al menos un grupo de hosts al que pertenezcan los hosts configurados. Este parámetro no es compatible si se configura el widget en un dashboard de template.
Hosts (widget/dashboard)	1	hostids._reference	En lugar del ID del host: `DASHBOARD.hostid` - establezca el selector de hosts del dashboard como fuente de datos para los hosts; `ABCDE._hostid` - establezca un widget compatible (con su parámetro *Referencia* establecido en "`ABCDE`") como fuente de datos para los hosts. Este parámetro no es compatible si se configura el widget en un dashboard de template.
Etiquetas de hosts			
Tipo de evaluación	0	host_tags_evaltype	0 - *(predeterminado)* Y/O; 2 - O. Este parámetro no es compatible si se configura el widget en un dashboard de template.
Nombre de etiqueta	1	host_tags.0.tag	Cualquier valor de cadena. Nota: El número del nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* si se configuran las *Etiquetas de hosts* Este parámetro no es compatible si se configura el widget en un dashboard de template.

Parámetro	tipo	nombre	valor
Operador	0	host_tags.0.operator	0 - Contiene; 1 - Es igual a; 2 - No contiene; 3 - No es igual a; 4 - Existe; 5 - No existe. Nota: El número del nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* si se configuran las *Etiquetas de hosts* Este parámetro no es compatible si se configura el widget en un dashboard de template.
Valor de etiqueta	1	host_tags.0.value	Cualquier valor de cadena. Nota: El número del nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* si se configuran las *Etiquetas de hosts* Este parámetro no es compatible si se configura el widget en un dashboard de template.
Patrones de items	1	items.0	Nombre o patrón del item. Nota: Para configurar varios patrones de items, cree un objeto de campo del widget del dashboard para cada patrón de item, con un número incrementado en el nombre de la propiedad.
Etiquetas de items			
Tipo de evaluación	0	item_tags_evaltype	0 - *(predeterminado)* Y/O; 2 - O.
Nombre de etiqueta	1	item_tags.0.tag	Cualquier valor de cadena. Nota: El número del nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* si se configuran las *Etiquetas de items*
Operador	0	item_tags.0.operator	0 - Contiene; 1 - Es igual a; 2 - No contiene; 3 - No es igual a; 4 - Existe; 5 - No existe. Nota: El número del nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* si se configuran las *Etiquetas de items*
Valor de etiqueta	1	item_tags.0.value	Cualquier valor de cadena. Nota: El número del nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* si se configuran las *Etiquetas de items*

Parámetro	tipo	nombre	valor
Estado	0	state	-1 - *(predeterminado)* Todos; 0 - Normal; 1 - No compatible.
Mostrar problemas	0	show_problems	0 - Todos; 1 - *(predeterminado)* No suprimidos; 2 - Ninguno.
Agrupar por			
Atributo	0	group_by.0.attribute	0 - Grupo de hosts; 1 - Nombre del host; 2 - Valor de la etiqueta del host; 3 - Valor de la etiqueta del item. Nota: El número del nombre de la propiedad hace referencia al orden del atributo en la lista de atributos de agrupación. Comportamiento del parámetro: - *obligatorio* si se configura *Agrupar por*
Valor	1	group_by.0.tag_name	Cualquier valor de cadena. Nota: El número del nombre de la propiedad hace referencia al conjunto de atributos de agrupación establecido en el parámetro *Atributo*. Comportamiento del parámetro: - *obligatorio* si se configura *Agrupar por* y *Atributo* se establece en ”Valor de la etiqueta del host” o ”Valor de la etiqueta del item”
Límite de items	0	show_lines	Los valores posibles van de 1 a 9999. Predeterminado: 100.
Referencia	1	reference	Cualquier valor de cadena compuesto por 5 caracteres (por ejemplo, `ABCDE` o `JBPNL`). Este valor debe ser único dentro del dashboard al que pertenece el widget. Comportamiento del parámetro: - *obligatorio*

Ejemplos

Los siguientes ejemplos tienen como objetivo describir únicamente la configuración de los objetos de campo del widget de panel para el widget *Navegador de métricas*. Para obtener más información sobre la configuración de un panel, consulte `dashboard.create`.

Configuración de un widget *Navegador de métricas*

Configure un widget *Navegador de métricas* que muestre hasta 1000 métricas agrupadas por su equipo y, luego, por el valor de la etiqueta de métrica ”component”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": "30",
        "auto_start": "1",
        "pages": [
            {
                "widgets": [
                    {
                        "type": "itemnavigator",
```

```
                        "name": "Item navigator",
                        "x": "0",
                        "y": "0",
                        "width": "12",
                        "height": "5",
                        "view_mode": "0",
                        "fields": [
                            {
                                "type": 0,
                                "name": "group_by.0.attribute",
                                "value": 0
                            },
                            {
                                "type": 0,
                                "name": "group_by.1.attribute",
                                "value": 3
                            },
                            {
                                "type": 1,
                                "name": "group_by.1.tag_name",
                                "value": "component"
                            },
                            {
                                "type": 0,
                                "name": "show_lines",
                                "value": 1000
                            },
                            {
                                "type": 1,
                                "name": "reference",
                                "value": "DFNLK"
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
```

```
}
```

Véase también

- Campo de widget de panel
- `dashboard.create`
- `dashboard.update`

18 Valor de la métrica

Descripción

Estos parámetros y los posibles valores de propiedad para los objetos de campo correspondientes del widget del dashboard permiten configurar el widget *Valor del item* en los métodos `dashboard.create` y `dashboard.update`.

Attention:
Las propiedades `fields` del widget no se validan durante la creación o actualización de un dashboard. Esto permite a los usuarios modificar widgets integrados y crear widgets personalizados, pero también introduce el riesgo de crear o actualizar widgets de forma incorrecta. Para garantizar la creación o actualización correcta del widget *Valor del item*, consulte el comportamiento de los parámetros descrito en las tablas siguientes.

Parámetros

Se admiten los siguientes parámetros para el widget *Valor del elemento*.

Parámetro	tipo	nombre	valor
Intervalo de actualización	0	rf_rate	0 - Sin actualización; 10 - 10 segundos; 30 - 30 segundos; 60 - *(por defecto)* 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - 15 minutos.
Elemento	4	itemid.0	ID del Elemento. Comportamiento del parámetro: - *obligatorio* si *Elemento (Widget)* no está establecido
Elemento (Widget)	1	itemid._reference	En lugar del ID del Elemento: `ABCDE._itemid` - establezca un widget compatible (con su parámetro *Referencia* configurado en "ABCDE") como fuente de datos para los elementos.
Mostrar	0	show.0	Comportamiento del parámetro: - *obligatorio* si *Elemento* no está establecido 1 - Descripción; 2 - Valor; 3 - Hora; 4 - Indicador de cambio; 5 - Minigráfico. Por defecto: 1, 2, 3, 4. Nota: Para configurar varios valores, cree un objeto de campo de widget de panel para cada valor con un número incrementado en el nombre de la propiedad.

Parámetro	tipo	nombre	valor
Anular equipo	1	override_hostid._reference	`ABCDE._hostid` - establezca un widget compatible (con su parámetro *Referencia* configurado en "`ABCDE`") como fuente de datos para los equipos; `DASHBOARD._hostid` - establezca el selector de equipo del panel como fuente de datos para los equipos. Este parámetro no es compatible si se configura el widget en un panel de plantilla.

Configuración avanzada

Los siguientes parámetros de configuración avanzada son compatibles con el widget *Valor de item*.

Note:
El número en el nombre de la propiedad *Umbrales* (por ejemplo, thresholds.0.color) hace referencia a la posición del umbral en una lista, ordenada de forma ascendente. Sin embargo, si los umbrales se configuran en un orden diferente, los valores se ordenarán de forma ascendente después de actualizar la configuración del widget en el frontend de Zabbix (por ejemplo, `"thresholds.0.threshold":"5"` → `"thresholds.0.threshold":"1"`; `"thresholds.1.threshold":"1"` → `"thresholds.1.threshold": "5"`).

Parámetro	tipo	nombre	valor
Color de fondo	1	bg_color	Código de color hexadecimal (por ejemplo, `FF0000`). Por defecto: `""` (vacío).
Umbrales			
Color	1	thresholds.0.color	Código de color hexadecimal (por ejemplo, `FF0000`).
Umbral	1	thresholds.0.threshold	Cualquier valor de cadena.
Función de agregación	0	aggregate_function	0 - *(por defecto)* no se usa; 1 - min; 2 - max; 3 - avg; 4 - count; 5 - sum; 6 - first; 7 - last.
Periodo de tiempo	1	time_period._reference	`DASHBOARD._timeperiod` - establece el selector de periodo de tiempo del dashboard como fuente de datos; `ABCDE._timeperiod` - establece un widget compatible (con su parámetro `reference` igual a `ABCDE`) como fuente de datos. Por defecto: `DASHBOARD._timeperiod` Alternativamente, puede establecer el periodo de tiempo solo en los parámetros *Desde* y *Hasta*. Comportamiento del parámetro: - *soportado* si *Función de agregación* está configurada como "min", "max", "avg", "count", "sum", "first", "last"
Desde	1	time_period.from	Cadena de tiempo válida en sintaxis absoluta (`YYYY-MM-DD hh:mm:ss`) o relativa (`now`, `now/d`, `now/w-1w`, etc.). Comportamiento del parámetro: - *soportado* si *Periodo de tiempo* no está configurado y *Función de agregación* está configurada como "min", "max", "avg", "count", "sum", "first", "last" - *requerido* si se establece `time_period.to`

Parámetro	tipo	nombre	valor
Hasta	1	time_period.to	Cadena de tiempo válida en sintaxis absoluta (`YYYY-MM-DD hh:mm:ss`) o relativa (`now`, `now/d`, `now/w-1w`, etc.). Comportamiento del parámetro: - *soportado* si *Periodo de tiempo* no está configurado y *Función de agregación* está configurada como ”min”, ”max”, ”avg”, ”count”, ”sum”, ”first”, ”last” - *requerido* si se establece `time_period.from`
Datos históricos	0	history	0 - *(por defecto)* Auto; 1 - History; 2 - Trends.

Descripción

Los siguientes parámetros de configuración avanzados son compatibles si *Mostrar* está establecido en ”Descripción”.

Parámetro	type	name	value
Descripción	1	description	Cualquier valor de cadena, incluidas las macros. Macros compatibles: `{HOST.*}`, `{ITEM.*}`, `{INVENTORY.*}`, macros de usuario. Predeterminado: `{ITEM.NAME}`.
Posición horizontal	0	desc_h_pos	0 - Izquierda; 1 - *(predeterminado)* Centro; 2 - Derecha. Dos o más elementos (Descripción, Valor, Tiempo) no pueden compartir la misma *Posición horizontal* y *Posición vertical*.
Posición vertical	0	desc_v_pos	0 - Arriba; 1 - Centro; 2 - *(predeterminado)* Abajo. Dos o más elementos (Descripción, Valor, Tiempo) no pueden compartir la misma *Posición horizontal* y *Posición vertical*.
Tamaño	0	desc_size	Los valores posibles van de 1 a 100. Predeterminado: 15.
Negrita	0	desc_bold	0 - *(predeterminado)* Deshabilitado; 1 - Habilitado.
Color	1	desc_color	Código de color hexadecimal (por ejemplo, `FF0000`). Predeterminado: `""` (vacío).

Valor

Los siguientes parámetros de configuración avanzados son compatibles si *Mostrar* está establecido en ”Valor”.

Parámetro	tipo	nombre	valor
Posiciones decimales			
Posiciones decimales	0	decimal_places	Los valores posibles van de 1 a 10. Predeterminado: 2.
Tamaño	0	decimal_size	Los valores posibles van de 1 a 100. Predeterminado: 35.
Posición			

Parámetro	tipo	nombre	valor
Posición horizontal	0	value_h_pos	0 - Izquierda; 1 - *(predeterminado)* Centro; 2 - Derecha. Dos o más elementos (Descripción, Valor, Hora) no pueden compartir la misma *Posición horizontal* y *Posición vertical*.
Posición vertical	0	value_v_pos	0 - Arriba; 1 - *(predeterminado)* Centro; 2 - Abajo. Dos o más elementos (Descripción, Valor, Hora) no pueden compartir la misma *Posición horizontal* y *Posición vertical*.
Tamaño	0	value_size	Los valores posibles van de 1 a 100. Predeterminado: 45.
Negrita	0	value_bold	0 - Deshabilitado; 1 - *(predeterminado)* Habilitado.
Color	1	value_color	Código de color hexadecimal (por ejemplo, FF0000). Predeterminado: "" (vacío).
Unidades			
Unidades (casilla de verificación)	0	units_show	0 - Deshabilitado; 1 - *(predeterminado)* Habilitado.
Unidades (valor)	1	units	Cualquier valor de cadena.
Posición	0	units_pos	0 - Antes del valor; 1 - Encima del valor; 2 - *(predeterminado)* Después del valor; 3 - Debajo del valor.
Tamaño	0	units_size	Los valores posibles van de 1 a 100. Predeterminado: 35.
Negrita	0	units_bold	0 - Deshabilitado; 1 - *(predeterminado)* Habilitado.
Color	1	units_color	Código de color hexadecimal (por ejemplo, FF0000). Predeterminado: "" (vacío).

Hora

Los siguientes parámetros de configuración avanzados son compatibles si *Mostrar* está establecido en ”Hora”.

Parámetro	tipo	nombre	valor
Posición horizontal	0	time_h_pos	0 - Izquierda; 1 - *(predeterminado)* Centro; 2 - Derecha. Dos o más elementos (Descripción, Valor, Hora) no pueden compartir la misma *Posición horizontal* y *Posición vertical*.
Posición vertical	0	time_v_pos	0 - *(predeterminado)* Arriba; 1 - Centro; 2 - Abajo. Dos o más elementos (Descripción, Valor, Hora) no pueden compartir la misma *Posición horizontal* y *Posición vertical*.
Tamaño	0	time_size	Los valores posibles van de 1 a 100. Predeterminado: 15.
Negrita	0	time_bold	0 - *(predeterminado)* Deshabilitado; 1 - Habilitado.

Parámetro	tipo	nombre	valor
Color	1	time_color	Código de color hexadecimal (por ejemplo, FF0000). Predeterminado: "" (vacío).

Indicador de cambio

Se admiten los siguientes parámetros de configuración avanzada si *Mostrar* está establecido en "Indicador de cambio".

Parámetro	tipo	nombre	valor
Color del indicador de cambio ↑	1	up_color	Código de color hexadecimal (por ejemplo, FF0000). Por defecto: "" (vacío).
Color del indicador de cambio ↓	1	down_color	Código de color hexadecimal (por ejemplo, FF0000). Por defecto: "" (vacío).
Color del indicador de cambio ↕	1	updown_color	Código de color hexadecimal (por ejemplo, FF0000). Por defecto: "" (vacío).

Sparkline

Los siguientes parámetros de configuración avanzados son compatibles si *Mostrar* está establecido en "Sparkline".

Parámetro	tipo	nombre	valor
Ancho	0	sparkline.width	Los valores posibles van de 0 a 10. Predeterminado: 1.
Relleno	0	sparkline.fill	Los valores posibles van de 0 a 10. Predeterminado: 3.
Color	1	sparkline.color	Código de color hexadecimal (por ejemplo, FF0000). Predeterminado: 42A5F5.
Período de tiempo	1	sparkline.time_period.from sparkline.time_period.to	DASHBOARD._timeperiod: establece el selector de período de tiempo del dashboard como fuente de datos; ABCDE._timeperiod: establece un widget compatible (con su parámetro reference igual a ABCDE) como fuente de datos. Predeterminado: DASHBOARD._timeperiod Como alternativa, puede establecer el período de tiempo únicamente en los parámetros *Desde* y *Hasta*.
Desde	1	sparkline.time_period.from	Cadena de tiempo válida en sintaxis absoluta (YYYY-MM-DD hh:mm:ss) o relativa (now, now/d, now/w-1w, etc.). Comportamiento del parámetro: - *obligatorio* si se establece sparkline.time_period.to
Hasta	1	sparkline.time_period.to	Cadena de tiempo válida en sintaxis absoluta (YYYY-MM-DD hh:mm:ss) o relativa (now, now/d, now/w-1w, etc.). Comportamiento del parámetro: - *obligatorio* si se establece sparkline.time_period.from
Datos históricos	0	sparkline.history	0 - *(predeterminado)* Automático; 1 - Historial; 2 - Tendencias.

Ejemplos

Los siguientes ejemplos tienen como objetivo describir únicamente la configuración de los objetos de campo del widget de panel para el widget *Valor de métrica*. Para obtener más información sobre la configuración de un panel, consulte dashboard.create.

Configuración de un widget *Valor de métrica*

Configure un widget *Valor de métrica* que muestre el valor de la métrica para la métrica "42266" (disponibilidad del agente Zabbix). Además, ajuste visualmente el widget con múltiples opciones avanzadas, incluyendo un color de fondo dinámico que cambia según el estado de disponibilidad del agente Zabbix.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "Mi panel",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "item",
                        "name": "Valor de métrica",
                        "x": 0,
                        "y": 0,
                        "width": 12,
                        "height": 3,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 4,
                                "name": "itemid.0",
                                "value": 42266
                            },
                            {
                                "type": 0,
                                "name": "show.0",
                                "value": 1
                            },
                            {
                                "type": 0,
                                "name": "show.1",
                                "value": 2
                            },
                            {
                                "type": 0,
                                "name": "show.2",
                                "value": 3
                            },
                            {
                                "type": 1,
                                "name": "description",
                                "value": "Estado del agente"
                            },
                            {
                                "type": 0,
                                "name": "desc_h_pos",
                                "value": 0
                            },
                            {
                                "type": 0,
                                "name": "desc_v_pos",
                                "value": 0
                            },
                            {
                                "type": 0,
```

```
            "name": "desc_bold",
            "value": 1
        },
        {
            "type": 1,
            "name": "desc_color",
            "value": "F06291"
        },
        {
            "type": 0,
            "name": "value_h_pos",
            "value": 0
        },
        {
            "type": 0,
            "name": "value_size",
            "value": 25
        },
        {
            "type": 1,
            "name": "value_color",
            "value": "FFFF00"
        },
        {
            "type": 0,
            "name": "units_show",
            "value": 0
        },
        {
            "type": 0,
            "name": "time_h_pos",
            "value": 2
        },
        {
            "type": 0,
            "name": "time_v_pos",
            "value": 2
        },
        {
            "type": 0,
            "name": "time_size",
            "value": 10
        },
        {
            "type": 0,
            "name": "time_bold",
            "value": 1
        },
        {
            "type": 1,
            "name": "time_color",
            "value": "9FA8DA"
        },
        {
            "type": 1,
            "name": "thresholds.0.color",
            "value": "E1E1E1"
        },
        {
            "type": 1,
            "name": "thresholds.0.threshold",
            "value": "0"
```

```
                            },
                            {
                                "type": 1,
                                "name": "thresholds.1.color",
                                "value": "D1C4E9"
                            },
                            {
                                "type": 1,
                                "name": "thresholds.1.threshold",
                                "value": "1"
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Véase también

- Campo de widget de panel
- `dashboard.create`
- `dashboard.update`

19 Mapa

Descripción

Estos parámetros y los posibles valores de propiedad para los objetos de campo del widget del dashboard correspondiente permiten configurar el widget *Map* en los métodos `dashboard.create` y `dashboard.update`.

Attention:
Las propiedades `fields` del widget no se validan durante la creación o actualización de un dashboard. Esto permite a los usuarios modificar widgets integrados y crear widgets personalizados, pero también introduce el riesgo de crear o actualizar widgets de forma incorrecta. Para garantizar la creación o actualización correcta del widget *Map*, consulte el comportamiento de los parámetros descrito en las tablas siguientes.

Parámetros

Los siguientes parámetros son compatibles para el widget *Mapa*.

Parámetro		tipo	nombre	valor
Intervalo de actualización		0	rf_rate	0 - Sin actualización; 10 - 10 segundos; 30 - 30 segundos; 60 - 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - *(por defecto)* 15 minutos.
Mapa		8	sysmapid.0	ID de Mapa. Comportamiento del parámetro: - *requerido* si *Mapa (Widget)* no está establecido
	Mapa (Widget)	1	sysmapid._reference	`ABCDE._mapid` - establece un widget *Árbol de navegación de mapas* (con su parámetro *Referencia* configurado en "`ABCDE`") como la fuente de datos para los mapas. Comportamiento del parámetro: - *requerido* si *Mapa* no está establecido
Referencia		1	reference	Cualquier valor de cadena que consista en 5 caracteres (por ejemplo, `ABCDE` o `JBPNL`). Este valor debe ser único dentro del panel al que pertenece el widget. Comportamiento del parámetro: - *requerido*

Ejemplos

Los siguientes ejemplos tienen como objetivo describir únicamente la configuración de los objetos de campo del widget de panel para el widget *Mapa*. Para obtener más información sobre la configuración de un panel, consulte `dashboard.create`.

Configuración de un widget *Mapa*

Configure un widget *Mapa* que muestre el mapa "1".

Petición:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "map",
                        "name": "Map",
                        "x": 0,
                        "y": 0,
                        "width": 54,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 8,
                                "name": "sysmapid.0",
                                "value": 1
                            }
                        ]
```

```
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Configuración de un widget *Mapa* vinculado

Configure un widget *Mapa* que esté vinculado a un widget *Árbol de navegación de mapas*.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "Mi panel",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "map",
                        "name": "Mapa",
                        "x": 0,
                        "y": 5,
                        "width": 54,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 1,
                                "name": "sysmapid._reference",
                                "value": "ABCDE._mapid"
                            }
                        ]
                    },
                    {
```

```
                "type": "navtree",
                "name": "Árbol de navegación de mapas",
                "x": 0,
                "y": 0,
                "width": 18,
                "height": 5,
                "view_mode": 0,
                "fields": [
                    {
                        "type": 1,
                        "name": "navtree.1.name",
                        "value": "Elemento A"
                    },
                    {
                        "type": 1,
                        "name": "navtree.2.name",
                        "value": "Elemento B"
                    },
                    {
                        "type": 1,
                        "name": "navtree.3.name",
                        "value": "Elemento C"
                    },
                    {
                        "type": 1,
                        "name": "navtree.4.name",
                        "value": "Elemento A1"
                    },
                    {
                        "type": 1,
                        "name": "navtree.5.name",
                        "value": "Elemento A2"
                    },
                    {
                        "type": 1,
                        "name": "navtree.6.name",
                        "value": "Elemento B1"
                    },
                    {
                        "type": 1,
                        "name": "navtree.7.name",
                        "value": "Elemento B2"
                    },
                    {
                        "type": 0,
                        "name": "navtree.4.parent",
                        "value": 1
                    },
                    {
                        "type": 0,
                        "name": "navtree.5.parent",
                        "value": 1
                    },
                    {
                        "type": 0,
                        "name": "navtree.6.parent",
                        "value": 2
                    },
                    {
                        "type": 0,
                        "name": "navtree.7.parent",
                        "value": 2
```

```
                        },
                        {
                            "type": 0,
                            "name": "navtree.1.order",
                            "value": 1
                        },
                        {
                            "type": 0,
                            "name": "navtree.2.order",
                            "value": 2
                        },
                        {
                            "type": 0,
                            "name": "navtree.3.order",
                            "value": 3
                        },
                        {
                            "type": 0,
                            "name": "navtree.4.order",
                            "value": 1
                        },
                        {
                            "type": 0,
                            "name": "navtree.5.order",
                            "value": 2
                        },
                        {
                            "type": 0,
                            "name": "navtree.6.order",
                            "value": 1
                        },
                        {
                            "type": 0,
                            "name": "navtree.7.order",
                            "value": 2
                        },
                        {
                            "type": 8,
                            "name": "navtree.6.sysmapid",
                            "value": 1
                        },
                        {
                            "type": 1,
                            "name": "reference",
                            "value": "ABCDE"
                        }
                    ]
                }
            ]
        }
    ],
    "userGroups": [
        {
            "usrgrpid": 7,
            "permission": 2
        }
    ],
    "users": [
        {
            "userid": 1,
            "permission": 3
        }
```

```
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Véase también

- Campo de widget de panel
- `dashboard.create`
- `dashboard.update`
- Árbol de navegación de mapas

20 Árbol de navegación de mapas

Descripción

Estos parámetros y los posibles valores de propiedad para los objetos de campo del widget del dashboard correspondiente permiten configurar el widget *Árbol de navegación de mapas* en los métodos `dashboard.create` y `dashboard.update`.

Attention:
Las propiedades `fields` del widget no se validan durante la creación o actualización de un dashboard. Esto permite a los usuarios modificar widgets integrados y crear widgets personalizados, pero también introduce el riesgo de crear o actualizar widgets de forma incorrecta. Para garantizar la creación o actualización correctas del widget *Árbol de navegación de mapas*, consulte el comportamiento de los parámetros descrito en las tablas siguientes.

Parámetros

Los siguientes parámetros son compatibles para el widget *Árbol de navegación de mapas*.

Parámetro	tipo	nombre	valor
Intervalo de actualización	0	rf_rate	0 - Sin actualización; 10 - 10 segundos; 30 - 30 segundos; 60 - 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - *(por defecto)* 15 minutos.
Mostrar mapas no disponibles	1	show_unavailable	0 - *(por defecto)* Deshabilitado; 1 - Habilitado.
Referencia	1	reference	Cualquier valor de cadena que consista en 5 caracteres (por ejemplo, `ABCDE` o `JBPNL`). Este valor debe ser único dentro del panel al que pertenece el widget. Comportamiento del parámetro: - *requerido*

Los siguientes parámetros son compatibles para configurar los elementos del árbol de navegación de mapas.

Parámetro	tipo	nombre	valor
Nombre	1	navtree.1.name	Cualquier valor de cadena. Nota: El número en el nombre de la propiedad establece el número de elemento.
Mapa vinculado	8	navtree.1.sysmapid	ID del Mapa. Nota: El número en el nombre de la propiedad hace referencia al elemento al que se vincula el mapa.
Parámetros para crear la jerarquía de elementos	0	navtree.1.parent	Número de elemento padre. Nota: El número en el nombre de la propiedad hace referencia al elemento hijo. El valor de la propiedad hace referencia al elemento padre.
	0	navtree.1.order	Posición del elemento en el árbol de navegación de mapas. Nota: El número en el nombre de la propiedad hace referencia al número de elemento. El valor de la propiedad hace referencia a la posición del elemento en el árbol de navegación de mapas. La posición del elemento padre se determina dentro de todo el árbol de navegación de mapas. La posición del elemento hijo se determina dentro del elemento padre.

Ejemplos

Los siguientes ejemplos tienen como objetivo describir únicamente la configuración de los objetos de campo del widget de panel para el widget *Árbol de navegación de mapas*. Para obtener más información sobre la configuración de un panel, consulte `dashboard.create`.

Configuración de un widget *Árbol de navegación de mapas*

Configure un widget *Árbol de navegación de mapas* que muestre el siguiente árbol de navegación de mapas:

- Elemento A
  - Elemento A1
  - Elemento A2
- Elemento B
  - Elemento B1 (contiene el mapa vinculado "1" que puede mostrarse en un *widget de mapa* vinculado)
  - Elemento B2
- Elemento C

Petición:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "Mi panel",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "navtree",
                        "name": "Árbol de navegación de mapas",
                        "x": 0,
                        "y": 0,
                        "width": 18,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 1,
                                "name": "navtree.1.name",
                                "value": "Elemento A"
```

```
    },
    {
        "type": 1,
        "name": "navtree.2.name",
        "value": "Elemento B"
    },
    {
        "type": 1,
        "name": "navtree.3.name",
        "value": "Elemento C"
    },
    {
        "type": 1,
        "name": "navtree.4.name",
        "value": "Elemento A1"
    },
    {
        "type": 1,
        "name": "navtree.5.name",
        "value": "Elemento A2"
    },
    {
        "type": 1,
        "name": "navtree.6.name",
        "value": "Elemento B1"
    },
    {
        "type": 1,
        "name": "navtree.7.name",
        "value": "Elemento B2"
    },
    {
        "type": 0,
        "name": "navtree.4.parent",
        "value": 1
    },
    {
        "type": 0,
        "name": "navtree.5.parent",
        "value": 1
    },
    {
        "type": 0,
        "name": "navtree.6.parent",
        "value": 2
    },
    {
        "type": 0,
        "name": "navtree.7.parent",
        "value": 2
    },
    {
        "type": 0,
        "name": "navtree.1.order",
        "value": 1
    },
    {
        "type": 0,
        "name": "navtree.2.order",
        "value": 2
    },
    {
```

```
                                "type": 0,
                                "name": "navtree.3.order",
                                "value": 3
                            },
                            {
                                "type": 0,
                                "name": "navtree.4.order",
                                "value": 1
                            },
                            {
                                "type": 0,
                                "name": "navtree.5.order",
                                "value": 2
                            },
                            {
                                "type": 0,
                                "name": "navtree.6.order",
                                "value": 1
                            },
                            {
                                "type": 0,
                                "name": "navtree.7.order",
                                "value": 2
                            },
                            {
                                "type": 8,
                                "name": "navtree.6.sysmapid",
                                "value": 1
                            },
                            {
                                "type": 1,
                                "name": "reference",
                                "value": "HJQXF"
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```


Respuesta:


```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
```

```
    },
    "id": 1
}
```

Véase también

- Campo de widget de panel
- `dashboard.create`
- `dashboard.update`
- Mapa

21 Gráfico circular

Descripción

Estos parámetros y los posibles valores de propiedad para los objetos de campo del widget del dashboard correspondiente permiten configurar el widget *Gráfico circular* en los métodos `dashboard.create` y `dashboard.update`.

Attention:
Las propiedades `fields` del widget no se validan durante la creación o actualización de un dashboard. Esto permite a los usuarios modificar widgets integrados y crear widgets personalizados, pero también introduce el riesgo de crear o actualizar widgets de forma incorrecta. Para garantizar la creación o actualización correctas del widget *Gráfico circular*, consulte el comportamiento de los parámetros descrito en las tablas siguientes.

Parámetros

Los siguientes parámetros son compatibles para el widget *Gráfico circular*.

Parámetro	tipo	nombre	valor
Intervalo de actualización	0	rf_rate	0 - Sin actualización; 10 - 10 segundos; 30 - 30 segundos; 60 - *(por defecto)* 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - 15 minutos.

Conjunto de datos

Los siguientes parámetros son compatibles con la configuración de un *Conjunto de datos*.

Note:
El primer número del nombre de la propiedad (por ejemplo, ds.0.hosts.0, ds.0.items.0) representa el conjunto de datos correspondiente, mientras que el segundo número, si está presente, representa el host o el item configurado.

Parámetro	tipo	nombre	valor
Tipo de conjunto de datos	0	ds.0.dataset_type	0 - Lista de items; 1 - Patrones de items *(predeterminado)*.

Parámetro	tipo	nombre	valor
Items	4	ds.0.itemids.0	ID del item. Al configurar el widget en un dashboard de template, solo deben establecerse los items configurados en el template. Nota: Para configurar varios items, cree un objeto de campo de widget de dashboard para cada item, incrementando el número en el nombre de la propiedad. Comportamiento del parámetro: - *obligatorio* si *Tipo de conjunto de datos* está establecido en «Lista de items» y *Items (Widget)* no está establecido
Items (Widget)	1	ds.0.itemids.0._reference	En lugar del ID del item: `ABCDE._itemid` - establece un widget compatible (con su parámetro *Referencia* establecido en «ABCDE») como origen de datos para los items. Nota: Para configurar varios widgets, cree un objeto de campo de widget de dashboard para cada widget, incrementando el número en el nombre de la propiedad. Comportamiento del parámetro: - *obligatorio* si *Tipo de conjunto de datos* está establecido en «Lista de items» y *Items* no está establecido
Color	1	ds.0.color.0	Código de color hexadecimal (por ejemplo, FF0000). Comportamiento del parámetro: - *obligatorio* si *Tipo de conjunto de datos* está establecido en «Lista de items»
Tipo de item	0	ds.0.type.0	0 - Normal *(predeterminado)*; 1 - Total. El valor «Total» solo puede establecerse para un item en todo el gráfico. Comportamiento del parámetro: - *compatible* si *Tipo de conjunto de datos* está establecido en «Lista de items»
Patrones de host	1	ds.0.hosts.0	Nombre o patrón de host (por ejemplo, «Zabbix*»). Comportamiento del parámetro: - *obligatorio* si *Tipo de conjunto de datos* está establecido en «Patrones de items» Este parámetro no es compatible al configurar el widget en un dashboard de template.
Patrones de items	1	ds.0.items.0	Nombre o patrón del item (por ejemplo, «*: Número de *valores procesados por segundo»). Al configurar el widget en un dashboard de template, solo deben establecerse los patrones de los items configurados en el template. Comportamiento del parámetro: - *obligatorio* si *Tipo de conjunto de datos* está establecido en «Patrones de items»
Color	1	ds.0.color	Código de color hexadecimal (por ejemplo, FF0000). Comportamiento del parámetro: - *compatible* si *Tipo de conjunto de datos* está establecido en «Patrones de items» y *Paleta de colores* no está establecida

Parámetro	tipo	nombre	valor
Paleta de colores	0	ds.0.color_palette	Índice de la paleta de colores. Los valores posibles van de 0 a 11. Predeterminado: 0. Comportamiento del parámetro: - *compatible* si *Tipo de conjunto de datos* está establecido en «Patrones de items» y *Color* no está establecido
Función de agregación	0	ds.0.aggregate_function	1 - mínimo; 2 - máximo; 3 - promedio; 4 - recuento; 5 - suma; 6 - primero; 7 - último *(predeterminado)*.
Agregación del conjunto de datos	0	ds.0.dataset_aggregation	0 - ninguna *(predeterminado)*; 1 - mínimo; 2 - máximo; 3 - promedio; 4 - recuento; 5 - suma. Comportamiento del parámetro: - *compatible* si *Tipo de item* está establecido en «Total»
Etiqueta del conjunto de datos	1	ds.0.data_set_label	Cualquier valor de cadena. Predeterminado: "" (vacío).
Etiquetas de item			
Tipo de evaluación	0	ds.0.item_tags_evaltype	0 - Y/O *(predeterminado)*; 2 - O.
Nombre de etiqueta	1	ds.0.item_tags.0.tag	Cualquier valor de cadena. Nota: El número del nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* al configurar *Etiquetas de item*
Operador	0	ds.0.item_tags.0.operator	0 - Contiene; 1 - Es igual a; 2 - No contiene; 3 - No es igual a; 4 - Existe; 5 - No existe. Nota: El número del nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* al configurar *Etiquetas de item*

Parámetro	tipo	nombre	valor
Valor de etiqueta	1	ds.0.item_tags.0.value	Cualquier valor de cadena. Nota: El número del nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* al configurar *Etiquetas de item*

Opciones de visualización

Los siguientes parámetros son compatibles con la configuración de las *Opciones de visualización*.

Parámetro	tipo	nombre	valor
Selección de datos históricos	0	source	0 - *(predeterminado)* Automático; 1 - Historial; 2 - Tendencias.
Dibujar	0	draw_type	0 - *(predeterminado)* Circular; 1 - Anillo.
Ancho	0	width	20 - 20 % del radio; 30 - 30 % del radio; 40 - 40 % del radio; 50 - *(predeterminado)* 50 % del radio. Comportamiento del parámetro: - *compatible* si *Dibujar* está establecido en ”Anillo”
Ancho del trazo	0	stroke	Los valores posibles van de 0 a 10. Predeterminado: 0. Comportamiento del parámetro: - *compatible* si *Dibujar* está establecido en ”Anillo”
Mostrar valor total	0	total_show	0 - *(predeterminado)* Deshabilitado; 1 - Habilitado. Comportamiento del parámetro: - *compatible* si *Dibujar* está establecido en ”Anillo”
Tamaño	0	value_size_type	0 - *(predeterminado)* Automático; 1 - Personalizado. Comportamiento del parámetro: - *compatible* si *Mostrar valor total* está establecido en ”Habilitado”
Tamaño (valor para el tamaño personalizado)	0	value_size	Los valores posibles van de 1 a 100. Predeterminado: 20. Comportamiento del parámetro: - *compatible* si *Mostrar valor total* está establecido en ”Habilitado”
Posiciones decimales	0	decimal_places	Los valores posibles van de 0 a 6. Predeterminado: 2. Comportamiento del parámetro: - *compatible* si *Mostrar valor total* está establecido en ”Habilitado”
Unidades (casilla de verificación)	0	units_show	0 - *(predeterminado)* Deshabilitado; 1 - Habilitado. Comportamiento del parámetro: - *compatible* si *Mostrar valor total* está establecido en ”Habilitado”

Parámetro	tipo	nombre	valor
Unidades (valor)	1	units	Cualquier valor de cadena. Comportamiento del parámetro: - *compatible* si *Unidades* (casilla de verificación) está establecido en ”Habilitado”
Negrita	0	value_bold	0 - *(predeterminado)* Deshabilitado; 1 - Habilitado. Comportamiento del parámetro: - *compatible* si *Mostrar valor total* está establecido en ”Habilitado”
Color	1	value_color	Código de color hexadecimal (por ejemplo, FF0000). Comportamiento del parámetro: - *compatible* si *Mostrar valor total* está establecido en ”Habilitado”
Espacio entre sectores	0	space	Los valores posibles van de 0 a 10. Predeterminado: 1.
Combinar sectores menores que N % (casilla de verificación)	0	merge	0 - *(predeterminado)* Deshabilitado; 1 - Habilitado.
Combinar sectores menores que N % (valor)	0	merge_percent	Los valores posibles van de 1 a 10. Predeterminado: 1. Comportamiento del parámetro: - *compatible* si *Combinar sectores menores que N %* (casilla de verificación) está establecido en ”Habilitado”
Combinar sectores menores que N % (color)	1	merge_color	Código de color hexadecimal (por ejemplo, FF0000). Comportamiento del parámetro: - *compatible* si *Combinar sectores menores que N %* (casilla de verificación) está establecido en ”Habilitado”

Periodo de tiempo

Se admiten los siguientes parámetros para configurar el *Periodo de tiempo*.

Parámetro	tipo	nombre	valor
Periodo de tiempo	1	time_period._referencia	`DASHBOARD._timeperiod` - establece el selector de periodo de tiempo del dashboard como fuente de datos; `ABCDE._timeperiod` - establece un widget compatible (con su parámetro *Referencia* configurado en "`ABCDE`") como fuente de datos. Por defecto: `DASHBOARD._timeperiod` Alternativamente, puede establecer el periodo de tiempo solo en los parámetros *Desde* y *Hasta*.
Desde	1	time_period.from	Cadena de tiempo válida en sintaxis absoluta (`YYYY-MM-DD hh:mm:ss`) o relativa (`now`, `now/d`, `now/w-1w`, etc.). Comportamiento del parámetro: - *admitido* si *Periodo de tiempo* no está establecido - *obligatorio* si se establece `time_period.to`
Hasta	1	time_period.to	Cadena de tiempo válida en sintaxis absoluta (`YYYY-MM-DD hh:mm:ss`) o relativa (`now`, `now/d`, `now/w-1w`, etc.). Comportamiento del parámetro: - *admitido* si *Periodo de tiempo* no está establecido - *obligatorio* si se establece `time_period.from`

Leyenda

Los siguientes parámetros son compatibles con la configuración de *Leyenda*.

Parámetro	tipo	nombre	valor
Mostrar leyenda	0	legend	0 - Deshabilitado; 1 - *(predeterminado)* Habilitado.
Mostrar valor	0	legend_value	0 - *(predeterminado)* Deshabilitado; 1 - Habilitado. Comportamiento del parámetro: - *compatible* si *Mostrar leyenda* está establecido en "Habilitado"
Mostrar función de agregación	0	legend_aggregation	0 - *(predeterminado)* Deshabilitado; 1 - Habilitado. Comportamiento del parámetro: - *compatible* si *Mostrar leyenda* está establecido en "Habilitado"
Filas	0	legend_lines_mode	0 - *(predeterminado)* Fijas; 1 - Variables. Comportamiento del parámetro: - *compatible* si *Mostrar leyenda* está establecido en "Habilitado"
Número de filas/ Número máximo de filas	0	legend_lines	Los valores posibles van de 1 a 10. Predeterminado: 1. Comportamiento del parámetro: - *compatible* si *Mostrar leyenda* está establecido en "Habilitado"
Número de columnas	0	legend_columns	Los valores posibles van de 1 a 4. Predeterminado: 4. Comportamiento del parámetro: - *compatible* si *Mostrar leyenda* está establecido en "Habilitado" y *Mostrar valor* está establecido en "Deshabilitado"

Ejemplos

Los siguientes ejemplos tienen como objetivo describir únicamente la configuración de los objetos de campo del widget del panel para el widget *Gráfico circular*. Para obtener más información sobre la configuración de un panel, consulte `dashboard.create`.

Configuración de un widget *Gráfico circular*

Configure un widget *Gráfico circular* de la siguiente manera:

- 2 conjuntos de datos para un total de 9 items en 1 host.
- El primer conjunto de datos es de tipo ”Lista de items” y consta de 3 items que son todos de tipo ”Normal” y representados por un color diferente.
- El segundo conjunto de datos es de tipo ”Patrones de items”, consta de 6 items, tiene una agregación configurada para cada item y se representa con un color personalizado.
- El segundo conjunto de datos también tiene una etiqueta de conjunto de datos personalizada.
- Los datos en el gráfico circular se muestran como un gráfico de anillo con un ancho personalizado y el valor total con unidades en el centro.
- Los datos en el gráfico circular se muestran y agregan para un período de tiempo personalizado de las últimas 3 horas.
- La leyenda del gráfico circular muestra los items configurados en 4 filas.

Request:


```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "piechart",
                        "name": "Pie chart",
                        "x": 0,
                        "y": 0,
                        "width": 24,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 0,
                                "name": "ds.0.dataset_type",
                                "value": 0
                            },
                            {
                                "type": 4,
                                "name": "ds.0.itemids.0",
                                "value": 23264
                            },
                            {
                                "type": 1,
                                "name": "ds.0.color.0",
                                "value": "FF0000"
                            },
                            {
                                "type": 0,
                                "name": "ds.0.type.0",
                                "value": 0
                            },
                            {
                                "type": 4,
                                "name": "ds.0.itemids.1",
                                "value": 23269
                            },
                            {
```

```
                    "type": 1,
                    "name": "ds.0.color.1",
                    "value": "BF00FF"
                },
                {
                  "type": 0,
                  "name": "ds.0.type.1",
                  "value": 0
                },
                {
                    "type": 4,
                    "name": "ds.0.itemids.2",
                    "value": 23257
                },
                {
                    "type": 1,
                    "name": "ds.0.color.2",
                    "value": "0040FF"
                },
                {
                  "type": 0,
                  "name": "ds.0.type.2",
                  "value": 0
                },
                {
                    "type": 1,
                    "name": "ds.1.hosts.0",
                    "value": "Zabbix server"
                },
                {
                    "type": 1,
                    "name": "ds.1.items.0",
                    "value": "*: Number of processed *values per second"
                },
                {
                    "type": 1,
                    "name": "ds.1.color",
                    "value": "000000"
                },
                {
                    "type": 0,
                    "name": "ds.1.aggregate_function",
                    "value": 3
                },
                {
                    "type": 1,
                    "name": "ds.1.data_set_label",
                    "value": "Number of processed values per second"
                },
                {
                    "type": 0,
                    "name": "draw_type",
                    "value": 1
                },
                {
                    "type": 0,
                    "name": "width",
                    "value": 30
                },
                {
                    "type": 0,
                    "name": "total_show",
```

```
                                "value": 1
                            },
                            {
                                "type": 0,
                                "name": "units_show",
                                "value": 1
                            },
                            {
                                "type": 0,
                                "name": "graph_time",
                                "value": 1
                            },
                            {
                                "type": 1,
                                "name": "time_period.from",
                                "value": "now-3h"
                            },
                            {
                                "type": 1,
                                "name": "time_period.to",
                                "value": "now"
                            },
                            {
                                "type": 0,
                                "name": "legend_lines",
                                "value": 4
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```


Respuesta:


```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```


Véase también

- Campo de widget de panel
- `dashboard.create`

- `dashboard.update`

22 Equipos con problemas

Descripción

Estos parámetros y los posibles valores de propiedad para los objetos de campo del widget del dashboard correspondiente permiten configurar el widget *Problem hosts* en los métodos `dashboard.create` y `dashboard.update`.

Attention:
Las propiedades `fields` del widget no se validan durante la creación o actualización de un dashboard. Esto permite a los usuarios modificar widgets integrados y crear widgets personalizados, pero también introduce el riesgo de crear o actualizar widgets de forma incorrecta. Para garantizar la creación o actualización correcta del widget *Problem hosts*, consulte el comportamiento de los parámetros descrito en las tablas siguientes.

Parámetros

Se admiten los siguientes parámetros para el widget *Equipos con problemas*.

Parámetro	tipo	nombre	valor
Intervalo de actualización	0	rf_rate	0 - Sin actualización; 10 - 10 segundos; 30 - 30 segundos; 60 - *(por defecto)* 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - 15 minutos.
Grupos de equipos	2	groupids.0	ID de Grupo de equipos. Nota: Para configurar varios grupos de equipos, cree un objeto de campo de widget de panel para cada grupo de equipos con un número incrementado en el nombre de la propiedad. Este parámetro no es compatible si se configura el widget en un panel de plantilla.
Grupos de equipos (Widget)	1	groupids._reference	En lugar del ID de Grupo de equipos: `ABCDE._hostgroupids` - establezca un widget compatible (con su parámetro *Referencia* configurado en "ABCDE") como fuente de datos para los grupos de equipos. Este parámetro no es compatible si se configura el widget en un panel de plantilla.
Excluir grupos de equipos	2	exclude_groupids.0	ID de Grupo de equipos. Nota: Para excluir varios grupos de equipos, cree un objeto de campo de widget de panel para cada grupo de equipos con un número incrementado en el nombre de la propiedad. Este parámetro no es compatible si se configura el widget en un panel de plantilla.
Equipos	3	hostids.0	ID de Equipo. Nota: Para configurar varios equipos, cree un objeto de campo de widget de panel para cada equipo con un número incrementado en el nombre de la propiedad. Para varios equipos, el parámetro *Grupos de equipos* debe no estar configurado en absoluto o configurado con al menos un grupo de equipos al que pertenezcan los equipos configurados. Este parámetro no es compatible si se configura el widget en un panel de plantilla.

Parámetro	tipo	nombre	valor
Equipos (Widget/Panel)	1	hostids._reference	En lugar del ID de Equipo: `DASHBOARD.hostids` - establezca el selector de equipos del panel como fuente de datos para los equipos; `ABCDE._hostids` - establezca un widget compatible (con su parámetro *Referencia* configurado en "`ABCDE`") como fuente de datos para los equipos. Este parámetro no es compatible si se configura el widget en un panel de plantilla.
Problema	1	problem	Nombre del evento del problema (no distingue mayúsculas/minúsculas, nombre completo o parte de él).
Gravedad	0	severities.0	0 - No clasificado; 1 - Información; 2 - Advertencia; 3 - Media; 4 - Alta; 5 - Desastre. Por defecto: vacío (todas habilitadas). Nota: Para configurar varios valores, cree un objeto de campo de widget de panel para cada valor con un número incrementado en el nombre de la propiedad.
Etiquetas de problema			
Tipo de evaluación	0	evaltype	0 - *(por defecto)* Y/O; 2 - O.
Nombre de etiqueta	1	tags.0.tag	Cualquier valor de cadena. Nota: El número en el nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* si se configuran *Etiquetas de problema*
Operador	0	tags.0.operator	0 - Contiene; 1 - Igual a; 2 - No contiene; 3 - No igual a; 4 - Existe; 5 - No existe. Nota: El número en el nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* si se configuran *Etiquetas de problema*
Valor de etiqueta	1	tags.0.value	Cualquier valor de cadena. Nota: El número en el nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* si se configuran *Etiquetas de problema*
Mostrar problemas suprimidos	0	show_suppressed	0 - *(por defecto)* Deshabilitado; 1 - Habilitado.

Parámetro	tipo	nombre	valor
Ocultar grupos sin problemas	0	hide_empty_groups	0 - *(por defecto)* Deshabilitado; 1 - Habilitado. Este parámetro no es compatible si se configura el widget en un panel de plantilla.
Visualización de problemas	0	ext_ack	0 - *(por defecto)* Todos; 1 - Solo no reconocidos; 2 - Separados.
Referencia	1	reference	Cualquier valor de cadena que consista en 5 caracteres (por ejemplo, ABCDE o JBPNL). Este valor debe ser único dentro del panel al que pertenece el widget. Comportamiento del parámetro: - *obligatorio*

Ejemplos

Los siguientes ejemplos tienen como objetivo describir únicamente la configuración de los objetos de campo del widget de panel para el widget *Equipos con problemas*. Para obtener más información sobre la configuración de un panel, consulte dashboard.create.

Configuración de un widget *Equipos con problemas*

Configure un widget *Equipos con problemas* que muestre los equipos de los grupos de equipos ”2” y ”4” que tengan problemas con un nombre que incluya la cadena ”CPU” y que tengan las siguientes severidades: ”Aviso”, ”Media”, ”Alta”, ”Desastre”.

Petición:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "Mi panel",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "problemhosts",
                        "name": "Equipos con problemas",
                        "x": 0,
                        "y": 0,
                        "width": 36,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 2,
                                "name": "groupids.0",
                                "value": 2
                            },
                            {
                                "type": 2,
                                "name": "groupids.1",
                                "value": 4
                            },
                            {
                                "type": 1,
```

```
                                "name": "problem",
                                "value": "cpu"
                            },
                            {
                                "type": 0,
                                "name": "severities.0",
                                "value": 2
                            },
                            {
                                "type": 0,
                                "name": "severities.1",
                                "value": 3
                            },
                            {
                                "type": 0,
                                "name": "severities.2",
                                "value": 4
                            },
                            {
                                "type": 0,
                                "name": "severities.3",
                                "value": 5
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```


Respuesta:


```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```


Véase también

- Campo de widget de panel
- `dashboard.create`
- `dashboard.update`

23 Problemas

Descripción

Estos parámetros y los posibles valores de propiedad para los objetos de campo del widget del dashboard correspondiente permiten configurar el widget *Problems* en los métodos `dashboard.create` y `dashboard.update`.

Attention:
Las propiedades `fields` del widget no se validan durante la creación o actualización de un dashboard. Esto permite a los usuarios modificar widgets integrados y crear widgets personalizados, pero también introduce el riesgo de crear o actualizar widgets de forma incorrecta. Para garantizar la creación o actualización correctas del widget *Problems*, consulte el comportamiento de los parámetros descrito en las tablas siguientes.

Parámetros

Los siguientes parámetros son compatibles con el widget *Problemas*.

Parámetro	type	name	value
Intervalo de actualización	0	rf_rate	0 - Sin actualización; 10 - 10 segundos; 30 - 30 segundos; 60 - *(predeterminado)* 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - 15 minutos.
Mostrar	0	show	1 - *(predeterminado)* Problemas recientes; 2 - Historial; 3 - Problemas.
Grupos de hosts	2	groupids.0	ID del grupo de hosts. Nota: Para configurar varios grupos de hosts, cree un objeto de campo del widget del dashboard para cada grupo de hosts, con un número incrementado en el nombre de la propiedad. Este parámetro no es compatible si se configura el widget en un dashboard de template.
Grupos de hosts (widget)	1	groupids._reference	En lugar del ID del grupo de hosts: `ABCDE._hostgroupids` - establezca un widget compatible (con su parámetro *Referencia* establecido en "ABCDE") como fuente de datos para los grupos de hosts. Este parámetro no es compatible si se configura el widget en un dashboard de template.
Excluir grupos de hosts	2	exclude_groupids.0	ID del grupo de hosts. Nota: Para excluir varios grupos de hosts, cree un objeto de campo del widget del dashboard para cada grupo de hosts, con un número incrementado en el nombre de la propiedad. Este parámetro no es compatible si se configura el widget en un dashboard de template.
Hosts	3	hostids.0	ID del host. Nota: Para configurar varios hosts, cree un objeto de campo del widget del dashboard para cada host, con un número incrementado en el nombre de la propiedad. Para varios hosts, el parámetro *Grupos de hosts* debe no estar configurado o estar configurado con al menos un grupo de hosts al que pertenezcan los hosts configurados. Este parámetro no es compatible si se configura el widget en un dashboard de template.

Parámetro	type	name	value
Hosts (widget/dashboard)	1	hostids._reference	En lugar del ID del host: `DASHBOARD.hostids` - establezca el selector de hosts del dashboard como fuente de datos para los hosts; `ABCDE._hostids` - establezca un widget compatible (con su parámetro *Referencia* establecido en "ABCDE") como fuente de datos para los hosts. Este parámetro no es compatible si se configura el widget en un dashboard de template.
Problema	1	problem	Nombre del evento del problema (no distingue entre mayúsculas y minúsculas, nombre completo o parte del nombre).
Gravedad	0	severities.0	0 - No clasificado; 1 - Información; 2 - Advertencia; 3 - Promedio; 4 - Alto; 5 - Desastre. Predeterminado: vacío (todos habilitados). Nota: Para configurar varios valores, cree un objeto de campo del widget del dashboard para cada valor, con un número incrementado en el nombre de la propiedad.
Etiquetas del problema			
Tipo de evaluación	0	evaltype	0 - *(predeterminado)* Y/O; 2 - O.
Nombre de la etiqueta	1	tags.0.tag	Cualquier valor de cadena. Nota: El número del nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* si se configuran las *Etiquetas del problema*
Operador	0	tags.0.operator	0 - Contiene; 1 - Es igual a; 2 - No contiene; 3 - No es igual a; 4 - Existe; 5 - No existe. Nota: El número del nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* si se configuran las *Etiquetas del problema*
Valor de la etiqueta	1	tags.0.value	Cualquier valor de cadena. Nota: El número del nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* si se configuran las *Etiquetas del problema*
Mostrar etiquetas	0	show_tags	0 - *(predeterminado)* Ninguna; 1 - 1; 2 - 2; 3 - 3.

Parámetro	type	name	value
Nombre de la etiqueta (formato)	0	tag_name_format	0 - *(predeterminado)* Completo; 1 - Abreviado; 2 - Ninguno. Comportamiento del parámetro: - *compatible* si *Mostrar etiquetas* está establecido en ”1”, ”2” o ”3”
Prioridad de visualización de etiquetas	1	tag_priority	Lista de etiquetas separadas por comas. Comportamiento del parámetro: - *compatible* si *Mostrar etiquetas* está establecido en ”1”, ”2” o ”3”
Mostrar datos operativos	0	show_opdata	0 - *(predeterminado)* Ninguno; 1 - Por separado; 2 - Con el nombre del problema.
Mostrar problemas suprimidos	0	show_suppressed	0 - *(predeterminado)* Deshabilitado; 1 - Habilitado.
Estado de reconocimiento	0	acknowledgement_status	0 - *(predeterminado)* Todos; 1 - No reconocidos; 2 - Reconocidos.
Por mí	0	acknowledged_by_me	0 - *(predeterminado)* Deshabilitado; 1 - Habilitado.
Ordenar entradas por	0	sort_triggers	1 - Gravedad (descendente); 2 - Host (ascendente); 3 - Hora (ascendente); 4 - *(predeterminado)* Hora (descendente); 13 - Gravedad (ascendente); 14 - Host (descendente); 15 - Problema (ascendente); 16 - Problema (descendente). Para todos los valores, excepto ”Hora (descendente)” y ”Hora (ascendente)”, el parámetro *Mostrar línea de tiempo* debe estar establecido en ”Deshabilitado”. Los valores ”Host (ascendente)” y ”Host (descendente)” no son compatibles si se configura el widget en un dashboard de template.
Mostrar línea de tiempo	0	show_timeline	0 - Deshabilitado; 1 - *(predeterminado)* Habilitado. Comportamiento del parámetro: - *compatible* si *Ordenar entradas por* está establecido en ”Hora (descendente)” o ”Hora (ascendente)”
Resaltar fila completa	0	highlight_row	0 - *(predeterminado)* Deshabilitado; 1 - Habilitado. Este parámetro no es compatible con los temas de alto contraste.

Parámetro	type	name	value
Mostrar líneas	0	show_lines	Los valores posibles van de 1 a 100. Predeterminado: 25.
Referencia	1	reference	Cualquier valor de cadena compuesto por 5 caracteres (por ejemplo, `ABCDE` o `JBPNL`). Este valor debe ser único dentro del dashboard al que pertenece el widget. Comportamiento del parámetro: - *obligatorio*

Ejemplos

Los siguientes ejemplos tienen como objetivo describir únicamente la configuración de los objetos de campo del widget de panel para el widget *Problemas*. Para obtener más información sobre cómo configurar un panel, consulte `dashboard.create`.

Configuración de un widget *Problemas*

Configure un widget *Problemas* que muestre los problemas para el grupo de equipos "4" que cumplan las siguientes condiciones:

- Problemas que tengan una etiqueta con el nombre "scope" que contenga los valores "performance" o "availability", o "capacity".
- Problemas que tengan las siguientes severidades: "Aviso", "Media", "Alta", "Desastre".

Además, configure el widget para mostrar etiquetas y datos operativos.

Petición:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "Mi panel",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "problems",
                        "name": "Problemas",
                        "x": 0,
                        "y": 0,
                        "width": 36,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 2,
                                "name": "groupids.0",
                                "value": 4
                            },
                            {
                                "type": 1,
                                "name": "tags.0.tag",
                                "value": "scope"
                            },
                            {
                                "type": 0,
                                "name": "tags.0.operator",
                                "value": 0
                            },
                            {
                                "type": 1,
                                "name": "tags.0.value",
```

```
            "value": "performance"
        },
        {
            "type": 1,
            "name": "tags.1.tag",
            "value": "scope"
        },
        {
            "type": 0,
            "name": "tags.1.operator",
            "value": 0
        },
        {
            "type": 1,
            "name": "tags.1.value",
            "value": "availability"
        },
        {
            "type": 1,
            "name": "tags.2.tag",
            "value": "scope"
        },
        {
            "type": 0,
            "name": "tags.2.operator",
            "value": 0
        },
        {
            "type": 1,
            "name": "tags.2.value",
            "value": "capacity"
        },
        {
            "type": 0,
            "name": "severities.0",
            "value": 2
        },
        {
            "type": 0,
            "name": "severities.1",
            "value": 3
        },
        {
            "type": 0,
            "name": "severities.2",
            "value": 4
        },
        {
            "type": 0,
            "name": "severities.3",
            "value": 5
        },
        {
            "type": 0,
            "name": "show_tags",
            "value": 1
        },
        {
            "type": 0,
            "name": "show_opdata",
            "value": 1
        }
```

```
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Véase también

- Campo de widget de panel
- `dashboard.create`
- `dashboard.update`

24 Problemas por severidad

Descripción

Estos parámetros y los posibles valores de propiedad para los objetos de campo del widget del dashboard correspondiente permiten configurar el widget *Problemas por gravedad* en los métodos `dashboard.create` y `dashboard.update`.

Attention:
Las propiedades `fields` del widget no se validan durante la creación o actualización de un dashboard. Esto permite a los usuarios modificar widgets integrados y crear widgets personalizados, pero también introduce el riesgo de crear o actualizar widgets de forma incorrecta. Para garantizar la creación o actualización correcta del widget *Problemas por gravedad*, consulte el comportamiento de los parámetros descrito en las tablas siguientes.

Parámetros

Los siguientes parámetros son compatibles con el widget *Problemas por gravedad*.

Parámetro	tipo	nombre	valor
Intervalo de actualización	0	rf_rate	0 - Sin actualización; 10 - 10 segundos; 30 - 30 segundos; 60 - *(por defecto)* 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - 15 minutos.
Grupos de equipos	2	groupids.0	ID de Grupo de equipos. Nota: Para configurar varios grupos de equipos, cree un objeto de campo de widget de panel para cada grupo de equipos con un número incrementado en el nombre de la propiedad. Este parámetro no es compatible si se configura el widget en un panel de plantilla.
Grupos de equipos (Widget)	1	groupids._reference	En lugar del ID de Grupo de equipos: `ABCDE._hostgroupids` - establezca un widget compatible (con su parámetro *Referencia* configurado en "`ABCDE`") como fuente de datos para los grupos de equipos. Este parámetro no es compatible si se configura el widget en un panel de plantilla.
Excluir grupos de equipos	2	exclude_groupids.0	ID de Grupo de equipos. Nota: Para excluir varios grupos de equipos, cree un objeto de campo de widget de panel para cada grupo de equipos con un número incrementado en el nombre de la propiedad. Este parámetro no es compatible si se configura el widget en un panel de plantilla.
Equipos	3	hostids.0	ID de Equipo. Nota: Para configurar varios equipos, cree un objeto de campo de widget de panel para cada equipo con un número incrementado en el nombre de la propiedad. Para varios equipos, el parámetro *Grupos de equipos* debe no estar configurado en absoluto o configurado con al menos un grupo de equipos al que pertenezcan los equipos configurados. Este parámetro no es compatible si se configura el widget en un panel de plantilla.
Equipos (Widget/Panel)	1	hostids._reference	En lugar del ID de Equipo: `DASHBOARD.hostids` - establezca el selector de equipos del panel como fuente de datos para los equipos; `ABCDE._hostids` - establezca un widget compatible (con su parámetro *Referencia* configurado en "`ABCDE`") como fuente de datos para los equipos. Este parámetro no es compatible si se configura el widget en un panel de plantilla.
Problema	1	problem	Nombre del evento del problema (no distingue mayúsculas/minúsculas, nombre completo o parte de él).

Parámetro	tipo	nombre	valor
Gravedad	0	severities.0	0 - No clasificado; 1 - Información; 2 - Advertencia; 3 - Media; 4 - Alta; 5 - Desastre. Por defecto: vacío (todas habilitadas). Nota: Para configurar varios valores, cree un objeto de campo de widget de panel para cada valor con un número incrementado en el nombre de la propiedad.
Etiquetas de problema			
Tipo de evaluación	0	evaltype	0 - *(por defecto)* Y/O; 2 - O.
Nombre de etiqueta	1	tags.0.tag	Cualquier valor de cadena. Nota: El número en el nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* si se configuran *Etiquetas de problema*
Operador	0	tags.0.operator	0 - Contiene; 1 - Igual a; 2 - No contiene; 3 - No igual a; 4 - Existe; 5 - No existe. Nota: El número en el nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* si se configuran *Etiquetas de problema*
Valor de etiqueta	1	tags.0.value	Cualquier valor de cadena. Nota: El número en el nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* si se configuran *Etiquetas de problema*
Mostrar	0	show_type	0 - *(por defecto)* Grupos de equipos; 1 - Totales. Este parámetro no es compatible si se configura el widget en un panel de plantilla, y por defecto se establece en "Totales".
Disposición	0	layout	0 - *(por defecto)* Horizontal; 1 - Vertical. Comportamiento del parámetro: - *compatible* si *Mostrar* está configurado en "Totales"
Mostrar datos operativos	0	show_opdata	0 - *(por defecto)* Ninguno; 1 - Por separado; 2 - Con el nombre del problema.

Parámetro	tipo	nombre	valor
Mostrar problemas suprimidos	0	show_suppressed	0 - *(por defecto)* Deshabilitado; 1 - Habilitado.
Ocultar grupos sin problemas	0	hide_empty_groups	0 - *(por defecto)* Deshabilitado; 1 - Habilitado. Comportamiento del parámetro: - *compatible* si *Mostrar* está configurado en "Grupos de equipos" Este parámetro no es compatible si se configura el widget en un panel de plantilla.
Visualización de problemas	0	ext_ack	0 - *(por defecto)* Todos; 1 - Solo no reconocidos; 2 - Separados.
Mostrar línea de tiempo	0	show_timeline	0 - Deshabilitado; 1 - *(por defecto)* Habilitado.
Referencia	1	reference	Cualquier valor de cadena que consista en 5 caracteres (por ejemplo, `ABCDE` o `JBPNL`). Este valor debe ser único dentro del panel al que pertenece el widget. Comportamiento del parámetro: - *obligatorio*

Ejemplos

Los siguientes ejemplos tienen como objetivo describir únicamente la configuración de los objetos de campo del widget del panel para el widget *Problemas por severidad*. Para obtener más información sobre la configuración de un panel, consulte `dashboard.create`.

Configuración de un widget *Problemas por gravedad*

Configure un widget *Problemas por gravedad* que muestre los totales de problemas para todos los grupos de equipos.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "Mi panel",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "problemsbysv",
                        "name": "Problemas por gravedad",
                        "x": 0,
                        "y": 0,
                        "width": 36,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
```

```
                                "type": 0,
                                "name": "show_type",
                                "value": 1
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Véase también

- Campo de widget de panel
- `dashboard.create`
- `dashboard.update`

25 Gráfico de dispersión

Descripción

Estos parámetros y los posibles valores de propiedad para los objetos de campo del widget del dashboard correspondiente permiten configurar el widget *Gráfico de dispersión* en los métodos `dashboard.create` y `dashboard.update`.

Attention:
Las propiedades `fields` del widget no se validan durante la creación o actualización de un dashboard. Esto permite a los usuarios modificar widgets integrados y crear widgets personalizados, pero también introduce el riesgo de crear o actualizar widgets de forma incorrecta. Para garantizar la creación o actualización correcta del widget *Gráfico de dispersión*, consulte el comportamiento de los parámetros descrito en las tablas siguientes.

Parámetros

Los siguientes parámetros son compatibles con el widget *Gráfico de dispersión*.

Parámetro	tipo	nombre	valor
Intervalo de actualización	0	rf_rate	0 - Sin actualización; 10 - 10 segundos; 30 - 30 segundos; 60 - *(por defecto)* 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - 15 minutos.
Referencia	1	reference	Cualquier valor de cadena que consista en 5 caracteres (por ejemplo, `ABCDE` o `JBPNL`). Este valor debe ser único dentro del dashboard al que pertenece el widget. Comportamiento del parámetro: - *obligatorio*

Conjunto de datos

Se admiten los siguientes parámetros para configurar un *Conjunto de datos*.

Note:
El primer número en el nombre de la propiedad (por ejemplo, ds.0.hosts.0, ds.0.items.0) representa el conjunto de datos en particular, mientras que el segundo número, si está presente, representa el host o item configurado.

Parámetro	tipo	nombre	valor
Tipo de conjunto de datos ***Conjunto de datos: Lista de items***	0	ds.0.dataset_type	0 - Lista de items; 1 - *(por defecto)* Patrones de items.
Color	1	ds.0.color.0	Código de color hexadecimal (por ejemplo, `FF0000`). Comportamiento del parámetro: - *requerido* si *Tipo de conjunto de datos* está configurado como "Lista de items"
Eje X	4	ds.0.x_axis_itemids.0	ID del item. Al configurar el widget en un dashboard de template, solo se deben establecer los items configurados en el template. Nota: Para configurar varios items, cree un objeto de campo de widget de dashboard para cada item con un número incrementado en el nombre de la propiedad. Comportamiento del parámetro: - *requerido* si *Tipo de conjunto de datos* está configurado como "Lista de items" y *Eje X (Widget)* no está configurado

Parámetro	tipo	nombre	valor
Eje X (Widget)	1	ds.0.x_axis_itemids.0	Referencia del ID del item: `ABCDE._itemid` - establezca un widget compatible (con su parámetro *Referencia* configurado como "ABCDE") como fuente de datos para los items. Nota: Para configurar varios widgets, cree un objeto de campo de widget de dashboard para cada widget con un número incrementado en el nombre de la propiedad. Comportamiento del parámetro: - *requerido* si *Tipo de conjunto de datos* está configurado como "Lista de items" y *Eje X* no está configurado
Eje Y	4	ds.0.y_axis_itemids.0	ID del item. Al configurar el widget en un dashboard de template, solo se deben establecer los items configurados en el template. Nota: Para configurar varios items, cree un objeto de campo de widget de dashboard para cada item con un número incrementado en el nombre de la propiedad. Comportamiento del parámetro: - *requerido* si *Tipo de conjunto de datos* está configurado como "Lista de items" y *Eje Y (Widget)* no está configurado
Eje Y (Widget)	1	ds.0.y_axis_itemids.0	Referencia del ID del item: `ABCDE._itemid` - establezca un widget compatible (con su parámetro *Referencia* configurado como "ABCDE") como fuente de datos para los items. Nota: Para configurar varios widgets, cree un objeto de campo de widget de dashboard para cada widget con un número incrementado en el nombre de la propiedad. Comportamiento del parámetro: - *requerido* si *Tipo de conjunto de datos* está configurado como "Lista de items" y *Eje Y* no está configurado
Sobrescribir host	1	ds.0.override_hostid.0	`ABCDE._hostid`- establezca un widget compatible (con su parámetro "Referencia" configurado como "ABCDE") como fuente de datos para los hosts; `DASHBOARD._hostid` - establezca el selector de host del dashboard como fuente de datos para los hosts. Este parámetro no es compatible si se configura el widget en un dashboard de template.
Conjunto de datos: Patrones de items			
Color	1	ds.0.color	Código de color hexadecimal (por ejemplo, `FF0000`). Comportamiento del parámetro: - *soportado* si *Tipo de conjunto de datos* está configurado como "Patrones de items" y *Paleta de colores* no está configurada

Parámetro	tipo	nombre	valor
Paleta de colores	0	ds.0.color_palette	Índice de la paleta de colores. Valores posibles: 0-11. Por defecto: 0. Comportamiento del parámetro: - *soportado* si *Tipo de conjunto de datos* está configurado como "Patrones de items" y *Color* no está configurado
Patrones de host	1	ds.0.hosts.0	Nombre o patrón del host (por ejemplo, "Zabbix*"). Comportamiento del parámetro: - *requerido* si *Tipo de conjunto de datos* está configurado como "Patrones de items" Este parámetro no es compatible si se configura el widget en un dashboard de template.
Patrones de items del eje X	1	ds.0.x_axis_items.0	Nombre o patrón del item (por ejemplo, "*: Número de valores procesados por segundo"). Al configurar el widget en un dashboard de template, solo se deben establecer los patrones para los items configurados en el template. Comportamiento del parámetro: - *requerido* si *Tipo de conjunto de datos* está configurado como "Patrones de items"
Patrones de items del eje Y	1	ds.0.y_axis_items.0	Nombre o patrón del item (por ejemplo, "*: Número de valores procesados por segundo"). Al configurar el widget en un dashboard de template, solo se deben establecer los patrones para los items configurados en el template. Comportamiento del parámetro: - *requerido* si *Tipo de conjunto de datos* está configurado como "Patrones de items"
Grupos de hosts	2	groupids.0	ID del grupo de hosts. Nota: Para configurar varios grupos de hosts, cree un objeto de campo de widget de dashboard para cada grupo de hosts con un número incrementado en el nombre de la propiedad. Este parámetro no es compatible si se configura el widget en un dashboard de template.
Grupos de hosts (Widget)	1	groupids._reference	En lugar del ID del grupo de hosts: `ABCDE._hostgroupids` - establezca un widget compatible (con su parámetro *Referencia* configurado como "ABCDE") como fuente de datos para los grupos de hosts. Este parámetro no es compatible si se configura el widget en un dashboard de template.
Sobrescribir host	1	ds.0.override_hostid._reference	En lugar del ID del host: `ABCDE._hostid`- establezca un widget compatible (con su parámetro "Referencia" configurado como "ABCDE") como fuente de datos para los hosts; `DASHBOARD._hostid` - establezca el selector de host del dashboard como fuente de datos para los hosts. Este parámetro no es compatible si se configura el widget en un dashboard de template.
Etiquetas de host			

Parámetro	tipo	nombre	valor
Tipo de evaluación	0	evaltype_host	0 - *(por defecto)* Y/O; 2 - O. Este parámetro no es compatible si se configura el widget en un dashboard de template.
Nombre de etiqueta	1	host_tags.0.tag	Cualquier valor de cadena. Nota: El número en el nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *requerido* si se configuran *Etiquetas de host* Este parámetro no es compatible si se configura el widget en un dashboard de template.
Operador	0	host_tags.0.operator	0 - Contiene; 1 - Igual; 2 - No contiene; 3 - No igual; 4 - Existe; 5 - No existe. Nota: El número en el nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *requerido* si se configuran *Etiquetas de host* Este parámetro no es compatible si se configura el widget en un dashboard de template.
Valor de etiqueta	1	host_tags.0.value	Cualquier valor de cadena. Nota: El número en el nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *requerido* si se configuran *Etiquetas de host* Este parámetro no es compatible si se configura el widget en un dashboard de template.
Marcador	0	ds.0.marker	0 - *(por defecto)* Elipse; 1 - Cuadrado; 2 - Triángulo; 3 - Diamante; 4 - Estrella; 5 - Cruz.
Tamaño del marcador	0	ds.0.marker_size	0 - *(por defecto)* Pequeño (6px); 1 - Mediano (9px); 2 - Grande (15px).
Desplazamiento de tiempo	1	ds.0.timeshift	Cadena de tiempo válida (por ejemplo, 3600, 1h, etc.). Puede utilizar sufijos de tiempo. También se permiten valores negativos. Por defecto: "" (vacío).
Intervalo de agregación	1	ds.0.aggregate_interval	Cadena de tiempo válida (por ejemplo, 3600, 1h, etc.). Puede utilizar sufijos de tiempo. Por defecto: 15m.

Parámetro	tipo	nombre	valor
Función de agre-gación	0	ds.0.aggregate_function	0 - no se utiliza; 1 - min; 2 - max; 3 - *(por defecto)* avg; 4 - count; 5 - sum; 6 - first; 7 - last.

Opciones de visualización

Se admiten los siguientes parámetros para configurar las *Opciones de visualización*.

Parámetro	tipo	nombre	valor
Selección de datos históricos	0	source	0 - *(por defecto)* Automático; 1 - Historial; 2 - Tendencias.
Nombres de host en etiquetas	0	show_hostnames	0 - *(por defecto)* Automático; 1 - Mostrar; 2 - Ocultar. Este parámetro no es compatible si se configura el widget en un dashboard de template.

Periodo de tiempo

Se admiten los siguientes parámetros para configurar el *Periodo de tiempo*.

Parámetro	tipo	nombre	valor
Periodo de tiempo	1	time_period._reference	`DASHBOARD._timeperiod` - establece el selector de periodo de tiempo del dashboard como fuente de datos; `ABCDE._timeperiod` - establece un widget compatible (con su parámetro *Referencia* configurado como "`ABCDE`") como fuente de datos. Por defecto: `DASHBOARD._timeperiod` Alternativamente, puede establecer el periodo de tiempo solo en los parámetros *Desde* y *Hasta*.
Desde	1	time_period.from	Cadena de tiempo válida en sintaxis absoluta (`YYYY-MM-DD hh:mm:ss`) o relativa (`now`, `now/d`, `now/w-1w`, etc.). Comportamiento del parámetro: - *admitido* si *Periodo de tiempo* no está establecido - *obligatorio* si se establece `time_period.to`
Hasta	1	time_period.to	Cadena de tiempo válida en sintaxis absoluta (`YYYY-MM-DD hh:mm:ss`) o relativa (`now`, `now/d`, `now/w-1w`, etc.). Comportamiento del parámetro: - *admitido* si *Periodo de tiempo* no está establecido - *obligatorio* si se establece `time_period.from`

Ejes

Se admiten los siguientes parámetros para configurar los *Ejes*.

Parámetro	tipo	nombre	valor
Eje X	0	x_axis	0 - Deshabilitado; 1 - *(por defecto)* Habilitado.
Eje Y	0	y_axis	0 - *(por defecto)* Deshabilitado; 1 - Habilitado.
Mín	1	x_axis_min	Cualquier valor numérico. Por defecto: "" (vacío).
		y_axis_min	
Máx	1	x_axis_max	Cualquier valor numérico. Por defecto: "" (vacío).
		y_axis_max	
Unidades (tipo)	0	x_axis_units	0 - *(por defecto)* Auto; 1 - Estático.
		y_axis_units	
Unidades (valor)	1	x_axis_static_units	Cualquier valor de cadena. Por defecto: "" (vacío).
		y_axis_static_units	

Leyenda

Los siguientes parámetros son compatibles para configurar la *Leyenda*.

Parámetro	tipo	nombre	valor
Mostrar leyenda	0	legend	0 - Deshabilitado; 1 - *(por defecto)* Habilitado.
Mostrar función de agregación	0	legend_aggregation	0 - *(por defecto)* Deshabilitado; 1 - Habilitado. Comportamiento del parámetro: - *soportado* si *Mostrar leyenda* está configurado como ”Habilitado”
Filas	0	legend_lines_mode	0 - *(por defecto)* Fijo; 1 - Variable. Comportamiento del parámetro: - *soportado* si *Mostrar leyenda* está configurado como ”Habilitado”
Número de filas/ Número máximo de filas	0	legend_lines	Rango de valores posibles: 1-10. Por defecto: 1. Comportamiento del parámetro: - *soportado* si *Mostrar leyenda* está configurado como ”Habilitado”
Número de columnas	0	legend_columns	Rango de valores posibles: 1-4. Por defecto: 4. Comportamiento del parámetro: - *soportado* si *Mostrar leyenda* está configurado como ”Habilitado”

Umbrales

Se admiten los siguientes parámetros para configurar los *Umbrales*.

Note:
El número en el nombre de la propiedad *Thresholds* (por ejemplo, thresholds.0.color) hace referencia a la posición del umbral en una lista, ordenada en orden ascendente. Sin embargo, si los umbrales se configuran en un orden diferente, los valores se ordenarán en orden ascendente después de actualizar la configuración del widget en el frontend de Zabbix (por ejemplo, `"thresholds.0.threshold":"5"` → `"thresholds.0.threshold":"1"`; `"thresholds.1.threshold":"1"` → `"thresholds.1.threshold": "5"`).

Parámetro	tipo	nombre	valor
Interpolación de color	0	interpolation	0 - *(por defecto)* Deshabilitado; 1 - Habilitado.
Color	1	thresholds.0.color	Código de color hexadecimal (por ejemplo, FF0000).
Eje X	1	thresholds.0.x_axis_threshold	Cualquier valor numérico. Se admiten Sufijos (por ejemplo, ”1d”, ”2w”, ”4K”, ”8G”).
Eje Y	1	thresholds.0.y_axis_threshold	Cualquier valor numérico. Se admiten Sufijos (por ejemplo, ”1d”, ”2w”, ”4K”, ”8G”).

Ejemplos

Los siguientes ejemplos tienen como objetivo describir únicamente la configuración de los objetos de campo del widget para el widget *Gráfico de dispersión*. Para obtener más información sobre la configuración de un dashboard, consulte `dashboard.create`.

Configuración de un widget *Gráfico de dispersión*

Configure un widget *Gráfico de dispersión* de la siguiente manera:

- 5 conjuntos de datos de tipo ”Patrones de item”, cada uno con la agregación predeterminada (avg, 15m) y diferentes marcadores.
- La leyenda muestra los items configurados en una sola fila y dos columnas.
- Los umbrales están configurados para colorear un marcador de rojo cuando su valor alcanza 80 en cualquier eje.

Request:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "scatterplot",
                        "name": "Scatter plot",
                        "x": 0,
                        "y": 0,
                        "width": 36,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 1,
                                "name": "ds.0.color",
                                "value": "0C5A87"
                            },
                            {
                                "type": 1,
                                "name": "ds.0.hosts.0",
                                "value": "de-frankfurt*"
                            },
                            {
                                "type": 1,
                                "name": "ds.0.x_axis_items.0",
                                "value": "Memory utilization"
                            },
                            {
                                "type": 1,
                                "name": "ds.0.y_axis_items.0",
                                "value": "CPU utilization"
                            },
                            {
```

```
                "type": 0,
                "name": "ds.0.marker",
                "value": 0
            },
            {
                "type": 0,
                "name": "ds.0.marker_size",
                "value": 1
            },
            {
                "type": 1,
                "name": "ds.1.color",
                "value": "66B0D9"
            },
            {
                "type": 1,
                "name": "ds.1.hosts.0",
                "value": "fr-paris*"
            },
            {
                "type": 1,
                "name": "ds.1.x_axis_items.0",
                "value": "Memory utilization"
            },
            {
                "type": 1,
                "name": "ds.1.y_axis_items.0",
                "value": "CPU utilization"
            },
            {
                "type": 0,
                "name": "ds.1.marker",
                "value": 1
            },
            {
                "type": 0,
                "name": "ds.1.marker_size",
                "value": 1
            },
            {
                "type": 1,
                "name": "ds.2.color",
                "value": "0A466A"
            },
            {
                "type": 1,
                "name": "ds.2.hosts.0",
                "value": "lv-riga*"
            },
            {
                "type": 1,
                "name": "ds.2.x_axis_items.0",
                "value": "Memory utilization"
            },
            {
                "type": 1,
                "name": "ds.2.y_axis_items.0",
                "value": "CPU utilization"
            },
            {
                "type": 0,
                "name": "ds.2.marker",
```

```
            "value": 2
        },
        {
            "type": 0,
            "name": "ds.2.marker_size",
            "value": 1
        },
        {
            "type": 1,
            "name": "ds.3.color",
            "value": "3394C3"
        },
        {
            "type": 1,
            "name": "ds.3.hosts.0",
            "value": "pl-warsaw*"
        },
        {
            "type": 1,
            "name": "ds.3.x_axis_items.0",
            "value": "Memory utilization"
        },
        {
            "type": 1,
            "name": "ds.3.y_axis_items.0",
            "value": "CPU utilization"
        },
        {
            "type": 0,
            "name": "ds.3.marker",
            "value": 3
        },
        {
            "type": 0,
            "name": "ds.3.marker_size",
            "value": 1
        },
        {
            "type": 1,
            "name": "ds.4.color",
            "value": "1492C8"
        },
        {
            "type": 1,
            "name": "ds.4.hosts.0",
            "value": "se-stockholm*"
        },
        {
            "type": 1,
            "name": "ds.4.x_axis_items.0",
            "value": "Memory utilization"
        },
        {
            "type": 1,
            "name": "ds.4.y_axis_items.0",
            "value": "CPU utilization"
        },
        {
            "type": 0,
            "name": "ds.4.marker",
            "value": 4
        },
```

```
                        {
                            "type": 0,
                            "name": "ds.4.marker_size",
                            "value": 1
                        },
                        {
                            "type": 0,
                            "name": "legend_columns",
                            "value": 2
                        },
                        {
                            "type": 0,
                            "name": "interpolation",
                            "value": 1
                        },
                        {
                            "type": 1,
                            "name": "thresholds.0.color",
                            "value": "D40000"
                        },
                        {
                            "type": 1,
                            "name": "thresholds.0.x_axis_threshold",
                            "value": "80"
                        },
                        {
                            "type": 1,
                            "name": "thresholds.0.y_axis_threshold",
                            "value": ""
                        },
                        {
                            "type": 1,
                            "name": "thresholds.1.color",
                            "value": "D40000"
                        },
                        {
                            "type": 1,
                            "name": "thresholds.1.x_axis_threshold",
                            "value": ""
                        },
                        {
                            "type": 1,
                            "name": "thresholds.1.y_axis_threshold",
                            "value": "80"
                        },
                        {
                            "type": 1,
                            "name": "reference",
                            "value": "JQISY"
                        }
                    ]
                }
            ]
        }
    ],
    "userGroups": [
        {
            "usrgrpid": 7,
            "permission": 2
        }
    ],
    "users": [
```

```
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Ver también

- Campo de widget de dashboard
- `dashboard.create`
- `dashboard.update`

26 Informe de SLA

Descripción

Estos parámetros y los posibles valores de propiedad para los objetos de campo correspondientes del widget del dashboard permiten configurar el widget *Informe de SLA* en los métodos `dashboard.create` y `dashboard.update`.

Attention:
Las propiedades `fields` del widget no se validan durante la creación o actualización de un dashboard. Esto permite a los usuarios modificar widgets integrados y crear widgets personalizados, pero también introduce el riesgo de crear o actualizar widgets de forma incorrecta. Para garantizar la creación o actualización correcta del widget *Informe de SLA*, consulte el comportamiento de los parámetros descrito en las tablas siguientes.

Parámetros

Los siguientes parámetros son compatibles con el widget de *informe de SLA*.

Parámetro	tipo	nombre	valor
Intervalo de actualización	0	rf_rate	0 - *(predeterminado)* Sin actualización; 10 - 10 segundos; 30 - 30 segundos; 60 - 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - 15 minutos.
SLA	10	slaid.0	ID de SLA. Comportamiento del parámetro: - *obligatorio*
Servicio	9	serviceid.0	ID del servicio.
Mostrar períodos	0	show_periods	Los valores posibles van de 1 a 100. Predeterminado: 20.
Desde	1	date_from	Cadena de fecha válida con el formato `YYYY-MM-DD`. Se admiten fechas relativas con los modificadores `d`, `w`, `M`, `y` (por ejemplo, `now`, `now/d`, `now/w-1w`, etc.).

Parámetro	tipo	nombre	valor
Hasta	1	date_to	Cadena de fecha válida con el formato `YYYY-MM-DD`. Se admiten fechas relativas con los modificadores `d`, `w`, `M`, `y` (por ejemplo, `now`, `now/d`, `now/w-1w`, etc.).

Ejemplos

Los siguientes ejemplos tienen como objetivo describir únicamente la configuración de los objetos de campo del widget de panel para el widget *Informe de SLA*. Para obtener más información sobre la configuración de un panel, consulte `dashboard.create`.

Configuración de un widget *Informe de SLA*

Configure un widget *Informe de SLA* que muestre el informe de SLA para el servicio "2" del SLA "4" para el período de los últimos 30 días.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "Mi panel",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "slareport",
                        "name": "Informe de SLA",
                        "x": 0,
                        "y": 0,
                        "width": 36,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 10,
                                "name": "slaid.0",
                                "value": 4
                            },
                            {
                                "type": 9,
                                "name": "serviceid.0",
                                "value": 2
                            },
                            {
                                "type": 1,
                                "name": "date_from",
                                "value": "now-30d"
                            },
                            {
                                "type": 1,
                                "name": "date_to",
                                "value": "now"
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
```

```
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Véase también

- Campo de widget de panel
- `dashboard.create`
- `dashboard.update`

27 Información del sistema

Descripción

Estos parámetros y los posibles valores de propiedad para los objetos de campo correspondientes del widget del dashboard permiten configurar el widget *Información del sistema* en los métodos `dashboard.create` y `dashboard.update`.

Attention:
Las propiedades `fields` del widget no se validan durante la creación o actualización de un dashboard. Esto permite a los usuarios modificar widgets integrados y crear widgets personalizados, pero también introduce el riesgo de crear o actualizar widgets de forma incorrecta. Para garantizar la creación o actualización correctas del widget *Información del sistema*, consulte el comportamiento de los parámetros descrito en las tablas siguientes.

Parámetros

Los siguientes parámetros son compatibles con el widget *System Information*.

Parameter	type	name	value
Intervalo de actualización	0	rf_rate	0 - Sin actualización; 10 - 10 segundos; 30 - 30 segundos; 60 - 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - *(predeterminado)* 15 minutos.
Mostrar	0	info_type	0 - *(predeterminado)* Estadísticas del sistema; 1 - Nodos de alta disponibilidad.

Parameter	type	name	value
Mostrar detalles de la comprobación de actualización de software	0	show_software_update_check_details	0 - *(predeterminado)* Deshabilitado; 1 - Habilitado. Comportamiento del parámetro: - *compatible* si `AllowSoftwareUpdateCheck` en la configuración del server de Zabbix está habilitado y *Mostrar* está establecido en "System stats"

Ejemplos

Los siguientes ejemplos tienen como objetivo describir únicamente la configuración de los objetos de campo del widget del panel para el widget *Información del sistema*. Para obtener más información sobre la configuración de un panel, consulte `dashboard.create`.

Configuración de un widget *Información del sistema*

Configure un widget *Información del sistema* que muestre estadísticas del sistema con un intervalo de actualización de 10 minutos y la comprobación de actualizaciones de software habilitada.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "Mi panel",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "systeminfo",
                        "name": "Información del sistema",
                        "x": 0,
                        "y": 0,
                        "width": 36,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 0,
                                "name": "rf_rate",
                                "value": 600
                            },
                            {
                                "type": 0,
                                "name": "show_software_update_check_details",
                                "value": 1
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
```

```
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Véase también

- Campo de widget de panel
- `dashboard.create`
- `dashboard.update`

28 Principales hosts

Descripción

Estos parámetros y los posibles valores de propiedad para los objetos de campo del widget del dashboard correspondiente permiten configurar el widget *Top hosts* en los métodos `dashboard.create` y `dashboard.update`.

Attention:
Las propiedades `fields` del widget no se validan durante la creación o actualización de un dashboard. Esto permite a los usuarios modificar widgets integrados y crear widgets personalizados, pero también introduce el riesgo de crear o actualizar widgets de forma incorrecta. Para garantizar la creación o actualización correcta del widget *Top hosts*, consulte el comportamiento de los parámetros descrito en las tablas siguientes.

Parámetros

Los siguientes parámetros son compatibles con el widget *Hosts principales*.

Parámetro	tipo	nombre	valor
Intervalo de actualización	0	rf_rate	0 - Sin actualización; 10 - 10 segundos; 30 - 30 segundos; 60 - *(predeterminado)* 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - 15 minutos.
Grupos de hosts	2	groupids.0	ID del grupo de hosts. Nota: Para configurar varios grupos de hosts, cree un objeto de campo del widget del dashboard para cada grupo de hosts, con un número incrementado en el nombre de la propiedad. Este parámetro no es compatible al configurar el widget en un dashboard de template.

Parámetro	tipo	nombre	valor
Grupos de hosts (widget)	1	groupids._reference	En lugar del ID del grupo de hosts: `ABCDE._hostgroupids` - establezca un widget compatible (con su parámetro *Referencia* establecido en "`ABCDE`") como fuente de datos para los grupos de hosts. Este parámetro no es compatible al configurar el widget en un dashboard de template.
Hosts	3	hostids.0	ID del host. Nota: Para configurar varios hosts, cree un objeto de campo del widget del dashboard para cada host, con un número incrementado en el nombre de la propiedad. Para varios hosts, el parámetro *Grupos de hosts* debe no estar configurado o estar configurado con al menos un grupo de hosts al que pertenezcan los hosts configurados. Este parámetro no es compatible al configurar el widget en un dashboard de template.
Hosts (widget/dashboard)	1	hostids._reference	En lugar del ID del host: `DASHBOARD.hostids` - establezca el selector de hosts del dashboard como fuente de datos para los hosts; `ABCDE._hostids` - establezca un widget compatible (con su parámetro *Referencia* establecido en "`ABCDE`") como fuente de datos para los hosts. Este parámetro no es compatible al configurar el widget en un dashboard de template.
Etiquetas de host			
Tipo de evaluación	0	evaltype	0 - *(predeterminado)* Y/O; 2 - O. Este parámetro no es compatible al configurar el widget en un dashboard de template.
Nombre de etiqueta	1	tags.0.tag	Cualquier valor de cadena. Nota: El número del nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* al configurar *Etiquetas de host* Este parámetro no es compatible al configurar el widget en un dashboard de template.
Operador	0	tags.0.operator	0 - Contiene; 1 - Es igual a; 2 - No contiene; 3 - No es igual a; 4 - Existe; 5 - No existe. Nota: El número del nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* al configurar *Etiquetas de host* Este parámetro no es compatible al configurar el widget en un dashboard de template.

Parámetro	tipo	nombre	valor
Valor de etiqueta	1	tags.0.value	Cualquier valor de cadena. Nota: El número del nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* al configurar *Etiquetas de host* Este parámetro no es compatible al configurar el widget en un dashboard de template.
Mostrar hosts en mantenimiento	0	maintenance	0 - *(predeterminado)* Deshabilitado; 1 - Habilitado.
Columnas (véase abajo)			
Ordenar por	0	column	Valor numérico de la columna entre las columnas configuradas.
Orden	0	order	2 - *(predeterminado)* Primeros N; 3 - Últimos N.
Límite de hosts	0	show_lines	Los valores posibles van de 1 a 1000. Predeterminado: 10. Este parámetro no es compatible al configurar el widget en un dashboard de template.

Columnas

Las columnas tienen parámetros comunes y parámetros adicionales dependiendo de la configuración del parámetro *Datos*.

Note:
Para todos los parámetros relacionados con las columnas, el número en el nombre de la propiedad (por ejemplo, columns.0.name) hace referencia a una columna para la que se configura el parámetro.

Los siguientes parámetros son compatibles con todas las columnas.

Parámetro	tipo	nombre	valor
Nombre	1	columns.0.name	Cualquier valor de cadena. Comportamiento del parámetro: - *obligatorio*
Datos	0	columns.0.data	1 - Valor de la métrica; 2 - Nombre del equipo; 3 - Texto. Comportamiento del parámetro: - *obligatorio*
Color base	1	columns.0.base_color	Código de color hexadecimal (por ejemplo, FF0000). Comportamiento del parámetro: - *obligatorio*

Valor del item

Los siguientes parámetros son compatibles si *Datos* está establecido en «Valor del item».

Note:
El primer número del nombre de la propiedad *Umbrales* (por ejemplo, columnsthresholds.0.color.0) hace referencia a la columna para la que se configuran los umbrales, mientras que el segundo número hace referencia a la posición del umbral en una lista ordenada de forma ascendente. Sin embargo, si los umbrales se configuran en un orden diferente, los valores se ordenarán de forma ascendente después de actualizar la configuración del widget en el frontend de Zabbix (por ejemplo, `"threshold.0.threshold":"5"` → `"threshold.0.threshold":"1"`; `"threshold.1.threshold":"1"` → `"threshold.1.threshold": "5"`).

Parámetro	type	name	value
Item	1	columns.0.item	Nombre de item válido. Al configurar el widget en un dashboard de template, solo deben establecerse los items configurados en el template.
Mostrar el valor del item como	0	columns.0.display_value_as	0 - *(predeterminado)* Numérico; 1 - Texto; 2 - Binario.
Mostrar	0	columns.0.display	1 - *(predeterminado)* Tal cual; 2 - Barra; 3 - Indicadores; 6 - Minigráfico. Comportamiento del parámetro: - *compatible* si *Mostrar el valor del item como* está establecido en «Numérico»
Mín.	1	columns.0.min	Cualquier valor numérico. Comportamiento del parámetro: - *compatible* si *Mostrar el valor del item como* está establecido en «Numérico» y *Mostrar* está establecido en «Barra» o «Indicadores»
Máx.	1	columns.0.max	Cualquier valor numérico. Comportamiento del parámetro: - *compatible* si *Mostrar el valor del item como* está establecido en «Numérico» y *Mostrar* está establecido en «Barra» o «Indicadores»
Posiciones dec-i-males	0	columns.0.decimal_places	Los valores posibles van de 0 a 10. Predeterminado: 2. Comportamiento del parámetro: - *compatible* si *Mostrar el valor del item como* está establecido en «Numérico»
Minigráfico			
Ancho	0	columns.0.sparkline.width	Los valores posibles van de 0 a 10. Predeterminado: 1. Comportamiento del parámetro: - *compatible* si *Mostrar el valor del item como* está establecido en «Numérico» y *Mostrar* está establecido en «Minigráfico»
Relleno	0	columns.0.sparkline.fill	Los valores posibles van de 0 a 10. Predeterminado: 3. Comportamiento del parámetro: - *compatible* si *Mostrar el valor del item como* está establecido en «Numérico» y *Mostrar* está establecido en «Minigráfico»

Parámetro	type	name	value
Color	1	columns.0.sparkline.color	Código de color hexadecimal (por ejemplo, FF0000). Predeterminado: 42A5F5. Comportamiento del parámetro: - *compatible* si *Mostrar el valor del item como* está establecido en «Numérico» y *Mostrar* está establecido en «Minigráfico»
Período de tiempo	1	columns.0.sparkline.time_period	DASHBOARD._timeperiod - establece el selector de período de tiempo del dashboard como fuente de datos; ABCDE._timeperiod - establece un widget compatible (con su parámetro reference igual a ABCDE) como fuente de datos. Predeterminado: DASHBOARD._timeperiod Como alternativa, puede establecer el período de tiempo solo en los parámetros *Desde* y *Hasta*. Comportamiento del parámetro: - *compatible* si *Mostrar el valor del item como* está establecido en «Numérico» y *Mostrar* está establecido en «Minigráfico»
Desde	1	columns.0.sparkline.time_period.from	Cadena de tiempo válida en sintaxis de tiempo absoluta (YYYY-MM-DD hh:mm:ss) o relativa (now, now/d, now/w-1w, etc.). Comportamiento del parámetro: - *compatible* si *Mostrar el valor del item como* está establecido en «Numérico» y *Mostrar* está establecido en «Minigráfico» - *obligatorio* si columns.0.sparkline.time_period.to está establecido
Hasta	1	columns.0.sparkline.time_period.to	Cadena de tiempo válida en sintaxis de tiempo absoluta (YYYY-MM-DD hh:mm:ss) o relativa (now, now/d, now/w-1w, etc.). Comportamiento del parámetro: - *compatible* si *Mostrar el valor del item como* está establecido en «Numérico» y *Mostrar* está establecido en «Minigráfico» - *obligatorio* si columns.0.sparkline.time_period.from está establecido
Datos históricos	0	columns.0.sparkline.history	*(predeterminado)* Automático; 1 - Historial; 2 - Tendencias. Comportamiento del parámetro: - *compatible* si *Mostrar el valor del item como* está establecido en «Numérico» y *Mostrar* está establecido en «Minigráfico»
Umbrales			
Color	1	columnsthresholds.0.color	Código de color hexadecimal (por ejemplo, FF0000). Comportamiento del parámetro: - *compatible* si *Mostrar el valor del item como* está establecido en «Numérico»
Umbral	1	columnsthresholds.0.threshold	Cualquier valor de cadena. Comportamiento del parámetro: - *compatible* si *Mostrar el valor del item como* está establecido en «Numérico»
Resaltados			
Color	1	columns.0.highlights.color	Código de color hexadecimal (por ejemplo, FF0000). Comportamiento del parámetro: - *compatible* si *Mostrar el valor del item como* está establecido en «Texto»

Parámetro	type	name	value
Patrón	1	columns.0.highlights.0.pattern	Cualquier valor de cadena. Comportamiento del parámetro: - *compatible* si *Mostrar el valor del item como* está establecido en «Texto»
Mostrar miniatura	0	columns.0.show_thumbnail	0 - *(predeterminado)* Deshabilitado; 1 - Habilitado. Comportamiento del parámetro: - *compatible* si *Mostrar el valor del item como* está establecido en «Binario»
Función de agregación	0	columns.0.aggregate_function	0 - *(predeterminado)* no utilizada; 1 - mín.; 2 - máx.; 3 - promedio; 4 - recuento; 5 - suma; 6 - primero; 7 - último.
Período de tiempo	1	columns.0.time_period.data_set	DASHBOARD._timeperiod - establece el selector de período de tiempo del dashboard como fuente de datos; ABCDE._timeperiod - establece un widget compatible (con su parámetro reference igual a ABCDE) como fuente de datos. Predeterminado: DASHBOARD._timeperiod Como alternativa, puede establecer el período de tiempo solo en los parámetros *Desde* y *Hasta*. Comportamiento del parámetro: - *compatible* si *Función de agregación* está establecida en «mín.», «máx.», «promedio», «recuento», «suma», «primero» o «último»
Desde	1	columns.0.time_period.from	Cadena de tiempo válida en sintaxis de tiempo absoluta (YYYY-MM-DD hh:mm:ss) o relativa (now, now/d, now/w-1w, etc.). Comportamiento del parámetro: - *compatible* si *Período de tiempo* no está establecido y *Función de agregación* está establecida en «mín.», «máx.», «promedio», «recuento», «suma», «primero» o «último» - *obligatorio* si columns.0.time_period.to está establecido
Hasta	1	columns.0.time_period.to	Cadena de tiempo válida en sintaxis de tiempo absoluta (YYYY-MM-DD hh:mm:ss) o relativa (now, now/d, now/w-1w, etc.). Comportamiento del parámetro: - *compatible* si *Período de tiempo* no está establecido y *Función de agregación* está establecida en «mín.», «máx.», «promedio», «recuento», «suma», «primero» o «último» - *obligatorio* si columns.0.time_period.from está establecido
Datos históricos	0	columns.0.history	0 - *(predeterminado)* Automático; 1 - Historial; 2 - Tendencias. Comportamiento del parámetro: - *compatible* si *Mostrar el valor del item como* está establecido en «Numérico»
Referencia	1	reference	Cualquier valor de cadena compuesto por 5 caracteres (por ejemplo, ABCDE o JBPNL). Este valor debe ser único dentro del dashboard al que pertenece el widget. Comportamiento del parámetro: - *obligatorio*

Texto

Se admiten los siguientes parámetros si *Data* está configurado como "Text".

Parameter	type	name	value
Text	1	columns.0.text	Cualquier valor de cadena, incluidas macros. Macros admitidas: {HOST.*}, {INVENTORY.*}. Comportamiento del parámetro: - *required* si *Data* está configurado como "Text"

Ejemplos

Los siguientes ejemplos tienen como objetivo describir únicamente la configuración de los objetos de campo del widget de panel para el widget *Equipos principales*. Para obtener más información sobre la configuración de un panel, consulte dashboard.create.

Configuración de un widget *Top hosts*

Configure un widget *Top hosts* que muestre los equipos principales por utilización de CPU en el grupo de equipos "4". Además, configure las siguientes columnas personalizadas: "Nombre del equipo", "Utilización de CPU en %", "1m avg", "5m avg", "15m avg", "Procesos".

Petición:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "tophosts",
                        "name": "Top hosts",
                        "x": 0,
                        "y": 0,
                        "width": 36,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 2,
                                "name": "groupids.0",
                                "value": 4
                            },
                            {
                                "type": 1,
                                "name": "columns.0.name",
                                "value": "Host"
                            },
                            {
                                "type": 0,
                                "name": "columns.0.data",
                                "value": 2
                            },
                            {
                                "type": 1,
                                "name": "columns.0.base_color",
                                "value": "FFFFFF"
                            },
                            {
```

```
            "type": 1,
            "name": "columns.1.name",
            "value": "CPU utilization in %"
        },
        {
            "type": 0,
            "name": "columns.1.data",
            "value": 1
        },
        {
            "type": 1,
            "name": "columns.1.base_color",
            "value": "4CAF50"
        },
        {
            "type": 1,
            "name": "columns.1.item",
            "value": "CPU utilization"
        },
        {
            "type": 0,
            "name": "columns.1.display",
            "value": 3
        },
        {
            "type": 1,
            "name": "columns.1.min",
            "value": "0"
        },
        {
            "type": 1,
            "name": "columns.1.max",
            "value": "100"
        },
        {
            "type": 1,
            "name": "columnsthresholds.1.color.0",
            "value": "FFFF00"
        },
        {
            "type": 1,
            "name": "columnsthresholds.1.threshold.0",
            "value": "50"
        },
        {
            "type": 1,
            "name": "columnsthresholds.1.color.1",
            "value": "FF8000"
        },
        {
            "type": 1,
            "name": "columnsthresholds.1.threshold.1",
            "value": "80"
        },
        {
            "type": 1,
            "name": "columnsthresholds.1.color.2",
            "value": "FF4000"
        },
        {
            "type": 1,
            "name": "columnsthresholds.1.threshold.2",
```

```
            "value": "90"
        },
        {
            "type": 1,
            "name": "columns.2.name",
            "value": "1m avg"
        },
        {
            "type": 0,
            "name": "columns.2.data",
            "value": 1
        },
        {
            "type": 1,
            "name": "columns.2.base_color",
            "value": "FFFFFF"
        },
        {
            "type": 1,
            "name": "columns.2.item",
            "value": "Load average (1m avg)"
        },
        {
            "type": 1,
            "name": "columns.3.name",
            "value": "5m avg"
        },
        {
            "type": 0,
            "name": "columns.3.data",
            "value": 1
        },
        {
            "type": 1,
            "name": "columns.3.base_color",
            "value": "FFFFFF"
        },
        {
            "type": 1,
            "name": "columns.3.item",
            "value": "Load average (5m avg)"
        },
        {
            "type": 1,
            "name": "columns.4.name",
            "value": "15m avg"
        },
        {
            "type": 0,
            "name": "columns.4.data",
            "value": 1
        },
        {
            "type": 1,
            "name": "columns.4.base_color",
            "value": "FFFFFF"
        },
        {
            "type": 1,
            "name": "columns.4.item",
            "value": "Load average (15m avg)"
        },
```

```
                            {
                                "type": 1,
                                "name": "columns.5.name",
                                "value": "Processes"
                            },
                            {
                                "type": 0,
                                "name": "columns.5.data",
                                "value": 1
                            },
                            {
                                "type": 1,
                                "name": "columns.5.base_color",
                                "value": "FFFFFF"
                            },
                            {
                                "type": 1,
                                "name": "columns.5.item",
                                "value": "Number of processes"
                            },
                            {
                                "type": 0,
                                "name": "columns.5.decimal_places",
                                "value": 0
                            },
                            {
                                "type": 0,
                                "name": "column",
                                "value": 1
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Véase también

- Campo de widget de panel
- `dashboard.create`
- `dashboard.update`

29 Principales items

Descripción

Estos parámetros y los posibles valores de propiedad para los objetos de campo correspondientes del widget del dashboard permiten configurar el widget *Top items* en los métodos `dashboard.create` y `dashboard.update`.

Attention:
Las propiedades `fields` del widget no se validan durante la creación o actualización de un dashboard. Esto permite a los usuarios modificar widgets integrados y crear widgets personalizados, pero también introduce el riesgo de crear o actualizar widgets de forma incorrecta. Para garantizar la creación o actualización correcta del widget *Top items*, consulte el comportamiento de los parámetros descrito en las tablas siguientes.

Parámetros

Se admiten los siguientes parámetros para el widget *Elementos principales*.

Parámetro	tipo	nombre	valor
Intervalo de actualización	0	rf_rate	0 - Sin actualización; 10 - 10 segundos; 30 - 30 segundos; 60 - *(por defecto)* 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - 15 minutos.
Grupos de equipos	2	groupids.0	ID de Grupo de equipos. Nota: Para configurar varios grupos de equipos, cree un objeto de campo de widget de panel para cada grupo de equipos con un número incrementado en el nombre de la propiedad. Este parámetro no es compatible si se configura el widget en un panel de plantilla.
Grupos de equipos (Widget)	1	groupids._reference	En lugar del ID de Grupo de equipos: `ABCDE._hostgroupids` - establezca un widget compatible (con su parámetro *Referencia* configurado en "ABCDE") como fuente de datos para los grupos de equipos. Este parámetro no es compatible si se configura el widget en un panel de plantilla.
Equipos	3	hostids.0	ID de Equipo. Nota: Para configurar varios equipos, cree un objeto de campo de widget de panel para cada equipo con un número incrementado en el nombre de la propiedad. Para varios equipos, el parámetro *Grupos de equipos* debe no estar configurado en absoluto o configurado con al menos un grupo de equipos al que pertenezcan los equipos configurados. Este parámetro no es compatible si se configura el widget en un panel de plantilla.

Parámetro	tipo	nombre	valor
Equipos (Widget/Panel)	1	hostids._reference	En lugar del ID de Equipo: `DASHBOARD.hostids` - establezca el selector de equipos del panel como fuente de datos para los equipos; `ABCDE._hostids` - establezca un widget compatible (con su parámetro *Referencia* configurado en "`ABCDE`") como fuente de datos para los equipos. Este parámetro no es compatible si se configura el widget en un panel de plantilla.
Etiquetas de equipos			
Tipo de evaluación	0	host_tags_evaltype	0 - *(por defecto)* Y/O; 2 - O. Este parámetro no es compatible si se configura el widget en un panel de plantilla.
Nombre de etiqueta	1	host_tags.0.tag	Cualquier valor de cadena. Nota: El número en el nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* si se configuran *Etiquetas de equipos* Este parámetro no es compatible si se configura el widget en un panel de plantilla.
Operador	0	host_tags.0.operator	0 - Contiene; 1 - Igual a; 2 - No contiene; 3 - No igual a; 4 - Existe; 5 - No existe. Nota: El número en el nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* si se configuran *Etiquetas de equipos* Este parámetro no es compatible si se configura el widget en un panel de plantilla.
Valor de etiqueta	1	host_tags.0.value	Cualquier valor de cadena. Nota: El número en el nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* si se configuran *Etiquetas de equipos* Este parámetro no es compatible si se configura el widget en un panel de plantilla.
Disposición	0	layout	0 - *(por defecto)* Horizontal; 1 - Vertical.
Mostrar problemas	0	show_problems	0 - Todos; 1 - *(por defecto)* No suprimidos; 2 - Ninguno.
Elementos (ver abajo)			

Configuración avanzada

Los siguientes parámetros de configuración avanzada son compatibles con el widget *Top items*.

Parameter	type	name	value
Orden de los host			
Ordenar por	0	host_ordering_order_by	0 - *(predeterminado)* Nombre del host; 3 - Valor del item.
Patrones de item	1	host_ordering_item.0	Nombre o patrón del item (por ejemplo, "*: Number of processed *values per second"). Nota: Para configurar varios patrones de item, cree un objeto de campo del widget del dashboard para cada patrón de item, con un número incrementado en el nombre de la propiedad. Al configurar el widget en un dashboard de template, solo deben establecerse los patrones de los items configurados en el template. Comportamiento del parámetro: - *obligatorio* si *Ordenar por* está establecido en "Valor del item"
Orden	0	host_ordering_order	2 - *(predeterminado)* Primeros N; 3 - Últimos N.
Límite	0	host_ordering_limit	Los valores posibles van de 1 a 100. Predeterminado: 10.
Orden de los items			
Ordenar por	0	item_ordering_order_by	1 - Host; 2 - Nombre del item; 3 - *(predeterminado)* Valor del item.
Patrones de host	1	item_ordering_host.0	Nombre o patrón del host. Nota: Para configurar varios patrones de host, cree un objeto de campo del widget del dashboard para cada patrón de host, con un número incrementado en el nombre de la propiedad. Comportamiento del parámetro: - *obligatorio* si *Ordenar por* está establecido en "Host"
Orden	0	item_ordering_order	2 - *(predeterminado)* Primeros N; 3 - Últimos N.
Límite	0	item_ordering_limit	Los valores posibles van de 1 a 100. Predeterminado: 10.

Columnas

Note:
El primer número del nombre de la propiedad (por ejemplo, columns.0.items.0, columns.0.item_tags_evaltype) representa la columna correspondiente, mientras que el segundo número, si está presente, representa la entidad configurada (por ejemplo, el patrón de item o la etiqueta).

Parámetro	type	name	value
Patrones de item	1	columns.0.items.0	Nombre o patrón de item (por ejemplo, "*: Número de valores * procesados por segundo"). Nota: Para configurar varios patrones de item, cree un objeto de campo de widget de dashboard para cada patrón de item, incrementando el segundo número del nombre de la propiedad. Al configurar el widget en un dashboard de template, solo deben establecerse los patrones de los item configurados en el template. Comportamiento del parámetro: - *obligatorio*
Etiquetas de item			
Tipo de evaluación	0	columns.0.item_tags.0.evaltype	0 - *(predeterminado)* Y/O; 2 - O.
Nombre de la etiqueta	1	columns.0.item_tags.0.tag	Cualquier valor de cadena. Nota: El segundo número del nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* si se configuran *Etiquetas de item*
Operador	0	columns.0.item_tags.0.operator	0 - Contiene; 1 - Es igual a; 2 - No contiene; 3 - No es igual a; 4 - Existe; 5 - No existe. Nota: El segundo número del nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* si se configuran *Etiquetas de item*
Valor de la etiqueta	1	columns.0.item_tags.0.value	Cualquier valor de cadena. Nota: El segundo número del nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* si se configuran *Etiquetas de item*
Color base	1	columns.0.base_color	Código de color hexadecimal (por ejemplo, FF0000).
Mostrar valor como	0	columns.0.display_value_as	1 - *(predeterminado)* Numérico; 2 - Texto.
Mostrar	0	columns.0.display	1 - *(predeterminado)* Tal cual; 2 - Barra; 3 - Indicadores; 6 - Minigráfico.
Mín.	1	columns.0.min	Cualquier valor numérico. Se admiten sufijos (por ejemplo, "1d", "2w", "4K", "8G"). Comportamiento del parámetro: - *compatible* si *Mostrar valor como* está establecido en "Numérico" y *Mostrar* está establecido en "Barra" o "Indicadores"

Parámetro	type	name	value
Máx.	1	columns.0.max	Cualquier valor numérico. Se admiten sufijos (por ejemplo, ”1d”, ”2w”, ”4K”, ”8G”). Comportamiento del parámetro: - *compatible* si *Mostrar valor como* está establecido en ”Numérico” y *Mostrar* está establecido en ”Barra” o ”Indicadores”
Minigráfico			
Ancho	0	columns.0.sparkline.width	Los valores posibles van de 0 a 10. Predeterminado: 1. Comportamiento del parámetro: - *compatible* si *Mostrar* está establecido en ”Minigráfico”
Relleno	0	columns.0.sparkline.fill	Los valores posibles van de 0 a 10. Predeterminado: 3. Comportamiento del parámetro: - *compatible* si *Mostrar* está establecido en ”Minigráfico”
Color	1	columns.0.sparkline.color	Código de color hexadecimal (por ejemplo, `FF0000`). Predeterminado: `42A5F5`. Comportamiento del parámetro: - *compatible* si *Mostrar* está establecido en ”Minigráfico”
Período de tiempo	1	columns.0.sparkline.time_period.from	`DASHBOARD._timeperiod` - establece el selector de período de tiempo del dashboard como fuente de datos; `ABCDE._timeperiod` - establece un widget compatible (con su parámetro `reference` igual a `ABCDE`) como fuente de datos. Predeterminado: `DASHBOARD._timeperiod` Como alternativa, puede establecer el período de tiempo únicamente en los parámetros *Desde* y *Hasta*. Comportamiento del parámetro: - *compatible* si *Mostrar* está establecido en ”Minigráfico”
Desde	1	columns.0.sparkline.time_period.from	Cadena de tiempo válida en sintaxis absoluta (`YYYY-MM-DD hh:mm:ss`) o relativa (`now`, `now/d`, `now/w-1w`, etc.). Comportamiento del parámetro: - *compatible* si *Mostrar* está establecido en ”Minigráfico” - *obligatorio* si `columns.0.sparkline.time_period.to` está establecido
Hasta	1	columns.0.sparkline.time_period.to	Cadena de tiempo válida en sintaxis absoluta (`YYYY-MM-DD hh:mm:ss`) o relativa (`now`, `now/d`, `now/w-1w`, etc.). Comportamiento del parámetro: - *compatible* si *Mostrar* está establecido en ”Minigráfico” - *obligatorio* si `columns.0.sparkline.time_period.from` está establecido
Datos históricos	0	columns.0.sparkline.history	*(predeterminado)* Automático; 1 - Historial; 2 - Tendencias. Comportamiento del parámetro: - *compatible* si *Mostrar* está establecido en ”Minigráfico”
Umbrales			
Color	1	columns.0.thresholds.color	Código de color hexadecimal (por ejemplo, `FF0000`). Comportamiento del parámetro: - *compatible* si *Mostrar valor como* está establecido en ”Numérico”

Parámetro	type	name	value
Umbral	1	columns.0.thresholds.0.threshold	Cualquier valor numérico. Se admiten sufijos (por ejemplo, ”1d”, ”2w”, ”4K”, ”8G”). Comportamiento del parámetro: - *compatible* si *Mostrar valor como* está establecido en ”Numérico”
Resaltados			
Resaltado	1	columns.0.highlights.0.color	Código de color hexadecimal (por ejemplo, `FF0000`). Comportamiento del parámetro: - *compatible* si *Mostrar valor como* está establecido en ”Texto”
Umbral	1	columns.0.highlights.0.pattern	Cualquier expresión regular. Comportamiento del parámetro: - *compatible* si *Mostrar valor como* está establecido en ”Texto”
Posiciones decimales	0	columns.0.decimal_places	Los valores posibles van de 0 a 10. Predeterminado: 2.
Configuración avanzada			
Función de agregación	0	columns.0.aggregate_function	Si *Mostrar valor como* está establecido en ”Numérico”: 0 - *(predeterminado)* no se utiliza; 1 - mín.; 2 - máx.; 3 - promedio; 4 - recuento; 5 - suma; 6 - primero; 7 - último. Si *Mostrar valor como* está establecido en ”Texto”: 0 - *(predeterminado)* no se utiliza; 4 - recuento; 6 - primero; 7 - último.
Período de tiempo	1	columns.0.time_period.from	`DASHBOARD._timeperiod` - establece el selector de período de tiempo del dashboard como fuente de datos; `ABCDE._timeperiod` - establece un widget compatible (con su parámetro `reference` igual a `ABCDE`) como fuente de datos. Predeterminado: `DASHBOARD._timeperiod` Como alternativa, puede establecer el período de tiempo únicamente en los parámetros *Desde* y *Hasta*. Comportamiento del parámetro: - *compatible* si *Función de agregación* está establecida en ”mín.”, ”máx.”, ”promedio”, ”recuento”, ”suma”, ”primero” o ”último”
Desde	1	columns.0.time_period.from	Cadena de tiempo válida en sintaxis absoluta (`YYYY-MM-DD hh:mm:ss`) o relativa (`now`, `now/d`, `now/w-1w`, etc.). Comportamiento del parámetro: - *compatible* si *Período de tiempo* no está establecido y *Función de agregación* está establecida en ”mín.”, ”máx.”, ”promedio”, ”recuento”, ”suma”, ”primero” o ”último” - *obligatorio* si `columns.0.time_period.to` está establecido

Parámetro	type	name	value
Hasta	1	columns.0.time_period.to	Cadena de tiempo válida en sintaxis absoluta (`YYYY-MM-DD hh:mm:ss`) o relativa (`now`, `now/d`, `now/w-1w`, etc.). Comportamiento del parámetro: - *compatible* si *Período de tiempo* no está establecido y *Función de agregación* está establecida en ”mín.”, ”máx.”, ”promedio”, ”recuento”, ”suma”, ”primero” o ”último” - *obligatorio* si `columns.0.time_period.from` está establecido
Datos históricos	0	columns.0.history	0 - *(predeterminado)* Automático; 1 - Historial; 2 - Tendencias. Comportamiento del parámetro: - *compatible* si *Mostrar valor como* está establecido en ”Numérico”
Agregado	0	columns.0.aggregate_columns	0 - *(predeterminado)* Cada item 1 - Combinado. Comportamiento del parámetro: - *compatible* si *Mostrar valor como* está establecido en ”Numérico” y *Mostrar* está establecido en ”Tal cual”, ”Barra” o ”Indicadores”.
Función de agregación combinada	0	columns.0.column_aggregate_function	1 - mín.; 2 - máx.; 3 - promedio; 4 - recuento; 5 - suma *(predeterminado)*. Comportamiento del parámetro: - *obligatorio* si se configura *Agregado*.
Nombre de columna combinada	1	columns.0.combined_column_name	Cualquier valor de cadena. Comportamiento del parámetro: - *obligatorio* si se configura *Agregado*.

Ejemplos

Los siguientes ejemplos tienen como objetivo describir únicamente la configuración de los objetos de campo del widget de panel para el widget *Principales métricas*. Para obtener más información sobre la configuración de un panel, consulte `dashboard.create`.

Configuración de un widget *Principales métricas*

Configure un widget *Principales métricas* que muestre datos para el equipo ”10084” y solo para las métricas para las cuales la etiqueta con el nombre ”component” contiene el valor ”cpu”. Además, muestre los datos con los equipos ubicados en la parte superior y utilice una barra de indicador de color para la representación de la celda.

Petición:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "topitems",
                        "name": "Top items",
                        "x": 0,
```

```
                        "y": 0,
                        "width": 36,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 3,
                                "name": "hostids.0",
                                "value": 10084
                            },
                            {
                                "type": 1,
                                "name": "columns.0.items.0",
                                "value": "*"
                            },
                            {
                                "type": 1,
                                "name": "columns.0.item_tags.0.tag",
                                "value": "component"
                            },
                            {
                                "type": 0,
                                "name": "columns.0.item_tags.0.operator",
                                "value": 0
                            },
                            {
                                "type": 1,
                                "name": "columns.0.item_tags.0.value",
                                "value": "cpu"
                            },
                            {
                                "type": 0,
                                "name": "columns.0.display",
                                "value": 2
                            },
                            {
                                "type": 0,
                                "name": "layout",
                                "value": 1
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```


Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Véase también

- Campo de widget de panel
- `dashboard.create`
- `dashboard.update`

30 Principales triggers

Descripción

Estos parámetros y los posibles valores de propiedad para los objetos de campo del widget del dashboard correspondiente permiten configurar el widget *Top triggers* en los métodos `dashboard.create` y `dashboard.update`.

Attention:
Las propiedades `fields` del widget no se validan durante la creación o actualización de un dashboard. Esto permite a los usuarios modificar widgets integrados y crear widgets personalizados, pero también introduce el riesgo de crear o actualizar widgets de forma incorrecta. Para garantizar la creación o actualización correctas del widget *Top triggers*, consulte el comportamiento de los parámetros descrito en las tablas siguientes.

Parámetros

Los siguientes parámetros son compatibles con el widget *Top triggers*.

Parameter	type	name	value
Intervalo de actualización	0	rf_rate	0 - *(predeterminado)* Sin actualización; 10 - 10 segundos; 30 - 30 segundos; 60 - 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - 15 minutos.
Grupos de hosts	2	groupids.0	ID del grupo de hosts. Nota: Para configurar varios grupos de hosts, cree un objeto de campo del widget del dashboard para cada grupo de hosts, con un número incrementado en el nombre de la propiedad. Este parámetro no es compatible al configurar el widget en un dashboard de template.
Hosts	3	hostids.0	ID del host. Nota: Para configurar varios hosts, cree un objeto de campo del widget del dashboard para cada host, con un número incrementado en el nombre de la propiedad. Para varios hosts, el parámetro *Grupos de hosts* debe no estar configurado o estar configurado con al menos un grupo de hosts al que pertenezcan los hosts configurados. Este parámetro no es compatible al configurar el widget en un dashboard de template.
Problema	1	problem	Nombre del evento del problema (no distingue entre mayúsculas y minúsculas, nombre completo o parte del nombre).

Parameter	type	name	value
Gravedad	0	severities.0	Gravedades de trigger. 0 - No clasificado; 1 - Información; 2 - Advertencia; 3 - Promedio; 4 - Alto; 5 - Desastre. Predeterminado: vacío (todas habilitadas). Nota: Para configurar varios valores, cree un objeto de campo del widget del dashboard para cada valor, con un número incrementado en el nombre de la propiedad.
Etiquetas del problema			
Tipo de evaluación	0	evaltype	0 - *(predeterminado)* Y/O; 2 - O.
Nombre de la etiqueta	1	tags.0.tag	Cualquier valor de cadena. Nota: El número del nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* al configurar *Etiquetas del problema*
Operador	0	tags.0.operator	0 - Contiene; 1 - Es igual a; 2 - No contiene; 3 - No es igual a; 4 - Existe; 5 - No existe. Nota: El número del nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* al configurar *Etiquetas del problema*
Valor de la etiqueta	1	tags.0.value	Cualquier valor de cadena. Nota: El número del nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* al configurar *Etiquetas del problema*
Período de tiempo	1	time_period._referenc	`DASHBOARD._timeperiod` - establece el selector de período de tiempo del dashboard como fuente de datos; `ABCDE._timeperiod` - establece un widget compatible (con su parámetro *Referencia* establecido en "`ABCDE`") como fuente de datos. Predeterminado: `DASHBOARD._timeperiod` Como alternativa, puede establecer el período de tiempo únicamente en los parámetros *Desde* y *Hasta*.
Desde	1	time_period.from	Cadena de tiempo válida en sintaxis absoluta (`YYYY-MM-DD hh:mm:ss`) o relativa (`now`, `now/d`, `now/w-1w`, etc.). Comportamiento del parámetro: - *compatible* si no se establece *Período de tiempo* - *obligatorio* si se establece `time_period.to`

Parameter	type	name	value
Hasta	1	time_period.to	Cadena de tiempo válida en sintaxis absoluta (`YYYY-MM-DD hh:mm:ss`) o relativa (`now`, `now/d`, `now/w-1w`, etc.). Comportamiento del parámetro: - *compatible* si no se establece *Período de tiempo* - *obligatorio* si se establece `time_period.from`
Límite de trig-gers	0	show_lines	Los valores posibles van de 1 a 100. Predeterminado: 10.

Ejemplos

Los siguientes ejemplos tienen como objetivo describir únicamente la configuración de los objetos de campo del widget de panel para el widget *Principales disparadores*. Para obtener más información sobre la configuración de un panel, consulte `dashboard.create`.

Configuración de un widget *Top triggers*

Configure un widget *Top triggers* que muestre los 5 triggers principales para el grupo de hosts "4" con el recuento de todos los problemas para cada trigger. El widget muestra solo los triggers que tienen las severidades "Warning", "Average", "High" o "Disaster", y los problemas que tienen una etiqueta con el nombre "scope" que contiene los valores "performance" o "availability", o "capacity".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "toptriggers",
                        "name": "Top triggers",
                        "x": 0,
                        "y": 0,
                        "width": 36,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 2,
                                "name": "groupids.0",
                                "value": 4
                            },
                            {
                                "type": 0,
                                "name": "severities.0",
                                "value": 2
                            },
                            {
                                "type": 0,
                                "name": "severities.1",
                                "value": 3
                            },
                            {
                                "type": 0,
                                "name": "severities.2",
```

```
                        "value": 4
                    },
                    {
                        "type": 0,
                        "name": "severities.3",
                        "value": 5
                    },
                    {
                        "type": 1,
                        "name": "tags.0.tag",
                        "value": "scope"
                    },
                    {
                        "type": 0,
                        "name": "tags.0.operator",
                        "value": 0
                    },
                    {
                        "type": 1,
                        "name": "tags.0.value",
                        "value": "performance"
                    },
                    {
                        "type": 1,
                        "name": "tags.1.tag",
                        "value": "scope"
                    },
                    {
                        "type": 0,
                        "name": "tags.1.operator",
                        "value": 0
                    },
                    {
                        "type": 1,
                        "name": "tags.1.value",
                        "value": "availability"
                    },
                    {
                        "type": 1,
                        "name": "tags.2.tag",
                        "value": "scope"
                    },
                    {
                        "type": 0,
                        "name": "tags.2.operator",
                        "value": 0
                    },
                    {
                        "type": 1,
                        "name": "tags.2.value",
                        "value": "capacity"
                    },
                    {
                        "type": 0,
                        "name": "show_lines",
                        "value": 5
                    }
                ]
            }
        ]
    }
],
```

```
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Véase también

- Campo de widget de panel
- `dashboard.create`
- `dashboard.update`

31 Resumen de triggers

Descripción

Estos parámetros y los posibles valores de propiedad para los objetos de campo del widget del dashboard correspondiente permiten configurar el widget *Resumen de triggers* en los métodos `dashboard.create` y `dashboard.update`.

Attention:
Las propiedades `fields` del widget no se validan durante la creación o actualización de un dashboard. Esto permite a los usuarios modificar widgets integrados y crear widgets personalizados, pero también introduce el riesgo de crear o actualizar widgets de forma incorrecta. Para garantizar la creación o actualización correctas del widget *Resumen de triggers*, consulte el comportamiento de los parámetros descrito en las tablas siguientes.

Parámetros

Se admiten los siguientes parámetros para el widget *Resumen de Disparadores*.

Parámetro	tipo	nombre	valor
Intervalo de actualización	0	rf_rate	0 - Sin actualización; 10 - 10 segundos; 30 - 30 segundos; 60 - *(por defecto)* 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - 15 minutos.
Mostrar	0	show	1 - *(por defecto)* Problemas recientes; 2 - Cualquiera; 3 - Problemas.

Parámetro	tipo	nombre	valor
Grupos de equipos	2	groupids.0	ID de Grupo de equipos. Nota: Para configurar varios grupos de equipos, cree un objeto de campo de widget de panel para cada grupo de equipos con un número incrementado en el nombre de la propiedad. Este parámetro no es compatible si se configura el widget en un panel de plantilla.
Grupos de equipos (Widget)	1	groupids._reference	En lugar del ID de Grupo de equipos: `ABCDE._hostgroupids` - establezca un widget compatible (con su parámetro *Referencia* configurado en "`ABCDE`") como fuente de datos para los grupos de equipos. Este parámetro no es compatible si se configura el widget en un panel de plantilla.
Equipos	3	hostids.0	ID de Equipo. Nota: Para configurar varios equipos, cree un objeto de campo de widget de panel para cada equipo con un número incrementado en el nombre de la propiedad. Para varios equipos, el parámetro *Grupos de equipos* debe no estar configurado en absoluto o configurado con al menos un grupo de equipos al que pertenezcan los equipos configurados. Este parámetro no es compatible si se configura el widget en un panel de plantilla.
Equipos (Widget/Panel)	1	hostids._reference	En lugar del ID de Equipo: `DASHBOARD.hostids` - establezca el selector de equipos del panel como fuente de datos para los equipos; `ABCDE._hostids` - establezca un widget compatible (con su parámetro *Referencia* configurado en "`ABCDE`") como fuente de datos para los equipos. Este parámetro no es compatible si se configura el widget en un panel de plantilla.
Etiquetas de problema			
Tipo de evaluación	0	evaltype	0 - *(por defecto)* Y/O; 2 - O.
Nombre de etiqueta	1	tags.0.tag	Cualquier valor de cadena. Nota: El número en el nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* si se configuran *Etiquetas de problema*
Operador	0	tags.0.operator	0 - Contiene; 1 - Igual a; 2 - No contiene; 3 - No igual a; 4 - Existe; 5 - No existe. Nota: El número en el nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* si se configuran *Etiquetas de problema*

Parámetro	tipo	nombre	valor
Valor de etiqueta	1	tags.0.value	Cualquier valor de cadena. Nota: El número en el nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* si se configuran *Etiquetas de problema*
Mostrar problemas suprimidos	0	show_suppressed	0 - *(por defecto)* Deshabilitado; 1 - Habilitado.
Disposición	0	layout	0 - *(por defecto)* Horizontal; 1 - Vertical.

Ejemplos

Los siguientes ejemplos tienen como objetivo describir únicamente la configuración de los objetos de campo del widget de panel para el widget *Resumen de disparadores*. Para obtener más información sobre la configuración de un panel, consulte `dashboard.create`.

Configuración de un widget *Resumen de disparadores*

Configure un widget *Resumen de disparadores* que muestre los estados de los disparadores para todos los grupos de equipos que tengan disparadores con una etiqueta cuyo nombre sea ”scope” y contenga el valor ”availability”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "trigover",
                        "name": "Trigger overview",
                        "x": 0,
                        "y": 0,
                        "width": 36,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 1,
                                "name": "tags.0.tag",
                                "value": "scope"
                            },
                            {
                                "type": 0,
                                "name": "tags.0.operator",
                                "value": 0
                            },
                            {
                                "type": 1,
                                "name": "tags.0.value",
                                "value": "availability"
```

```
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Véase también

- Campo de widget de panel
- `dashboard.create`
- `dashboard.update`

32 URL

Descripción

Estos parámetros y los posibles valores de propiedad para los objetos de campo correspondientes del widget del dashboard permiten configurar el widget *URL* en los métodos `dashboard.create` y `dashboard.update`.

Attention:
Las propiedades `fields` del widget no se validan durante la creación o actualización de un dashboard. Esto permite a los usuarios modificar widgets integrados y crear widgets personalizados, pero también introduce el riesgo de crear o actualizar widgets de forma incorrecta. Para garantizar la creación o actualización correctas del widget *URL*, consulte el comportamiento de los parámetros descrito en las tablas siguientes.

Parámetros

Se admiten los siguientes parámetros para el widget *URL*.

Parámetro	tipo	nombre	valor
Intervalo de actualización	0	rf_rate	0 - *(por defecto)* Sin actualización; 10 - 10 segundos; 30 - 30 segundos; 60 - 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - 15 minutos.
URL	1	url	Cadena de URL válida. Comportamiento del parámetro: - *obligatorio*
Anular host	1	override_hostid._reference	`ABCDE._hostid` - establecer un widget compatible (con su parámetro *Referencia* configurado en "ABCDE") como fuente de datos para los hosts; `DASHBOARD._hostid` - establecer el selector de host del panel como fuente de datos para los hosts. Este parámetro no es compatible si se configura el widget en un panel de plantilla.

Ejemplos

Los siguientes ejemplos tienen como objetivo describir únicamente la configuración de los objetos de campo del widget de panel para el widget *URL*. Para obtener más información sobre la configuración de un panel, consulte `dashboard.create`.

Configuración de un widget *URL*

Configure un widget *URL* que muestre la página principal del manual de Zabbix.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "Mi panel",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "url",
                        "name": "URL",
                        "x": 0,
                        "y": 0,
                        "width": 36,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 1,
                                "name": "url",
                                "value": "https://www.zabbix.com/documentation/8.0/en"
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
```

```
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Véase también

- Campo de widget de panel
- `dashboard.create`
- `dashboard.update`

33 Monitorización web

Descripción

Estos parámetros y los posibles valores de propiedad para los objetos de campo del widget del dashboard correspondiente permiten configurar el widget *Supervisión web* en los métodos `dashboard.create` y `dashboard.update`.

Attention:
Las propiedades `fields` del widget no se validan durante la creación o actualización de un dashboard. Esto permite a los usuarios modificar widgets integrados y crear widgets personalizados, pero también introduce el riesgo de crear o actualizar widgets de forma incorrecta. Para garantizar la creación o actualización correcta del widget *Supervisión web*, consulte el comportamiento de los parámetros descrito en las tablas siguientes.

Parámetros

Se admiten los siguientes parámetros para el widget de *Supervisión web*.

Parámetro	tipo	nombre	valor
Intervalo de actualización	0	rf_rate	0 - Sin actualización; 10 - 10 segundos; 30 - 30 segundos; 60 - *(por defecto)* 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - 15 minutos.

Parámetro	tipo	nombre	valor
Grupos de equipos	2	groupids.0	ID de Grupo de equipos. Nota: Para configurar varios grupos de equipos, cree un objeto de campo de widget de panel para cada grupo de equipos con un número incrementado en el nombre de la propiedad. Este parámetro no es compatible si se configura el widget en un panel de plantilla.
Grupos de equipos (Widget)	1	groupids._reference	En lugar del ID de Grupo de equipos: `ABCDE._hostgroupids` - establezca un widget compatible (con su parámetro *Referencia* configurado en "`ABCDE`") como fuente de datos para los grupos de equipos. Este parámetro no es compatible si se configura el widget en un panel de plantilla.
Excluir grupos de equipos	2	exclude_groupids.0	ID de Grupo de equipos. Nota: Para excluir varios grupos de equipos, cree un objeto de campo de widget de panel para cada grupo de equipos con un número incrementado en el nombre de la propiedad. Este parámetro no es compatible si se configura el widget en un panel de plantilla.
Equipos	3	hostids.0	ID de Equipo. Nota: Para configurar varios equipos, cree un objeto de campo de widget de panel para cada equipo con un número incrementado en el nombre de la propiedad. Para varios equipos, el parámetro *Grupos de equipos* debe no estar configurado en absoluto o configurado con al menos un grupo de equipos al que pertenezcan los equipos configurados. Este parámetro no es compatible si se configura el widget en un panel de plantilla.
Equipos (Widget/Panel)	1	hostids._reference	En lugar del ID de Equipo: `DASHBOARD.hostids` - establezca el selector de equipos del panel como fuente de datos para los equipos; `ABCDE._hostids` - establezca un widget compatible (con su parámetro *Referencia* configurado en "`ABCDE`") como fuente de datos para los equipos. Este parámetro no es compatible si se configura el widget en un panel de plantilla.
Etiquetas de escenario			
Tipo de evaluación	0	evaltype	0 - *(por defecto)* Y/O; 2 - O.
Nombre de la etiqueta	1	tags.0.tag	Cualquier valor de cadena. Nota: El número en el nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* si se configuran *Etiquetas de escenario*

Parámetro	tipo	nombre	valor
Operador	0	tags.0.operator	0 - Contiene; 1 - Igual a; 2 - No contiene; 3 - No igual a; 4 - Existe; 5 - No existe. Nota: El número en el nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* si se configuran *Etiquetas de escenario*
Valor de la etiqueta	1	tags.0.value	Cualquier valor de cadena. Nota: El número en el nombre de la propiedad hace referencia al orden de la etiqueta en la lista de evaluación de etiquetas. Comportamiento del parámetro: - *obligatorio* si se configuran *Etiquetas de escenario*
Mostrar equipos en mantenimiento	0	maintenance	0 - Deshabilitado; 1 - *(por defecto)* Habilitado.
Referencia	1	reference	Cualquier valor de cadena que consista en 5 caracteres (por ejemplo, `ABCDE` o `JBPNL`). Este valor debe ser único dentro del panel al que pertenece el widget. Comportamiento del parámetro: - *obligatorio*

Ejemplos

Los siguientes ejemplos tienen como objetivo describir únicamente la configuración de los objetos de campo del widget de panel para el widget *Supervisión web*. Para obtener más información sobre la configuración de un panel, consulte `dashboard.create`.

Configuración de un widget *Supervisión web*

Configure un widget *Supervisión web* que muestre un resumen del estado de los escenarios activos de supervisión web para el grupo de equipos "4".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "web",
                        "name": "Supervisión web",
                        "x": 0,
                        "y": 0,
                        "width": 18,
                        "height": 3,
                        "view_mode": 0,
                        "fields": [
```

```
                            {
                                "type": 2,
                                "name": "groupids.0",
                                "value": 4
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Véase también

- Campo de widget de panel
- `dashboard.create`
- `dashboard.update`

Directorio de usuarios

Esta clase está diseñada para trabajar con directorios de usuarios.

Referencias de objetos:

- Directorio de usuarios
  - Mapeos de tipos de medios
  - Mapeos de grupos de aprovisionamiento

Métodos disponibles:

- userdirectory.create - crear un nuevo directorio de usuarios
- userdirectory.delete - eliminar un directorio de usuarios
- userdirectory.get - recuperar un directorio de usuarios
- userdirectory.update - actualizar un directorio de usuarios
- userdirectory.test - probar la conexión del directorio de usuarios

Objeto de directorio de usuario

Los siguientes objetos están directamente relacionados con la API `userdirectory`.

Directorio de usuarios

El objeto de directorio de usuarios tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
userdirectoryid	ID	ID del directorio de usuarios. Si se elimina un directorio de usuarios, el valor de la propiedad `userdirectoryid` del objeto User se establece en ”0” para todos los usuarios vinculados al directorio de usuarios eliminado. Comportamiento de la propiedad: - *solo lectura* - *obligatoria* para las operaciones de actualización
idp_type	integer	Tipo de protocolo de autenticación utilizado por el proveedor de identidad para el directorio de usuarios. Tenga en cuenta que solo puede existir un directorio de usuarios de tipo SAML. Valores posibles: 1 - Directorio de usuarios de tipo LDAP; 2 - Directorio de usuarios de tipo SAML. Comportamiento de la propiedad: - *obligatoria* para las operaciones de creación
group_name	string	Atributo del directorio de usuarios LDAP/SAML que contiene el nombre del grupo utilizado para asignar grupos entre el directorio de usuarios LDAP/SAML y Zabbix. Ejemplo: *cn* Comportamiento de la propiedad: - *obligatoria* si `provision_status` está establecido en ”Enabled” y `saml_jit_status` del objeto Authentication está establecido en ”Enabled for configured SAML IdPs”
user_username	string	Atributo del directorio de usuarios LDAP/SAML (también atributo SCIM si `scim_status` está establecido en ”SCIM provisioning is enabled”) que contiene el nombre del usuario, utilizado como valor de la propiedad `name` del objeto User cuando se aprovisiona el usuario. Ejemplos: *cn*, *commonName*, *displayName*, *name*
user_lastname	string	Atributo del directorio de usuarios LDAP/SAML (también atributo SCIM si `scim_status` está establecido en ”SCIM provisioning is enabled”) que contiene el apellido del usuario, utilizado como valor de la propiedad `surname` del objeto User cuando se aprovisiona el usuario. Ejemplos: *sn*, *surname*, *lastName*
provision_status	integer	Estado de aprovisionamiento del directorio de usuarios. Valores posibles: 0 - *(predeterminado)* Deshabilitado (el aprovisionamiento de usuarios creados por este directorio de usuarios está deshabilitado); 1 - Habilitado (el aprovisionamiento de usuarios creados por este directorio de usuarios está habilitado; además, el estado del aprovisionamiento LDAP o SAML (`ldap_jit_status` o `saml_jit_status` del objeto Authentication) debe estar habilitado).
provision_groups	array	Matriz de objetos de asignaciones de grupos de aprovisionamiento para asignar el patrón de grupo de usuarios LDAP/SAML al grupo de usuarios y al rol de usuario de Zabbix. Comportamiento de la propiedad: - *obligatoria* si `provision_status` está establecido en ”Enabled”

Propiedad	Tipo	Descripción
provision_media	array	Matriz de objetos de asignaciones de tipos de medios para asignar los atributos de medios LDAP/SAML del usuario (por ejemplo, el correo electrónico) a los medios de usuario de Zabbix para enviar notificaciones.
Propiedades específicas de LDAP:		
name	string	Nombre único del directorio de usuarios. Comportamiento de la propiedad: - *obligatoria* si `idp_type` está establecido en "User directory of type LDAP"
host	string	Nombre de host, IP o URI del server LDAP. La URI debe contener el esquema (`ldap://` o `ldaps://`), el host y el puerto (opcional). Ejemplos: *host.example.com* *127.0.0.1* *ldap://ldap.example.com:389* Comportamiento de la propiedad: - *obligatoria* si `idp_type` está establecido en "User directory of type LDAP"
port	integer	Puerto del server LDAP. Comportamiento de la propiedad: - *obligatoria* si `idp_type` está establecido en "User directory of type LDAP"
base_dn	string	Ruta base del directorio de usuarios LDAP a las cuentas de usuario. Ejemplos: *ou=Users,dc=example,dc=org* *ou=Users,ou=system* (para OpenLDAP) *DC=company,DC=com* (para Microsoft Active Directory) *uid=%{user},dc=example,dc=com* (para la vinculación directa del usuario; el marcador de posición "*%{user}*" es obligatorio) Comportamiento de la propiedad: - *obligatoria* si `idp_type` está establecido en "User directory of type LDAP"
search_attribute	string	Atributo del directorio de usuarios LDAP mediante el cual se identifica la cuenta de usuario a partir de la información proporcionada en la solicitud de inicio de sesión. Ejemplos: *uid* (para OpenLDAP) *sAMAccountName* (para Microsoft Active Directory) Comportamiento de la propiedad: - *obligatoria* si `idp_type` está establecido en "User directory of type LDAP"

Propiedad	Tipo	Descripción
bind_dn	string	Cuenta del server LDAP para vincularse y realizar búsquedas en el server LDAP. Para la vinculación directa del usuario y la vinculación anónima, `bind_dn` debe estar vacío. Ejemplos: *uid=ldap_search,ou=system* (para OpenLDAP) *CN=ldap_search,OU=user_group,DC=company,DC=com* (para Microsoft Active Directory) *CN=Admin,OU=Users,OU=Zabbix,DC=zbx,DC=local* Comportamiento de la propiedad: - *compatible* si `idp_type` está establecido en ”User directory of type LDAP”
bind_password	string	Contraseña LDAP de la cuenta para vincularse y realizar búsquedas en el server LDAP. Para la vinculación directa del usuario y la vinculación anónima, `bind_password` debe estar vacía. Comportamiento de la propiedad: - *compatible* si `idp_type` está establecido en ”User directory of type LDAP”
description	string	Descripción del directorio de usuarios. Comportamiento de la propiedad: - *compatible* si `idp_type` está establecido en ”User directory of type LDAP”
group_basedn	string	Ruta base del directorio de usuarios LDAP a los grupos; se utiliza para configurar una comprobación de pertenencia de usuarios en el directorio de usuarios LDAP. Se ignora al aprovisionar un usuario si se establece `group_membership`. Ejemplo: *ou=Groups,dc=example,dc=com* Comportamiento de la propiedad: - *compatible* si `idp_type` está establecido en ”User directory of type LDAP”

Propiedad	Tipo	Descripción
group_filter	string	Cadena de filtro para recuperar los grupos del directorio de usuarios LDAP a los que pertenece el usuario; se utiliza para configurar una comprobación de pertenencia de usuarios en el directorio de usuarios LDAP. Se ignora al aprovisionar un usuario si se establece `group_membership`. Marcadores de posición `group_filter` compatibles: *%{attr}* - atributo de búsqueda (se reemplaza por el valor de la propiedad `search_attribute`); *%{groupattr}* - atributo de grupo (se reemplaza por el valor de la propiedad `group_member`); *%{host}* - nombre de host, IP o URI del server LDAP (se reemplaza por el valor de la propiedad `host`); *%{user}* - nombre de usuario de Zabbix. Predeterminado: *(%{groupattr}=%{user})* Ejemplos: - *(member=uid=%{ref},ou=Users,dc=example,dc=com)* coincidirá con ”User1” si un objeto de grupo LDAP contiene el atributo ”*member*” con el valor ”*uid=User1,ou=Users,dc=example,dc=com*” y devolverá el grupo al que pertenece ”User1”; - *(%{groupattr}=cn=%{ref},ou=Users,ou=Zabbix,DC=example,DC=com)* coincidirá con ”User1” si un objeto de grupo LDAP contiene el atributo especificado en la propiedad `group_member` con el valor ”*cn=User1,ou=Users,ou=Zabbix,DC=example,DC=com*” y devolverá el grupo al que pertenece ”User1”. Comportamiento de la propiedad: - *compatible* si `idp_type` está establecido en ”User directory of type LDAP”
group_member	string	Atributo del directorio de usuarios LDAP que contiene información sobre los miembros del grupo; se utiliza para configurar una comprobación de pertenencia de usuarios en el directorio de usuarios LDAP. Se ignora al aprovisionar un usuario si se establece `group_membership`. Comportamiento de la propiedad: - *compatible* si `idp_type` está establecido en ”User directory of type LDAP”
group_membership	string	Atributo del directorio de usuarios LDAP que contiene información sobre los grupos a los que pertenece un usuario. Ejemplo: *memberOf* Comportamiento de la propiedad: - *compatible* si `idp_type` está establecido en ”User directory of type LDAP”

Propiedad	Tipo	Descripción
search_filter	string	Cadena de filtro personalizada utilizada para localizar y autenticar a un usuario en un directorio de usuarios LDAP según la información proporcionada en la solicitud de inicio de sesión. Marcadores de posición `search_filter` compatibles: *%{attr}* - nombre del atributo de búsqueda (por ejemplo, *uid*, *sAMAccountName*); *%{user}* - nombre de usuario de Zabbix. Predeterminado: *(%{attr}=%{user})* Comportamiento de la propiedad: - *compatible* si `idp_type` está establecido en ”User directory of type LDAP”
start_tls	integer	Opción de configuración del server LDAP que permite proteger la comunicación con el server LDAP mediante Transport Layer Security (TLS). Tenga en cuenta que `start_tls` debe estar establecido en ”Disabled” para los hosts que utilizan el protocolo `ldaps://`. Valores posibles: 0 - *(predeterminado)* Deshabilitado; 1 - Habilitado. Comportamiento de la propiedad: - *compatible* si `idp_type` está establecido en ”User directory of type LDAP”
user_ref_attr	string	Atributo del directorio de usuarios LDAP utilizado para hacer referencia a un objeto de usuario. El valor de `user_ref_attr` se utiliza para obtener valores del atributo especificado en el directorio de usuarios y colocarlos en lugar del marcador de posición *%{ref}* en la cadena `group_filter`. Ejemplos: *cn*, *uid*, *member*, *uniqueMember* Comportamiento de la propiedad: - *compatible* si `idp_type` está establecido en ”User directory of type LDAP”
Propiedades específicas de SAML:		
idp_entityid	string	URI que identifica al proveedor de identidad y se utiliza para comunicarse con él en los mensajes SAML. Ejemplo: *https://idp.example.com/idp* Comportamiento de la propiedad: - *obligatoria* si `idp_type` está establecido en ”User directory of type SAML”
sp_entityid	string	URL o cualquier cadena que identifique al proveedor de servicios del proveedor de identidad. Ejemplos: *https://idp.example.com/sp* *zabbix* Comportamiento de la propiedad: - *obligatoria* si `idp_type` está establecido en ”User directory of type SAML”

Propiedad	Tipo	Descripción
username_attribute	string	Atributo del directorio de usuarios SAML (también atributo SCIM si `scim_status` está establecido en "SCIM provisioning is enabled") que contiene el nombre de usuario, comparado con el valor de la propiedad `username` del objeto User durante la autenticación. Ejemplos: *uid*, *userprincipalname*, *samaccountname*, *username*, *userusername*, *urn:oid:0.9.2342.19200300.100.1.1*, *urn:oid:1.3.6.1.4.1.5923.1.1.1.13*, *urn:oid:0.9.2342.19200300.100.1.44* Comportamiento de la propiedad: - *obligatoria* si `idp_type` está establecido en "User directory of type SAML"
sso_url	string	URL del servicio de inicio de sesión único SAML del proveedor de identidad, al que Zabbix enviará las solicitudes de autenticación SAML. Ejemplo: *http://idp.example.com/idp/sso/saml* Comportamiento de la propiedad: - *obligatoria* si `idp_type` está establecido en "User directory of type SAML"
slo_url	string	URL del servicio de cierre de sesión único SAML del proveedor de identidad, al que Zabbix enviará las solicitudes de cierre de sesión SAML. Ejemplo: *https://idp.example.com/idp/slo/saml* Comportamiento de la propiedad: - *compatible* si `idp_type` está establecido en "User directory of type SAML"
encrypt_nameid	integer	Indica si se debe cifrar el ID de nombre SAML. Valores posibles: 0 - *(predeterminado)* No cifrar el ID de nombre; 1 - Cifrar el ID de nombre. Comportamiento de la propiedad: - *compatible* si `idp_type` está establecido en "User directory of type SAML"
encrypt_assertions	integer	Indica si se deben cifrar las aserciones SAML. Valores posibles: 0 - *(predeterminado)* No cifrar las aserciones; 1 - Cifrar las aserciones. Comportamiento de la propiedad: - *compatible* si `idp_type` está establecido en "User directory of type SAML"
nameid_format	string	Formato del ID de nombre del proveedor de servicios del proveedor de identidad SAML. Ejemplos: *urn:oasis:names:tc:SAML:2.0:nameid-format:persistent* *urn:oasis:names:tc:SAML:2.0:nameid-format:transient* *urn:oasis:names:tc:SAML:2.0:nameid-format:kerberos* *urn:oasis:names:tc:SAML:2.0:nameid-format:entity* Comportamiento de la propiedad: - *compatible* si `idp_type` está establecido en "User directory of type SAML"

Propiedad	Tipo	Descripción
scim_status	integer	Indica si el aprovisionamiento SCIM para SAML está habilitado o deshabilitado. Valores posibles: 0 - *(predeterminado)* El aprovisionamiento SCIM está deshabilitado; 1 - El aprovisionamiento SCIM está habilitado. Comportamiento de la propiedad: - *compatible* si `idp_type` está establecido en ”User directory of type SAML”
sign_assertions	integer	Indica si las aserciones SAML deben firmarse con una firma SAML. Valores posibles: 0 - No firmar las aserciones; 1 - *(predeterminado)* Firmar las aserciones. Comportamiento de la propiedad: - *compatible* si `idp_type` está establecido en ”User directory of type SAML”
sign_authn_requests	integer	Indica si las solicitudes AuthN SAML deben firmarse con una firma SAML. Valores posibles: 0 - No firmar las solicitudes AuthN; 1 - *(predeterminado)* Firmar las solicitudes AuthN. Comportamiento de la propiedad: - *compatible* si `idp_type` está establecido en ”User directory of type SAML”
sign_messages	integer	Indica si los mensajes SAML deben firmarse con una firma SAML. Valores posibles: 0 - No firmar los mensajes; 1 - *(predeterminado)* Firmar los mensajes. Comportamiento de la propiedad: - *compatible* si `idp_type` está establecido en ”User directory of type SAML”
sign_logout_requests	integer	Indica si las solicitudes de cierre de sesión SAML deben firmarse con una firma SAML. Valores posibles: 0 - No firmar las solicitudes de cierre de sesión; 1 - *(predeterminado)* Firmar las solicitudes de cierre de sesión. Comportamiento de la propiedad: - *compatible* si `idp_type` está establecido en ”User directory of type SAML”
sign_logout_responses	integer	Indica si las respuestas de cierre de sesión SAML deben firmarse con una firma SAML. Valores posibles: 0 - No firmar las respuestas de cierre de sesión; 1 - *(predeterminado)* Firmar las respuestas de cierre de sesión. Comportamiento de la propiedad: - *compatible* si `idp_type` está establecido en ”User directory of type SAML”

Propiedad	Tipo	Descripción
idp_certificate	string	Contenido del certificado del proveedor de servicios (SP) para configurar el servicio de inicio de sesión único (SSO) SAML. Ejemplos: *-----BEGIN CERTIFICATE----- MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEA7... ...more encoded data... -----END CERTIFICATE-----* Comportamiento de la propiedad: - *compatible* si `idp_type` está establecido en ”User directory of type SAML” y `$SSO['CERT_STORAGE']` está establecido en `database` en el archivo de configuración del frontend - *solo escritura*
sp_private_key	string	Contenido de la clave privada del proveedor de servicios (SP) para configurar el servicio de inicio de sesión único (SSO) SAML. Proporciona autenticación segura e intercambio de datos con el proveedor de identidad (IdP). Ejemplos: *-----BEGIN CERTIFICATE----- MIIEvQIBADANBgkqhkiG9w0BA... ...more encoded data... -----END CERTIFICATE-----* Comportamiento de la propiedad: - *compatible* si `idp_type` está establecido en ”User directory of type SAML” y `$SSO['CERT_STORAGE']` está establecido en `database` en el archivo de configuración del frontend - *solo escritura*
sp_certificate	string	Contenido del certificado del proveedor de servicios (SP) para configurar el servicio de inicio de sesión único (SSO) SAML. Ejemplos: *-----BEGIN CERTIFICATE----- MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEA7... ...more encoded data... -----END CERTIFICATE-----* Comportamiento de la propiedad: - *compatible* si `idp_type` está establecido en ”User directory of type SAML” y `$SSO['CERT_STORAGE']` está establecido en `database` en el archivo de configuración del frontend - *solo escritura*
idp_certificate_hash	string	El hash md5 del valor de idp_certificate. Devuelve una cadena vacía si idp_certificate está vacío. Comportamiento de la propiedad: - *compatible* si `idp_type` está establecido en ”User directory of type SAML” y `$SSO['CERT_STORAGE']` está establecido en `database` en el archivo de configuración del frontend - *solo lectura*
sp_private_key_hash	string	El hash md5 del valor de sp_private_key. Devuelve una cadena vacía si sp_private_key está vacío. Comportamiento de la propiedad: - *compatible* si `idp_type` está establecido en ”User directory of type SAML” y `$SSO['CERT_STORAGE']` está establecido en `database` en el archivo de configuración del frontend - *solo lectura*

Propiedad	Tipo	Descripción
sp_certificate_hash	string	El hash md5 del valor de sp_certificate. Devuelve una cadena vacía si sp_certificate está vacío. Comportamiento de la propiedad: - *compatible* si `idp_type` está establecido en "User directory of type SAML" y `$SSO['CERT_STORAGE']` está establecido en `database` en el archivo de configuración del frontend - *solo lectura*

Asignaciones de tipos de medios

El objeto de asignaciones de tipos de medios tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
userdirectory_mediaid	ID	ID de la asignación de tipo de medio. Comportamiento de la propiedad: - *solo lectura*
name	string	Nombre visible en la lista de asignaciones de tipos de medios. Comportamiento de la propiedad: - *requerido*
mediatypeid	ID	ID del tipo de medio que se va a crear; se utiliza como valor para la propiedad `mediatypeid` del objeto Media. Comportamiento de la propiedad: - *requerido*
attribute	string	Atributo del directorio de usuarios LDAP/SAML (también atributo SCIM si `scim_status` está configurado como "El aprovisionamiento SCIM está habilitado") que contiene el medio del usuario (por ejemplo, *user@example.com*) que se utiliza como valor para la propiedad `sendto` del objeto Media. Si está presente en los datos recibidos del proveedor de identidad LDAP/SAML, y el valor no está vacío, esto activará la creación del medio para el usuario aprovisionado. Comportamiento de la propiedad: - *requerido*
active	integer	Valor de la propiedad `active` del medio del usuario cuando se crea el medio para el usuario aprovisionado. Valores posibles: 0 - *(por defecto)* habilitado; 1 - deshabilitado.
severity	integer	Valor de la propiedad `severity` del medio del usuario cuando se crea el medio para el usuario aprovisionado. Por defecto: 63.
period	string	Valor de la propiedad `period` del medio del usuario cuando se crea el medio para el usuario aprovisionado. Por defecto: 1-7,00:00-24:00.

Asignación de grupos de aprovisionamiento

La asignación de grupos de aprovisionamiento tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
name	string	Nombre completo de un grupo (por ejemplo, *Zabbix administrators*) en el directorio de usuarios LDAP/SAML (también SCIM si `scim_status` está establecido en "El aprovisionamiento SCIM está habilitado"). Admite el carácter comodín "*". Único en todas las asignaciones de grupos de aprovisionamiento. Comportamiento de la propiedad: - *obligatorio*
roleid	ID	ID del rol de usuario que se asignará al usuario. Si se corresponden varias asignaciones de grupos de aprovisionamiento, se asigna al usuario el rol del tipo de usuario más alto (*User*, *Admin* o *Super admin*). Si hay varios roles con el mismo tipo de usuario, se asigna al usuario el primer rol (ordenado alfabéticamente). Comportamiento de la propiedad: - *obligatorio*
user_groups	array	Matriz de objetos de ID de grupo de usuarios de Zabbix. Cada objeto tiene las siguientes propiedades: `usrgrpid` - `(ID)` ID del grupo de usuarios de Zabbix que se asignará al usuario. Si se corresponden varias asignaciones de grupos de aprovisionamiento, se asignan al usuario los grupos de usuarios de Zabbix de todas las asignaciones coincidentes. Comportamiento de la propiedad: - *obligatorio*

userdirectory.create

Descripción

`object userdirectory.create(object/array userDirectory)`

Este método permite crear nuevos directorios de usuarios.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*.

Parámetros

`(objeto/array)` Directorios de usuario a crear.

El método acepta directorios de usuario con las propiedades estándar de directorio de usuario.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los directorios de usuario creados bajo la propiedad `userdirectoryids`. El orden de los IDs devueltos coincide con el orden de los directorios de usuario pasados.

Ejemplos

Creando un directorio de usuarios

Cree un directorio de usuarios para autenticar usuarios con StartTLS sobre LDAP. Tenga en cuenta que para autenticar usuarios a través de LDAP, la autenticación LDAP debe estar habilitada.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "userdirectory.create",
    "params": {
        "idp_type": "1",
        "name": "LDAP API server #1",
        "host": "ldap://local.ldap",
        "port": "389",
```

```
        "base_dn": "ou=Users,dc=example,dc=org",
        "bind_dn": "cn=ldap_search,dc=example,dc=org",
        "bind_password": "ldapsecretpassword",
        "search_attribute": "uid",
        "start_tls": "1"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userdirectoryids": [
            "3"
        ]
    },
    "id": 1
}
```

Creando un directorio de usuarios (provisión JIT habilitada)

Cree un directorio de usuarios para autenticar usuarios mediante LDAP (con provisión JIT habilitada). Tenga en cuenta que para autenticar usuarios mediante LDAP, la autenticación LDAP debe estar habilitada.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "userdirectory.create",
    "params": {
            "idp_type": "1",
            "name": "AD server",
            "provision_status": "1",
            "description": "",
            "host": "host.example.com",
            "port": "389",
            "base_dn": "DC=zbx,DC=local",
            "search_attribute": "sAMAccountName",
            "bind_dn": "CN=Admin,OU=Users,OU=Zabbix,DC=zbx,DC=local",
            "start_tls": "0",
            "search_filter": "",
            "group_basedn": "OU=Zabbix,DC=zbx,DC=local",
            "group_name": "CN",
            "group_member": "member",
            "group_filter": "(%{groupattr}=CN=%{ref},OU=Users,OU=Zabbix,DC=zbx,DC=local)",
            "group_membership": "",
            "user_username": "givenName",
            "user_lastname": "sn",
            "user_ref_attr": "CN",
            "provision_media": [
                {
                    "name": "example.com",
                    "mediatypeid": "1",
                    "attribute": "user@example.com"
                }
            ],
            "provision_groups": [
                {
                    "name": "*",
                    "roleid": "4",
                    "user_groups": [
                        {
                            "usrgrpid": "8"
```

```
                        }
                    ]
                },
                {
                    "name": "Zabbix administrators",
                    "roleid": "2",
                    "user_groups": [
                        {
                            "usrgrpid": "7"
                        },
                        {
                            "usrgrpid": "8"
                        }
                    ]
                }
            ]
        },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userdirectoryids": [
            "2"
        ]
    },
    "id": 1
}
```

Fuente

CUserDirectory::create() en *ui/include/classes/api/services/CUserDirectory.php*.

userdirectory.delete

Descripción

`object userdirectory.delete(array userDirectoryIds)`

Este método le permite eliminar directorios de usuarios. Un directorio de usuarios no se puede eliminar cuando se usa directamente para al menos un grupo de usuarios.
 El directorio de usuarios LDAP predeterminado no se puede eliminar cuando `authentication.ldap_configured` está establecido en 1 o cuando quedan más directorios de usuarios.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*.

Parámetros

`(array)` IDs de los directorios de usuario a eliminar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los directorios de usuario eliminados bajo la propiedad `userdirectoryids`.

Ejemplos

Eliminando múltiples directorios de usuario

Eliminar dos directorios de usuario.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "userdirectory.delete",
```

```
    "params": [
        "2",
        "12"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userdirectoryids": [
            "2",
            "12"
        ]
    },
    "id": 1
}
```

Fuente

CUserDirectory::delete() en *ui/include/classes/api/services/CUserDirectory.php*.

userdirectory.get

Descripción

`integer/array userdirectory.get(object parameters)`

El método permite recuperar directorios de usuarios según los parámetros proporcionados.

Note:
Este método solo está disponible para tipos de usuario *Super admin*.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
userdirectoryids	ID/array	Devolver solo los directorios de usuario con los IDs dados.
selectUsrgrps	query	Devuelve una propiedad `usrgrps` con los grupos de usuarios asociados a un directorio de usuario. Admite `count`.
selectProvisionMedia	query	Devuelve una propiedad `provision_media` con las asignaciones de tipos de medios asociadas a un directorio de usuario.
selectProvisionGroups	query	Devuelve una propiedad `provision_groups` con las asignaciones de grupos de aprovisionamiento asociadas a un directorio de usuario.
sortfield	string/array	Ordena el resultado por las propiedades dadas. Valores posibles: `name`.
filter	object	Devuelve solo aquellos resultados que coincidan exactamente con el filtro dado. Acepta un objeto, donde las claves son nombres de propiedades y los valores son un único valor o un array de valores. Admite las propiedades: `userdirectoryid`, `idp_type`, `provision_status`.

Parámetro	Tipo	Descripción
search	object	Devuelve resultados que coincidan con el patrón dado (no distingue mayúsculas/minúsculas). Acepta un objeto, donde las claves son nombres de propiedades y los valores son cadenas a buscar. Si no se dan opciones adicionales, esto realizará una búsqueda `LIKE "%…%"`. Propiedades admitidas: `name`, `description`. El directorio de usuario de tipo SAML tendrá un valor vacío tanto para las propiedades `name` como `description`. Ambas propiedades pueden cambiarse con la operación `userdirectory.update`.
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
excludeSearch	boolean	
limit	integer	
output	query	
preservekeys	boolean	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

`(entero/array)` Devuelve:

- un array de objetos
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`

Ejemplos

Recuperar directorios de usuarios

Recupera todos los directorios de usuarios con propiedades adicionales que muestran los mapeos de tipos de medios y los mapeos de grupos de aprovisionamiento asociados a cada directorio de usuarios.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "userdirectory.get",
    "params": {
        "output": "extend",
        "selectProvisionMedia": "extend",
        "selectProvisionGroups": "extend"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
       {
            "userdirectoryid": "1",
            "idp_type": "2",
            "name": "",
            "provision_status": "1",
            "description": "",
            "group_name": "groups",
            "user_username": "",
            "user_lastname": "",
            "idp_entityid": "http://example.com/simplesaml/saml2/idp/metadata.php",
            "sso_url": "http://example.com/simplesaml/saml2/idp/SSOService.php",
```

```
            "slo_url": "",
            "username_attribute": "uid",
            "sp_entityid": "zabbix",
            "nameid_format": "",
            "sign_messages": "0",
            "sign_assertions": "0",
            "sign_authn_requests": "0",
            "sign_logout_requests": "0",
            "sign_logout_responses": "0",
            "encrypt_nameid": "0",
            "encrypt_assertions": "0",
            "scim_status": "1",
            "provision_media": [
                {
                    "userdirectory_mediaid": "1",
                    "name": "example.com",
                    "mediatypeid": "1",
                    "attribute": "user@example.com",
                    "active": "0",
                    "severity": "63",
                    "period": "1-7,00:00-24:00"
                }
            ],
            "provision_groups": [
                {
                    "name": "*",
                    "roleid": "1",
                    "user_groups": [
                        {
                            "usrgrpid": "13"
                        }
                    ]
                }
            ]
        },
        {
            "userdirectoryid": "2",
            "idp_type": "1",
            "name": "AD server",
            "provision_status": "1",
            "description": "",
            "host": "host.example.com",
            "port": "389",
            "base_dn": "DC=zbx,DC=local",
            "search_attribute": "sAMAccountName",
            "bind_dn": "CN=Admin,OU=Users,OU=Zabbix,DC=zbx,DC=local",
            "start_tls": "0",
            "search_filter": "",
            "group_basedn": "OU=Zabbix,DC=zbx,DC=local",
            "group_name": "CN",
            "group_member": "member",
            "group_filter": "(%{groupattr}=CN=%{ref},OU=Users,OU=Zabbix,DC=zbx,DC=local)",
            "group_membership": "",
            "user_username": "givenName",
            "user_lastname": "sn",
            "user_ref_attr": "CN",
            "provision_media": [
                {
                    "userdirectory_mediaid": "2",
                    "name": "example.com",
                    "mediatypeid": "1",
                    "attribute": "user@example.com",
```

```
                    "active": "0",
                    "severity": "63",
                    "period": "1-7,00:00-24:00"
                }
            ],
            "provision_groups": [
                {
                    "name": "*",
                    "roleid": "4",
                    "user_groups": [
                        {
                            "usrgrpid": "8"
                        }
                    ]
                },
                {
                    "name": "Zabbix administrators",
                    "roleid": "2",
                    "user_groups": [
                        {
                            "usrgrpid": "7"
                        },
                        {
                            "usrgrpid": "8"
                        }
                    ]
                }
            ]
        },
        {
            "userdirectoryid": "3",
            "idp_type": "1",
            "name": "LDAP API server #1",
            "provision_status": "0",
            "description": "",
            "host": "ldap://local.ldap",
            "port": "389",
            "base_dn": "ou=Users,dc=example,dc=org",
            "search_attribute": "uid",
            "bind_dn": "cn=ldap_search,dc=example,dc=org",
            "start_tls": "1",
            "search_filter": "",
            "group_basedn": "",
            "group_name": "",
            "group_member": "",
            "group_filter": "",
            "group_membership": "",
            "user_username": "",
            "user_lastname": "",
            "user_ref_attr": "",
            "provision_media": [],
            "provision_groups": []
        }
    ],
    "id": 1
}
```


Véase también

- Grupo de usuarios

Fuente

CUserDirectory::get() en *ui/include/classes/api/services/CUserDirectory.php*.

userdirectory.test

Descripción

`object userdirectory.test(array userDirectory)`

Este método le permite probar la configuración de conexión del directorio de usuarios.

> **Note:**
> Este método también le permite probar qué datos configurados coinciden con la configuración del directorio de usuarios para el aprovisionamiento de usuarios (por ejemplo, qué rol de usuario, grupos de usuarios y medios de usuario se asignarán al usuario). Para este tipo de prueba, la solicitud de la API debe realizarse para un user directory que tenga `provision_status` establecido en habilitado.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*.

Parámetros

`(object)` Propiedades del directorio de usuario.

Dado que la API `userdirectory.get` no devuelve el campo `bind_password`, se debe proporcionar `userdirectoryid` y/o `bind_password`.
 Además de las propiedades estándar del directorio de usuario, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
test_username	string	Nombre de usuario a probar en el directorio de usuario.
test_password	string	Contraseña asociada al nombre de usuario a probar en el directorio de usuario.

Valores de retorno

`(bool)` Devuelve verdadero en caso de éxito.

Ejemplos

Probar el directorio de usuarios para un usuario existente

Probar el directorio de usuarios "3" para "user1".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "userdirectory.test",
    "params": {
        "userdirectoryid": "3",
        "host": "127.0.0.1",
        "port": "389",
        "base_dn": "ou=Users,dc=example,dc=org",
        "search_attribute": "uid",
        "bind_dn": "cn=ldap_search,dc=example,dc=org",
        "bind_password": "password",
        "test_username": "user1",
        "test_password": "password"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": true,
    "id": 1
}
```

Probar el directorio de usuarios para un usuario inexistente

Probar el directorio de usuarios ”3” para el ”user2” inexistente.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "userdirectory.test",
    "params": {
        "userdirectoryid": "3",
        "host": "127.0.0.1",
        "port": "389",
        "base_dn": "ou=Users,dc=example,dc=org",
        "search_attribute": "uid",
        "bind_dn": "cn=ldap_search,dc=example,dc=org",
        "test_username": "user2",
        "test_password": "password"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "error": {
        "code": -32500,
        "message": "Application error.",
        "data": "Incorrect user name or password or account is temporarily blocked."
    },
    "id": 1
}
```

Probar el directorio de usuarios para el aprovisionamiento de usuarios

Probar el directorio de usuarios ”3” para ver qué datos configurados coinciden con la configuración del directorio de usuarios para el aprovisionamiento de ”user3” (por ejemplo, qué rol de usuario, grupos de usuarios, medios de usuario se asignarán al usuario).

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "userdirectory.test",
    "params": {
        "userdirectoryid": "2",
        "host": "host.example.com",
        "port": "389",
        "base_dn": "DC=zbx,DC=local",
        "search_attribute": "sAMAccountName",
        "bind_dn": "CN=Admin,OU=Users,OU=Zabbix,DC=zbx,DC=local",
        "test_username": "user3",
        "test_password": "password"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "username": "user3",
        "name": "John",
        "surname": "Doe",
        "medias": [],
        "usrgrps": [
            {
```

```
                "usrgrpid": "8"
            },
            {
                "usrgrpid": "7"
            }
        ],
        "roleid": "2",
        "userdirectoryid": "2"
    },
    "id": 1
}
```

Origen

CUserDirectory::test() en *ui/include/classes/api/services/CUserDirectory.php*.

userdirectory.update

Descripción

`object userdirectory.update(object/array userDirectory)`

Este método le permite actualizar directorios de usuarios existentes.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*.

Parámetros

`(objeto/array)` Propiedades del directorio de usuarios que se van a actualizar.

La propiedad `userdirectoryid` debe definirse para cada directorio de usuarios, todas las demás propiedades son opcionales. Solo se actualizarán las propiedades indicadas, todas las demás permanecerán sin cambios.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los directorios de usuario actualizados bajo la propiedad `userdirectoryids`.

Ejemplos

Actualizar la contraseña de enlace para el directorio de usuarios

Establecer una nueva contraseña de enlace para un directorio de usuarios.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "userdirectory.update",
    "params": {
        "userdirectoryid": "3",
        "bind_password": "newldappassword"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userdirectoryids": [
            "3"
        ]
    },
    "id": 1
}
```

Actualizar asignaciones para el directorio de usuarios

Actualizar las asignaciones de grupos de aprovisionamiento y las asignaciones de tipos de medios para el directorio de usuarios "2".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "userdirectory.update",
    "params": {
        "userdirectoryid": "2",
        "provision_media": [
                {
                    "userdirectory_mediaid": "2"
                }
            ],
            "provision_groups": [
                {
                    "name": "Zabbix administrators",
                    "roleid": "2",
                    "user_groups": [
                        {
                            "usrgrpid": "7"
                        },
                        {
                            "usrgrpid": "8"
                        },
                        {
                            "usrgrpid": "11"
                        }
                    ]
                }
            ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userdirectoryids": [
            "2"
        ]
    },
    "id": 1
}
```

Fuente

CUserDirectory::update() en *ui/include/classes/api/services/CUserDirectory.php*.

Dispositivo

Esta clase está diseñada para trabajar con dispositivos móviles para su uso con la aplicación móvil de Zabbix.

Referencias de objetos:

- Device

Métodos disponibles:

- device.get: recuperar dispositivos
- device.offboard: eliminar dispositivos (retirarlos)

Objeto de dispositivo

El objeto Device representa un dispositivo móvil para utilizarlo con la aplicación móvil de Zabbix.

Property	Type	Description
deviceid	ID	ID del dispositivo (interno, por ejemplo, 1, 2). Comportamiento de la propiedad: - Solo lectura
userid	ID	ID del usuario del dispositivo. Comportamiento de la propiedad: - Constante
uuid	string	Identificador único universal del dispositivo (UUIDv7) (externo, por ejemplo, `019f930e-0ea2-7c1f-9dcc-0fbb60c40664`). Comportamiento de la propiedad: - Solo lectura
name	string	Nombre del dispositivo. Comportamiento de la propiedad: - Solo lectura
status	integer	Estado del dispositivo. Valores posibles: 0 - Nuevo 1 - Activo 2 - Sin asignar Comportamiento de la propiedad: - Solo lectura
activated_at	timestamp	Fecha y hora en que se añadió el dispositivo en la aplicación móvil, o 0 si aún no se ha añadido. Comportamiento de la propiedad: - Solo lectura
lastaccess	timestamp	Fecha y hora en que el dispositivo se autenticó por última vez en la aplicación móvil, o 0 si aún no se ha autenticado. Comportamiento de la propiedad: - Solo lectura

device.get

Este método permite recuperar dispositivos móviles para utilizarlos con la aplicación móvil de Zabbix.

El método está disponible para usuarios *User*, *Admin* y *Super admin* si `EnableMobileDevices` está habilitado y el usuario tiene acceso a los dispositivos. El acceso a los métodos de la API se puede revocar en la configuración del rol de usuario.

Los usuarios *User* y *Admin* solo pueden recuperar sus propios dispositivos.

Firma del método: `integer/array device.get(object parameters)`

Origen del método: `CDevice::get()` en `ui/include/classes/api/services/CDevice.php`

Parámetros

Parámetro	Tipo	Descripción
deviceids	ID/array	Devuelve solo los dispositivos con los ID de dispositivo indicados.
userids	ID/array	Devuelve solo los dispositivos pertenecientes a usuarios con los ID de usuario indicados.
roleids	ID/array	Devuelve solo los dispositivos pertenecientes a usuarios con los ID de rol de usuario indicados.
usrgrpids	ID/array	Devuelve solo los dispositivos pertenecientes a usuarios con los ID de grupo de usuarios indicados.

Parámetro	Tipo	Descripción
sortfield	string/array	Ordena el resultado según las propiedades indicadas. Valores posibles: `deviceid`, `name`.
countOutput	boolean	Estos parámetros se describen en los comentarios de referencia.
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores devueltos

Devuelve un array de objetos o el número de objetos recuperados (si se utiliza `countOutput`).

Ejemplos

Recuperar dispositivos

Recupere los dispositivos pertenecientes a los usuarios con estos ID de usuario: 1, 2, 3.
 Tenga en cuenta que el usuario 2 (invitado) no tiene dispositivos y que el dispositivo 3 aún no se ha añadido ni autenticado en la aplicación móvil de Zabbix.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "device.get",
    "params": {
        "output": "extend",
        "userids": [1, 2, 3]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "deviceid": "1",
            "userid": "1",
            "uuid": "019f930e-0ea2-7c1f-9dcc-0fbb60c40664",
            "name": "Google Pixel 10",
            "status": "1",
            "activated_at": "1784533122",
            "lastaccess": "1784879933"
        },
        {
            "deviceid": "2",
            "userid": "1",
            "uuid": "019d00fb-9e87-7ad5-bb4e-978467254aff",
            "name": "Apple iPhone 15 Pro",
            "status": "1",
            "activated_at": "1784533225",
            "lastaccess": "0"
        },
        {
            "deviceid": "3",
            "userid": "3",
```

```
            "uuid": "019d00fb-9e87-7ad5-bb4e-6ed88a2297ee",
            "name": "",
            "status": "0",
            "activated_at": "0",
            "lastaccess": "0"
        }
    ],
    "id": 1
}
```

device.offboard

Este método permite eliminar dispositivos móviles para dejar de utilizarlos con la aplicación móvil de Zabbix.

El método está disponible para usuarios *User*, *Admin* y *Super admin* si `EnableMobileDevices` está habilitado y el usuario tiene acceso a los dispositivos. El acceso a los métodos de la API se puede revocar en la configuración del rol de usuario.

Los usuarios *User* y *Admin* solo pueden eliminar sus propios dispositivos.

Firma del método: `object device.offboard(object parameters)`

Origen del método: `CDevice::offboard()` en `ui/include/classes/api/services/CDevice.php`

Parámetros

Parameter	Type	Description
uuid	string	Identificador único universal del dispositivo (UUIDv7). Coincide con `deviceid`.
force	boolean	Establezca `true` para ignorar el error "Device not found" y eliminar el dispositivo de forma forzada. Valor predeterminado: `false`

Valores devueltos

Devuelve un objeto que contiene el UUID del dispositivo eliminado.

Ejemplos

Eliminar dispositivo

Elimina un dispositivo con el UUID indicado, ignorando el error "Dispositivo no encontrado".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "device.offboard",
    "params": {
        "uuid": "019d0106-5eef-7154-a159-bb29e35d38e5",
        "force": true
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "uuid": "019d0106-5eef-7154-a159-bb29e35d38e5"
    },
    "id": 1
}
```

Escenario web

Esta clase está diseñada para trabajar con escenarios web.

Referencias de objetos:

- Escenario web
- Paso de escenario
  - Campo HTTP
- Etiqueta de escenario web

Métodos disponibles:

- httptest.create - crear nuevos escenarios web
- httptest.delete - eliminar escenarios web
- httptest.get - recuperar escenarios web
- httptest.update - actualizar escenarios web

Objeto de escenario web

Los siguientes objetos están directamente relacionados con la API `webcheck`.

Escenario web

El objeto de escenario web tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
httptestid	ID	ID del escenario web. Comportamiento de la propiedad: - *solo lectura* - *obligatoria* para las operaciones de actualización
hostid	ID	ID del host al que pertenece el escenario web. Comportamiento de la propiedad: - *constante* - *obligatoria* para las operaciones de creación
name	string	Nombre del escenario web. Comportamiento de la propiedad: - *obligatoria* para las operaciones de creación
agent	string	Cadena de user agent que utilizará el escenario web. Valor predeterminado: Zabbix
authentication	integer	Método de autenticación que utilizará el escenario web. Valores posibles: 0 - *(predeterminado)* ninguna; 1 - autenticación HTTP básica; 2 - autenticación NTLM; 3 - autenticación Kerberos; 4 - autenticación Digest.
delay	string	Intervalo de ejecución del escenario web. Acepta un número de segundos (por ejemplo, 1, 60, 300), un valor con un sufijo de tiempo (por ejemplo, 30s, 1m, 2h, 1d) o una macro de usuario. Valor predeterminado: 1m.
headers	array	Encabezados HTTP que se enviarán al realizar una solicitud.
http_password	string	Contraseña utilizada para la autenticación HTTP básica, NTLM, Kerberos o Digest.
http_proxy	string	proxy que utilizará el escenario web, indicado como *http://[username[:password]@]proxy.example.com[:port]*.

Propiedad	Tipo	Descripción
http_user	string	Nombre de usuario utilizado para la autenticación HTTP básica, NTLM, Kerberos o Digest.
retries	integer	Número de veces que un escenario web intentará ejecutar cada paso antes de producir un error. Valor predeterminado: 1.
ssl_cert_file	string	Nombre del archivo de certificado SSL utilizado para la autenticación del cliente (debe estar en formato PEM).
ssl_key_file	string	Nombre del archivo de clave privada SSL utilizado para la autenticación del cliente (debe estar en formato PEM).
ssl_key_password	string	Contraseña de la clave privada SSL.
status	integer	Indica si el escenario web está habilitado. Valores posibles: 0 - *(predeterminado)* habilitado; 1 - deshabilitado.
templateid	ID	ID del escenario web del template principal. Comportamiento de la propiedad: - *solo lectura*
variables	array	Variables del escenario web.
verify_host	integer	Indica si se debe validar que el nombre del host de la conexión coincida con el del certificado del host. Valores posibles: 0 - *(predeterminado)* omitir la verificación del host; 1 - verificar el host.
verify_peer	integer	Indica si se debe validar que el certificado del host sea auténtico. Valores posibles: 0 - *(predeterminado)* omitir la verificación del par; 1 - verificar el par.
uuid	string	Identificador único global, utilizado para vincular escenarios web importados con otros ya existentes. Se genera automáticamente si no se proporciona. Comportamiento de la propiedad: - *compatible* si el escenario web pertenece a un template

Paso del escenario

El objeto de paso del escenario define una comprobación específica del escenario web.
Tiene las siguientes propiedades.

Property	Type	Description
name	string	Nombre del paso del escenario. Comportamiento de la propiedad: - *required*
no	integer	Número de secuencia del paso en un escenario web. Comportamiento de la propiedad: - *required*
url	string	URL que se comprobará. Comportamiento de la propiedad: - *required*

Property	Type	Description
follow_redirects	integer	Indica si se deben seguir las redirecciones HTTP. Valores posibles: 0 - no seguir las redirecciones; 1 - *(predeterminado)* seguir las redirecciones.
headers	array	Encabezados HTTP que se enviarán al realizar una solicitud. Los encabezados del paso del escenario sobrescribirán los encabezados especificados para el escenario web.
posts	string/array	Variables HTTP POST como una cadena (datos POST sin procesar) o como un array de campos HTTP (datos de campos del formulario).
required	string	Texto que debe estar presente en la respuesta.
retrieve_mode	integer	Parte de la respuesta HTTP que debe recuperar el paso del escenario. Valores posibles: 0 - *(predeterminado)* solo el cuerpo; 1 - solo los encabezados; 2 - encabezados y cuerpo.
status_codes	string	Intervalos de códigos de estado HTTP requeridos, separados por comas.
timeout	string	Tiempo de espera de la solicitud en segundos. Acepta un número de segundos (por ejemplo, 1, 60, 300), un valor con un sufijo de tiempo (por ejemplo, 30s, 1m, 1h) o una macro de usuario. Predeterminado: 15s. Máximo: 1h. Mínimo: 1s.
variables	array	Variables del paso del escenario.
query_fields	array	Campos de consulta: array de campos HTTP que se añadirán a la URL al realizar una solicitud.

Campo HTTP

El objeto de campo HTTP define el nombre y el valor que se utilizan para especificar las variables del escenario web, las cabeceras HTTP y los campos POST o los campos de consulta. Tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
name	string	Nombre de la cabecera/variable/campo POST o GET. Comportamiento de la propiedad: - *obligatorio*
value	string	Valor de la cabecera/variable/campo POST o GET. Comportamiento de la propiedad: - *obligatorio*

Etiqueta de escenario web

El objeto de etiqueta de escenario web tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
tag	string	Nombre de la etiqueta del escenario web. Comportamiento de la propiedad: - *obligatorio*
value	string	Valor de la etiqueta del escenario web. Comportamiento de la propiedad: - *solo lectura*.

Propiedad	Tipo	Descripción
object	integer	Tipo de objeto del que se heredó la etiqueta. Valores posibles: 0 - template; 1 - host. Comportamiento de la propiedad: - *solo lectura*.
objectid	ID	ID del objeto del que se heredó la etiqueta. Comportamiento de la propiedad: - *solo lectura*.

httptest.create

Descripción

`object httptest.create(object/array webScenarios)`

Este método le permite crear nuevos escenarios web.

Note:
Al crear un escenario web, se creará automáticamente un conjunto de elementos de monitorización web.

Note:
Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para invocar el método se pueden revocar en la configuración del rol de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object/array)` Escenarios web a crear.

Además de las propiedades estándar del escenario web, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
steps	array	Pasos del escenario. Comportamiento del parámetro: - *obligatorio*
tags	array	Etiquetas del escenario web.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los escenarios web creados bajo la propiedad `httptestids`. El orden de los IDs devueltos coincide con el orden de los escenarios web enviados.

Ejemplos

Creando un escenario web

Cree un escenario web para monitorizar la página principal de la empresa. El escenario tendrá dos pasos, para comprobar la página principal y la página "Acerca de" y asegurarse de que devuelven el código de estado HTTP 200.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "httptest.create",
    "params": {
        "name": "Homepage check",
        "hostid": "10085",
        "steps": [
            {
```

```
                "name": "Homepage",
                "url": "http://example.com",
                "status_codes": "200",
                "no": 1
            },
            {
                "name": "Homepage / About",
                "url": "http://example.com/about",
                "status_codes": "200",
                "no": 2
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "httptestids": [
            "5"
        ]
    },
    "id": 1
}
```

Ver también

- Paso de escenario

Origen

CHttpTest::create() en *ui/include/classes/api/services/CHttpTest.php*.

httptest.delete

Descripción

`object httptest.delete(array webScenarioIds)`

Este método permite eliminar escenarios web.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(array)` IDs de los escenarios web a eliminar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los escenarios web eliminados bajo la propiedad `httptestids`.

Ejemplos

Eliminando múltiples escenarios web

Eliminar dos escenarios web.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "httptest.delete",
    "params": [
        "2",
```

```
        "3"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "httptestids": [
            "2",
            "3"
        ]
    },
    "id": 1
}
```

Fuente

CHttpTest::delete() en *ui/include/classes/api/services/CHttpTest.php*.

httptest.get

Descripción

`integer/array httptest.get(object parameters)`

El método permite recuperar escenarios web según los parámetros proporcionados.

Note:
Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
groupids	ID/array	Devolver solo los escenarios web que pertenecen a los grupos de hosts indicados.
hostids	ID/array	Devolver solo los escenarios web que pertenecen a los hosts indicados.
httptestids	ID/array	Devolver solo los escenarios web con los IDs indicados.
inherited	boolean	Si se establece en `true`, devuelve solo los escenarios web heredados de una template.
inheritedTags	boolean	Devuelve los escenarios web que tienen las `tags` indicadas también en la template/host/templates vinculadas. Valores posibles: `true` - la template/host/templates vinculadas también deben tener las tags indicadas; `false` - *(por defecto)* se ignoran las tags de la template/host/templates vinculadas.
monitored	boolean	Si se establece en `true`, devuelve solo los escenarios web habilitados que pertenecen a hosts monitorizados.
selectInheritedTags	query	Devuelve una propiedad `inheritedTags` con las tags que están en la template/host/templates vinculadas.
templated	boolean	Si se establece en `true`, devuelve solo los escenarios web que pertenecen a templates.
templateids	ID/array	Devolver solo los escenarios web que pertenecen a las templates indicadas.
expandName	flag	Expande macros en el nombre del escenario web.
expandStepName	flag	Expande macros en los nombres de los pasos del escenario.

Parámetro	Tipo	Descripción
evaltype	integer	Método de evaluación de las tags. Valores posibles: 0 - *(por defecto)* And/Or; 2 - Or.
tags	array	Devolver solo los escenarios web con las tags indicadas. Formato: `[{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]`. Un array vacío devuelve todos los escenarios web. Valores posibles de operator: 0 - *(por defecto)* Contiene; 1 - Igual a; 2 - No contiene; 3 - No es igual a; 4 - Existe; 5 - No existe.
selectHosts	query	Devuelve los hosts a los que pertenece el escenario web como un array en la propiedad `hosts`.
selectSteps	query	Devuelve los pasos del escenario web en la propiedad `steps`. Admite `count`.
selectTags	query	Devuelve las tags del escenario web en la propiedad `tags`.
sortfield	string/array	Ordena el resultado por las propiedades indicadas. Valores posibles: `httptestid`, `name`.
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

`(entero/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperar un escenario web

Recuperar todos los datos sobre el escenario web "9".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "httptest.get",
    "params": {
        "output": "extend",
        "selectSteps": "extend",
        "httptestids": "9"
    },
    "id": 1
}
```

Respuesta:


```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "httptestid": "9",
            "name": "Homepage check",
            "delay": "1m",
            "status": "0",
            "agent": "Zabbix",
            "authentication": "0",
            "http_user": "",
            "http_password": "",
            "hostid": "10084",
            "templateid": "0",
            "http_proxy": "",
            "retries": "1",
            "ssl_cert_file": "",
            "ssl_key_file": "",
            "ssl_key_password": "",
            "verify_peer": "0",
            "verify_host": "0",
            "uuid": "",
            "headers": [],
            "variables": [],
            "steps": [
                {
                    "httpstepid": "36",
                    "httptestid": "9",
                    "name": "Homepage",
                    "no": "1",
                    "url": "http://example.com",
                    "timeout": "15s",
                    "posts": "",
                    "required": "",
                    "status_codes": "200",
                    "follow_redirects": "1",
                    "retrieve_mode": "0",
                    "post_type": "1",
                    "headers": [],
                    "variables": [
                        {
                            "name":"{var}",
                            "value":"12"
                        }
                    ],
                    "query_fields": []
                },
                {
                    "httpstepid": "37",
                    "httptestid": "9",
                    "name": "Homepage / About",
                    "no": "2",
                    "url": "http://example.com/about",
                    "timeout": "15s",
                    "posts": "",
                    "required": "",
                    "status_codes": "200",
                    "follow_redirects": "1",
                    "retrieve_mode": "0",
                    "post_type": "1",
                    "headers": [],
```

```
                    "variables": [],
                    "query_fields": []
                }
            ]
        }
    ],
    "id": 1
}
```

Ver también

- Host
- Paso del escenario

Fuente

CHttpTest::get() en *ui/include/classes/api/services/CHttpTest.php*.

httptest.update

Descripción

`object httptest.update(object/array webScenarios)`

Este método permite actualizar escenarios web existentes.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object/array)` Propiedades del escenario web que se van a actualizar.

La propiedad `httptestid` debe estar definida para cada escenario web, todas las demás propiedades son opcionales. Sólo se actualizarán las propiedades pasadas, todas las demás permanecerán sin cambios.

Además de las propiedades estándar del escenario web, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
steps	array	Pasos del escenario para reemplazar los pasos existentes.
tags	array	Etiquetas del escenario web.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los escenarios web actualizados bajo la propiedad `httptestid`.

Ejemplos

Habilitar un escenario web

Habilite un escenario web, es decir, establezca su estado en "0".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "httptest.update",
    "params": {
        "httptestid": "5",
        "status": 0
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "httptestids": [
            "5"
        ]
    },
    "id": 1
}
```

Ver también

- Paso de escenario

Fuente

CHttpTest::update() en *ui/include/classes/api/services/CHttpTest.php*.

Event

Esta clase está diseñada para trabajar con eventos.

Referencias de objetos:

- Event
- Event tag
- Media type URL

Métodos disponibles:

- event.get - recuperar eventos
- event.acknowledge - reconocer eventos

Objeto evento

Los siguientes objetos están directamente relacionados con la API de `event`.

Evento

> **Note:**
> Los eventos son creados por el servidor Zabbix y no pueden ser modificados a través de la API.

El objeto evento tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
eventid	ID	ID del evento.
source	integer	Tipo de evento. Valores posibles: 0 - evento creado por un trigger; 1 - evento creado por una regla de descubrimiento; 2 - evento creado por autoregistro activo de agent; 3 - evento interno; 4 - evento creado en la actualización del estado del servicio.

Propiedad	Tipo	Descripción
object	integer	Tipo de objeto relacionado con el evento. Valores posibles si `source` está establecido en ”evento creado por un trigger”: 0 - trigger. Valores posibles si `source` está establecido en ”evento creado por una regla de descubrimiento”: 1 - host descubierto; 2 - servicio descubierto. Valores posibles si `source` está establecido en ”evento creado por autoregistro activo de agent”: 3 - host autoregistrado. Valores posibles si `source` está establecido en ”evento interno”: 0 - trigger; 4 - item; 5 - regla LLD. Valores posibles si `source` está establecido en ”evento creado en la actualización del estado del servicio”: 6 - servicio.
objectid	ID	ID del objeto relacionado.
acknowledged	integer	Indica si el evento ha sido reconocido.
clock	timestamp	Hora en que se creó el evento.
ns	integer	Nanosegundos en que se creó el evento.
name	string	Nombre del evento resuelto.
value	integer	Estado del objeto relacionado. Valores posibles si `source` está establecido en ”evento creado por un trigger” o ”evento creado en la actualización del estado del servicio”: 0 - OK; 1 - problema. Valores posibles si `source` está establecido en ”evento creado por una regla de descubrimiento”: 0 - host o servicio activo; 1 - host o servicio inactivo; 2 - host o servicio descubierto; 3 - host o servicio perdido. Valores posibles si `source` está establecido en ”evento interno”: 0 - estado ”normal”; 1 - estado ”desconocido” o ”no soportado”. Comportamiento de la propiedad: - *soportado* si `source` está establecido en ”evento creado por un trigger”, ”evento creado por una regla de descubrimiento”, ”evento interno” o ”evento creado en la actualización del estado del servicio”
severity	integer	Severidad actual del evento. Valores posibles: 0 - no clasificado; 1 - información; 2 - advertencia; 3 - media; 4 - alta; 5 - desastre.
r_eventid	ID	ID del evento de recuperación.

Propiedad	Tipo	Descripción
c_eventid	ID	ID del evento que se utilizó para sobrescribir (cerrar) el evento actual bajo la regla de correlación global. Consulte `correlationid` para identificar la regla de correlación exacta. Este parámetro solo se define cuando el evento se cierra por una regla de correlación global.
cause_eventid	ID	ID del evento causa.
correlationid	ID	ID de la regla de correlación que generó el cierre del problema. Este parámetro solo se define cuando el evento se cierra por una regla de correlación global.
userid	ID	ID del usuario que cerró el evento (si el evento se cerró manualmente).
suppressed	integer	Indica si el evento está suprimido. Valores posibles: 0 - el evento está en estado normal; 1 - el evento está suprimido.
opdata	string	Datos operativos con macros expandidas.
urls	array	URLs de tipo de medio activas.

Etiqueta de evento

El objeto de etiqueta de evento tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
tag	string	Nombre de la etiqueta de evento.
value	string	Valor de la etiqueta de evento.

URL del tipo de medio

El objeto URL del tipo de medio tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
name	string	Nombre de la URL definida por el tipo de medio.
url	string	Valor de la URL definida por el tipo de medio.

Los resultados contendrán entradas solo para los tipos de medio activos con la entrada de menú de evento habilitada. La macro utilizada en las propiedades se expandirá, pero si una de las propiedades contiene una macro sin expandir, ambas propiedades se excluirán de los resultados. Para las macros soportadas, consulte *Macros soportadas*.

event.acknowledge

Descripción

`object event.acknowledge(object/array parameters)`

Este método permite actualizar eventos. Se pueden realizar las siguientes acciones de actualización:

- Cerrar evento. Si el evento ya está resuelto, esta acción se omitirá.
- Reconocer evento. Si el evento ya está reconocido, esta acción se omitirá.
- Anular reconocimiento del evento. Si el evento no está reconocido, esta acción se omitirá.
- Agregar mensaje.
- Cambiar la severidad del evento. Si el evento ya tiene la misma severidad, esta acción se omitirá.
- Suprimir evento. Si el evento ya está suprimido, esta acción se omitirá.
- Anular supresión del evento. Si el evento no está suprimido, esta acción se omitirá.
- Cambiar la clasificación del evento a causa. Si el evento ya está clasificado como causa, esta acción se omitirá.
- Cambiar la clasificación del evento a síntoma. Si el evento ya está clasificado como síntoma, esta acción se omitirá.

Attention:
Solo se pueden actualizar eventos de trigger.
 Solo se pueden actualizar eventos de problema.
 Se requieren permisos de lectura/escritura para el trigger para cerrar el evento o cambiar la severidad del evento.
 Para cerrar un evento, el cierre manual debe estar permitido en el trigger.

Note:
Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración del rol de usuario. Consulte User roles para obtener más información.

Parámetros

`(object/array)` Parámetros que contienen los ID de los eventos y las operaciones de actualización que deben realizarse.

Parameter	Type	Description
eventids	ID/array	ID de los eventos que se deben reconocer. Comportamiento del parámetro: - *obligatorio*
action	integer	Acción o acciones de actualización del evento. Valores de bitmap posibles: 1 - cerrar el problema; 2 - reconocer el evento; 4 - añadir un mensaje; 8 - cambiar la gravedad; 16 - dejar de reconocer el evento; 32 - suprimir el evento; 64 - dejar de suprimir el evento; 128 - cambiar el rango del evento a causa; 256 - cambiar el rango del evento a síntoma. Este es un campo de máscara de bits; se acepta cualquier suma de los valores de bitmap posibles (por ejemplo, 6 para reconocer un evento y dejar un mensaje, o 34 para reconocer y suprimir un evento). Comportamiento del parámetro: - *obligatorio*
cause_eventid	ID	ID del evento causante. Comportamiento del parámetro: - *obligatorio* si `action` contiene el bit «cambiar el rango del evento a síntoma»
message	string	Texto del mensaje. Comportamiento del parámetro: - *obligatorio* si `action` contiene el bit «añadir un mensaje»
severity	integer	Nueva gravedad de los eventos. Valores posibles: 0 - no clasificado; 1 - información; 2 - advertencia; 3 - promedio; 4 - alto; 5 - desastre. Comportamiento del parámetro: - *obligatorio* si `action` contiene el bit «cambiar la gravedad»
suppress_until	integer	Marca de tiempo de Unix hasta la que se debe suprimir el evento. Si se establece en «0», la supresión será indefinida. Comportamiento del parámetro: - *obligatorio* si `action` contiene el bit «suprimir el evento»

Valores de retorno

(`object`) Devuelve un objeto que contiene los IDs de los eventos actualizados en la propiedad `eventids`.

Ejemplos

Reconociendo un evento

Reconocer un solo evento y dejar un mensaje.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "event.acknowledge",
    "params": {
        "eventids": "20427",
        "action": 6,
        "message": "Problema resuelto."
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "eventids": [
            "20427"
        ]
    },
    "id": 1
}
```

Cambiar la gravedad del evento

Cambiar la gravedad de varios eventos y dejar un mensaje.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "event.acknowledge",
    "params": {
        "eventids": ["20427", "20428"],
        "action": 12,
        "message": "Se requiere mantenimiento para solucionarlo.",
        "severity": 4
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "eventids": [
            "20427",
            "20428"
        ]
    },
    "id": 1
}
```

Suprimir un evento indefinidamente

Suprime un único evento sin tiempo de expiración.

Request:

```
{
    "jsonrpc": "2.0",
    "method": "event.acknowledge",
    "params": {
        "eventids": "20427",
        "action": 32,
        "suppress_until": 0
    },
    "id": 1
}
```

Response:

```
{
    "jsonrpc": "2.0",
    "result": {
        "eventids": [
            20427
        ]
    },
    "id": 1
}
```

Anular la supresión de un evento

Anule la supresión de un único evento previamente suprimido.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "event.acknowledge",
    "params": {
        "eventids": "20427",
        "action": 64
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "eventids": [
            20427
        ]
    },
    "id": 1
}
```

Reconocer y suprimir un evento

Reconozca uno o más eventos, agregue un mensaje y suprímalos en la misma solicitud.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "event.acknowledge",
    "params": {
        "eventids": [
            "20427",
            "20428"
        ],
        "action": 38,
        "message": "Suppressing during planned maintenance.",
        "suppress_until": 0
```

```
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "eventids": [
            20427,
            20428
        ]
    },
    "id": 1
}
```

Fuente

CEvent::acknowledge() en *ui/include/classes/api/services/CEvent.php*.

event.get

Descripción

`integer/array event.get(object parameters)`

El método permite recuperar eventos según los parámetros proporcionados.

Attention:
Este método puede devolver eventos de una entidad eliminada si estos eventos aún no han sido eliminados por el housekeeper.

Note:
Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parameter	Type	Description
eventids	ID/array	Devuelve solo eventos con los IDs dados.
groupids	ID/array	Devuelve solo eventos creados por objetos que pertenecen a los grupos de hosts dados.
hostids	ID/array	Devuelve solo eventos creados por objetos que pertenecen a los hosts dados.
objectids	ID/array	Devuelve solo eventos creados por los objetos dados.
source	integer	Devuelve solo eventos con el tipo dado. Consulte la página del objeto event para obtener una lista de los tipos de evento admitidos. Predeterminado: 0 - eventos de trigger.
object	integer	Devuelve solo eventos creados por objetos del tipo dado. Consulte la página del objeto event para obtener una lista de los tipos de objeto admitidos. Predeterminado: 0 - trigger.
acknowledged	boolean	Si se establece en `true`, devuelve solo eventos reconocidos.

Parameter	Type	Description
action	integer	Devuelve solo eventos para los que se han realizado las acciones de actualización de evento dadas. Para varias acciones, use una suma de cualquier valor de mapa de bits aceptable como máscara de bits (por ejemplo, 34 para reconocer y suprimir el evento).
action_userids	ID/array	Devuelve solo eventos con los IDs dados de los usuarios que realizaron las acciones de actualización del evento.
suppressed	boolean	Si se establece en `true`, devuelve solo eventos suprimidos.
symptom	boolean	Si se establece en `true`, devuelve solo eventos de síntoma.
severities	integer/array	Devuelve solo eventos con las severidades de evento dadas. Se aplica solo si object es trigger.
trigger_severities	integer/array	Devuelve solo eventos con las severidades de trigger dadas. Se aplica solo si object es trigger.
evaltype	integer	Método de evaluación de etiquetas. Valores posibles: 0 - *(predeterminado)* And/Or; 2 - Or.
tags	array	Devuelve solo eventos con las etiquetas dadas. Formato: `[{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]`. Un array vacío devuelve todos los eventos. Valores posibles de operator: 0 - *(predeterminado)* Contiene; 1 - Igual a; 2 - No contiene; 3 - No es igual a; 4 - Existe; 5 - No existe.
eventid_from	string	Devuelve solo eventos con IDs mayores o iguales que el ID dado.
eventid_till	string	Devuelve solo eventos con IDs menores o iguales que el ID dado.
time_from	timestamp	Devuelve solo eventos creados en o después de la hora dada.
time_till	timestamp	Devuelve solo eventos creados en o antes de la hora dada.
problem_time_from	timestamp	Devuelve solo eventos que estuvieron en estado de problema desde `problem_time_from`, independientemente de su estado actual. Se aplica solo cuando la fuente es un evento de trigger y el objeto es un trigger. Este parámetro es obligatorio si se especifica `problem_time_till`.
problem_time_till	timestamp	Devuelve solo eventos que estuvieron en estado de problema hasta `problem_time_till`, independientemente de su estado actual. Se aplica solo cuando la fuente es un evento de trigger y el objeto es un trigger. Este parámetro es obligatorio si se especifica `problem_time_from`.
value	integer/array	Devuelve solo eventos con los valores dados.

Parameter	Type	Description
selectAcknowledges	query	Devuelve una propiedad `acknowledges` con las actualizaciones del evento. Las actualizaciones del evento se ordenan en orden cronológico inverso. El objeto de actualización del evento tiene las siguientes propiedades: `acknowledgeid` - `(ID)` ID del reconocimiento; `userid` - `(ID)` ID del usuario que actualizó el evento; `clock` - `(timestamp)` hora en que se actualizó el evento; `message` - `(string)` texto del mensaje; `action` - `(integer)` acción de actualización que se realizó, consulte `event.acknowledge`; `old_severity` - `(integer)` severidad del evento antes de esta acción de actualización; `new_severity` - `(integer)` severidad del evento después de esta acción de actualización; `suppress_until` - `(timestamp)` hora hasta la que el evento estará suprimido; `taskid` - `(ID)` ID de la tarea si el evento actual está sufriendo un cambio de rango; `maintenanceid` - `(ID)` ID del mantenimiento; `username` - `(string)` nombre de usuario del usuario que actualizó el evento; `name` - `(string)` nombre del usuario que actualizó el evento; `surname` - `(string)` apellido del usuario que actualizó el evento. Admite `count`.
selectAlerts	query	Devuelve una propiedad `alerts` con las alertas generadas por el evento. Las alertas se ordenan en orden cronológico inverso.
selectHosts	query	Devuelve una propiedad `hosts` con los hosts que contienen el objeto que creó el evento. Solo se admite para eventos generados por triggers, items o reglas LLD.
selectRelatedObject	query	Devuelve una propiedad `relatedObject` con el objeto que creó el evento. El tipo de objeto devuelto depende del tipo de evento.
selectSuppressionData	query	Devuelve una propiedad `suppression_data` con la lista de mantenimientos activos y supresiones manuales: `maintenanceid` - `(ID)` ID del mantenimiento; `userid` - `(ID)` ID del usuario que suprimió el evento; `suppress_until` - `(integer)` tiempo hasta el que el evento está suprimido.
selectTags	query	Devuelve una propiedad `tags` con las etiquetas del evento.
filter	object	Devuelve solo aquellos resultados que coinciden exactamente con el filtro dado. Acepta un objeto, donde las claves son nombres de propiedades y los valores son un único valor o un array de valores con los que comparar. No admite propiedades del tipo de datos data type `text`.
sortfield	string/array	Ordena el resultado por las propiedades dadas. Valores posibles: `eventid`, `objectid`, `clock`. Valores posibles cuando se usa junto con `groupBy`: `objectid`. Valores posibles cuando se usa junto con `countOutput` y `groupBy`: `objectid`, `rowscount`.
groupBy	string/array	Agrupa los resultados por las propiedades dadas. Las propiedades especificadas se devolverán en los resultados. Valores posibles: `objectid`.
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
editable	boolean	

Parameter	Type	Description
excludeSearch	boolean	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

`(integer/array)` Devuelve cualquiera de los siguientes:

- un array de objetos
- el recuento de objetos recuperados, si se ha usado el parámetro `countOutput`, pero no se ha usado el parámetro `groupBy`
- un array de objetos con resultados de agregación, si se ha usado el parámetro `groupBy`

Ejemplos

Recuperando eventos de trigger

Recupere los eventos más recientes del trigger "22395".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "event.get",
    "params": {
        "output": "extend",
        "selectAcknowledges": "extend",
        "selectSuppressionData": "extend",
        "selectTags": "extend",
        "objectids": "22395",
        "sortfield": ["clock", "eventid"],
        "sortorder": "DESC"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "eventid": "20",
            "source": "0",
            "object": "0",
            "objectid": "22395",
            "clock": "1728658089",
            "value": "0",
            "acknowledged": "0",
            "ns": "461809482",
            "name": "El promedio de carga es demasiado alto (carga por CPU superior a 1.5 durante 5m)",
            "severity": "0",
            "r_eventid": "0",
            "c_eventid": "0",
            "correlationid": "0",
            "userid": "0",
            "suppressed": "0",
            "cause_eventid": "0",
            "acknowledges": [],
            "opdata": "Promedios de carga(1m 5m 15m): (1.6328 3.0522 2.2515), # de CPUs: 2",
```

```
            "suppression_data": [],
            "tags": [
                {
                    "tag": "scope",
                    "value": "capacity"
                },
                {
                    "tag": "scope",
                    "value": "performance"
                },
                {
                    "tag": "component",
                    "value": "cpu"
                },
                {
                    "tag": "class",
                    "value": "os"
                },
                {
                    "tag": "target",
                    "value": "linux"
                }
            ],
            "urls": []
        },
        {
            "eventid": "4",
            "source": "0",
            "object": "0",
            "objectid": "22395",
            "clock": "1728657737",
            "value": "1",
            "acknowledged": "1",
            "ns": "460759366",
            "name": "El promedio de carga es demasiado alto (carga por CPU superior a 1.5 durante 5m)",
            "severity": "3",
            "r_eventid": "20",
            "c_eventid": "0",
            "correlationid": "0",
            "userid": "0",
            "suppressed": "1",
            "cause_eventid": "0",
            "acknowledges": [
                {
                    "acknowledgeid": "1",
                    "userid": "1",
                    "clock": "1728657938",
                    "message": "Testing environment. Please disregard this alert.",
                    "action": "38",
                    "old_severity": "0",
                    "new_severity": "0",
                    "suppress_until": "1728744338",
                    "taskid": "0",
                    "maintenanceid": "0",
                    "username": "Admin",
                    "name": "Zabbix",
                    "surname": "Administrator"
                }
            ],
            "opdata": "Promedios de carga(1m 5m 15m): (1.6328 3.0522 2.2515), # de CPUs: 2",
            "suppression_data": [
                {
```

```
                    "maintenanceid": "0",
                    "suppress_until": "1728744338",
                    "userid": "1"
                }
            ],
            "tags": [
                {
                    "tag": "scope",
                    "value": "capacity"
                },
                {
                    "tag": "scope",
                    "value": "performance"
                },
                {
                    "tag": "component",
                    "value": "cpu"
                },
                {
                    "tag": "class",
                    "value": "os"
                },
                {
                    "tag": "target",
                    "value": "linux"
                }
            ],
            "urls": []
        }
    ],
    "id": 1
}
```


Recuperar eventos por periodo de tiempo

Recuperar todos los eventos que se han creado entre el 17 y el 18 de octubre de 2012, en orden cronológico inverso.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "event.get",
    "params": {
        "output": "extend",
        "time_from": "1350432000",
        "time_till": "1350518400",
        "sortfield": ["clock", "eventid"],
        "sortorder": "DESC"
    },
    "id": 1
}
```

Respuesta:


```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "eventid": "20617",
            "source": "0",
            "object": "0",
            "objectid": "14282",
            "clock": "1350477816",
            "value": "1",
            "acknowledged": "0",
```

```
            "ns": "0",
            "name": "Less than 25% free in the history cache",
            "severity": "3",
            "r_eventid": "0",
            "c_eventid": "0",
            "correlationid": "0",
            "userid": "0",
            "cause_eventid": "0",
            "opdata": "",
            "suppressed": "0",
            "urls": []
        },
        {
            "eventid": "20616",
            "source": "0",
            "object": "0",
            "objectid": "14281",
            "clock": "1350477814",
            "value": "0",
            "acknowledged": "0",
            "ns": "0",
            "name": "Zabbix trapper processes more than 75% busy",
            "severity": "0",
            "r_eventid": "0",
            "c_eventid": "0",
            "correlationid": "0",
            "userid": "0",
            "cause_eventid": "0",
            "opdata": "",
            "suppressed": "0",
            "urls": []
        },
        {
            "eventid": "20615",
            "source": "0",
            "object": "0",
            "objectid": "14281",
            "clock": "1350477541",
            "value": "1",
            "acknowledged": "0",
            "ns": "0",
            "name": "Zabbix trapper processes more than 75% busy",
            "severity": "3",
            "r_eventid": "20616",
            "c_eventid": "0",
            "correlationid": "0",
            "userid": "0",
            "cause_eventid": "0",
            "opdata": "",
            "suppressed": "0",
            "urls": []
        }
    ],
    "id": 1
}
```


Recuperar eventos reconocidos por un usuario especificado

Recuperar eventos reconocidos por el usuario con ID=10

Solicitud:


```
{
    "jsonrpc": "2.0",
```

```
    "method": "event.get",
    "params": {
        "output": "extend",
        "action": 2,
        "action_userids": [10],
        "selectAcknowledges": ["userid", "action"],
        "sortfield": ["eventid"],
        "sortorder": "DESC"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "eventid": "503",
            "source": "0",
            "object": "0",
            "objectid": "23162",
            "clock": "1747212236",
            "value": "1",
            "acknowledged": "1",
            "ns": "413470863",
            "name": "Number of installed packages has been changed",
            "severity": "2",
            "r_eventid": "0",
            "c_eventid": "0",
            "correlationid": "0",
            "userid": "0",
            "suppressed": "0",
            "cause_eventid": "0",
            "acknowledges": [
                {
                    "userid": "10",
                    "action": "2"
                }
            ],
            "opdata": "",
            "urls": []
        }
    ],
    "id": 1
}
```

Recuperar los triggers principales con el recuento de eventos de problema

Recuperar los 5 triggers principales que tienen las severidades ”Warning”, ”Average”, ”High” o ”Disaster”, junto con el número de eventos de problema dentro de un período de tiempo especificado.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "event.get",
    "params": {
        "countOutput": true,
        "groupBy": "objectid",
        "source": 0,
        "object": 0,
        "value": 1,
        "time_from": 1672531200,
        "time_till": 1677628800,
```

```
        "trigger_severities": [2, 3, 4, 5],
        "sortfield": ["rowscount"],
        "sortorder": "DESC",
        "limit": 5
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "objectid": "232124",
            "rowscount": "27"
        },
        {
            "objectid": "29055",
            "rowscount": "23"
        },
        {
            "objectid": "253731",
            "rowscount": "18"
        },
        {
            "objectid": "254062",
            "rowscount": "11"
        },
        {
            "objectid": "23216",
            "rowscount": "7"
        }
    ],
    "id": 1
}
```

Véase también

- Alerta
- Métrica
- Equipo
- Regla LLD
- Servicio
- Disparador

Fuente

CEvent::get() en *ui/include/classes/api/services/CEvent.php*.

Expresión regular

Esta clase está diseñada para trabajar con expresiones regulares globales.

Referencias de objetos:

- Expresión regular
- Expresiones

Métodos disponibles:

- regexp.create - crear nuevas expresiones regulares
- regexp.delete - eliminar expresiones regulares
- regexp.get - recuperar expresiones regulares
- regexp.update - actualizar expresiones regulares

Objeto de expresión regular

Los siguientes objetos están directamente relacionados con la API `regexp`.

Expresión regular

El objeto de expresión regular global tiene las siguientes propiedades.

Property	Type	Description
regexpid	ID	ID de la expresión regular. Property behavior: - *read-only* - *required* for update operations.
name	string	Nombre de la expresión regular. Property behavior: - *required* for create operations.
description	text	Descripción de la expresión regular.
test_string	string	Cadena de prueba.

Expresiones

El objeto expressions tiene las siguientes propiedades.

Property	Type	Description
expression	string	Expresión regular. Property behavior: - *required*
expression_type	integer	Tipo de expresión regular. Valores posibles: 0 - Contiene la cadena; 1 - Contiene cualquier subcadena de la lista; 2 - No contiene la cadena; 3 - Coincide con la expresión regular; 4 - No coincide con la expresión regular. Property behavior: - *required*
exp_delimiter	string	Delimitador de la expresión. Valor predeterminado: `","`. Valores posibles: `","` o `"."`, o `"/"`. Property behavior: - *supported* si `expression_type` está establecido en *Contains any substring from list*.
case_sensitive	integer	Distinción entre mayúsculas y minúsculas. Valor predeterminado: 0. Valores posibles: 0 - No distingue mayúsculas y minúsculas; 1 - Distingue mayúsculas y minúsculas.

regexp.create

Descripción

`object regexp.create(object/array regularExpressions)`

Este método permite crear nuevas expresiones regulares globales.

> **Note:**
> Este método solo está disponible para tipos de usuario *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object/array)` Expresiones regulares a crear.

Además de las propiedades estándar, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
expressions	array	Opciones de expresiones. Comportamiento del parámetro: - *obligatorio*

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las expresiones regulares creadas bajo la propiedad `regexpids`.

Ejemplos

Creación de una nueva expresión regular global.

Request:

```
{
    "jsonrpc": "2.0",
    "method": "regexp.create",
    "params": {
        "name": "Storage devices for SNMP discovery",
        "description": "Global regular expression used to discover and identify memory resources on the mo
        "test_string": "/boot",
        "expressions": [
            {
                "expression": "^(Physical memory|Virtual memory|Memory buffers|Cached memory|Swap space)$"
                "expression_type": "4",
                "case_sensitive": "1"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "regexpids": [
            "9"
        ]
    },
    "id": 1
}
```

Origen

CRegexp::create() en *ui/include/classes/api/services/CRegexp.php*.

regexp.delete

Descripción

`object regexp.delete(array regexpids)`

Este método permite eliminar expresiones regulares globales.

> **Note:**
> Este método solo está disponible para tipos de usuario *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(array)` IDs de las expresiones regulares a eliminar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las expresiones regulares eliminadas bajo la propiedad `regexpids`.

Ejemplos

Eliminando múltiples expresiones regulares globales.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "regexp.delete",
    "params": [
        "16",
        "17"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "regexpids": [
            "16",
            "17"
        ]
    },
    "id": 1
}
```

Fuente

CRegexp::delete() en *ui/include/classes/api/services/CRegexp.php*.

regexp.get

Descripción

`integer/array regexp.get(object parameters)`

El método permite recuperar expresiones regulares globales según los parámetros proporcionados.

> **Note:**
> Este método está disponible solo para *Super Admin*. Los permisos para llamar al método pueden revocarse en la configuración del rol de usuario. Consulte User roles para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
regexpids	ID/array	Devolver solo las expresiones regulares con los IDs dados.
selectExpressions	query	Devolver una propiedad `expressions`.
sortfield	string/array	Ordenar el resultado por las propiedades dadas. Valores posibles: `regexpid`, `name`.
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperación de expresiones regulares globales.

Request:

```
{
    "jsonrpc": "2.0",
    "method": "regexp.get",
    "params": {
        "output": ["regexpid","name","description"],
        "selectExpressions": ["expression","expression_type"],
        "regexpids": [1,2],
        "preservekeys": true
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "1": {
            "regexpid": "1",
            "name": "Sistemas de archivos para descubrimiento",
            "description": "",
            "expressions": [
                {
                    "expression": "^(btrfs|ext2|ext3|ext4|reiser|xfs|ffs|ufs|jfs|jfs2|vxfs|hfs|apfs|refs|n
                    "expression_type": "3"
                }
            ]
        },
        "2": {
            "regexpid": "2",
            "name": "Interfaces de red para descubrimiento",
            "description": "",
            "expressions": [
```

```
                {
                    "expression": "^Software Loopback Interface",
                    "expression_type": "4"
                },
                {
                    "expression": "^(In)?[Ll]oop[Bb]ack[0-9._]*$",
                    "expression_type": "4"
                },
                {
                    "expression": "^NULL[0-9.]*$",
                    "expression_type": "4"
                },
                {
                    "expression": "^[Ll]o[0-9.]*$",
                    "expression_type": "4"
                },
                {
                    "expression": "^[Ss]ystem$",
                    "expression_type": "4"
                },
                {
                    "expression": "^Nu[0-9.]*$",
                    "expression_type": "4"
                }
            ]
        }
    },
    "id": 1
}
```


Fuente

CRegexp::get() en *ui/include/classes/api/services/CRegexp.php*.

regexp.update

Descripción

`object regexp.update(object/array regularExpressions)`

Este método permite actualizar las expresiones regulares globales existentes.

> **Note:**
> Este método solo está disponible para tipos de usuario *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object/array)` Propiedades de la expresión regular que se van a actualizar.

La propiedad `regexpid` debe estar definida para cada objeto, todas las demás propiedades son opcionales. Sólo se actualizarán las propiedades pasadas, todas las demás permanecerán sin cambios.

Además de las propiedades estándar, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
expressions	array	Opciones de expresiones.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las expresiones regulares actualizadas bajo la propiedad `regexpids`.

Ejemplos

Actualización de la expresión regular global para el descubrimiento de sistemas de archivos.

Request:

```
{
    "jsonrpc": "2.0",
    "method": "regexp.update",
    "params": {
        "regexpid": "1",
        "name": "File systems for discovery",
        "description": "Global regular expression used to discover and identify file systems on the monito
        "test_string": "",
        "expressions": [
            {
                "expression": "^(btrfs|ext2|ext3|ext4|reiser|xfs|ffs|ufs|jfs|jfs2|vxfs|hfs|apfs|refs|zfs)$
                "expression_type": "3",
                "case_sensitive": "0"
            },
            {
                "expression": "^(ntfs|fat32|fat16)$",
                "expression_type": "3",
                "case_sensitive": "0"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "regexpids": [
            "1"
        ]
    },
    "id": 1
}
```

Fuente

CRegexp::update() en *ui/include/classes/api/services/CRegexp.php*.

Grupo de hosts

Esta clase está diseñada para trabajar con grupos de hosts.

Referencias de objetos:

- Grupo de hosts

Métodos disponibles:

- hostgroup.create - crear nuevos grupos de hosts
- hostgroup.delete - eliminar grupos de hosts
- hostgroup.get - obtener grupos de hosts
- hostgroup.massadd - añadir objetos relacionados a los grupos de hosts
- hostgroup.massremove - eliminar objetos relacionados de los grupos de hosts
- hostgroup.propagate - propagar permisos y filtros de etiquetas a los subgrupos de los grupos de hosts
- hostgroup.update - actualizar grupos de hosts

Objeto de grupo de equipos

Los siguientes objetos están directamente relacionados con la API de `hostgroup`.

Grupo de hosts

El objeto grupo de hosts tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
groupid	ID	ID del grupo de hosts. Comportamiento de la propiedad: - *solo lectura* - *requerido* para operaciones de actualización
name	string	Nombre del grupo de hosts. Comportamiento de la propiedad: - *requerido* para operaciones de creación
flags	integer	Origen del grupo de hosts. Valores posibles: 0 - un grupo de hosts normal; 4 - un grupo de hosts descubierto. Comportamiento de la propiedad: - *solo lectura*
uuid	string	Identificador único universal, utilizado para vincular grupos de hosts importados a los ya existentes. Se genera automáticamente si no se proporciona.

hostgroup.create

Descripción

`object hostgroup.create(object/array hostGroups)`

Este método le permite crear nuevos grupos de hosts.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object/array)` Grupos de host a crear.

El método acepta grupos de host con las propiedades estándar de los grupos de host.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los grupos de equipos creados bajo la propiedad `groupids`. El orden de los IDs devueltos coincide con el orden de los grupos de equipos pasados.

Ejemplos

Creando un grupo de equipos

Cree un grupo de equipos llamado ”Linux servers”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "hostgroup.create",
    "params": {
        "name": "Linux servers"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
```

```
        "groupids": [
            "107819"
        ]
    },
    "id": 1
}
```

Fuente

CHostGroup::create() en *ui/include/classes/api/services/CHostGroup.php*.

hostgroup.delete

Descripción

`object hostgroup.delete(array hostGroupIds)`

Este método le permite eliminar grupos de hosts.

No se puede eliminar un grupo de hosts si:

- contiene hosts que pertenecen solo a este grupo;
- está marcado como interno;
- lo utiliza un host prototype;
- se utiliza en un script global;
- se utiliza en una condición de correlación.

Note:
Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(array)` IDs de los grupos de equipos a eliminar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los grupos de equipos eliminados bajo la propiedad `groupids`.

Ejemplos

Eliminando varios grupos de equipos

Eliminar dos grupos de equipos.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "hostgroup.delete",
    "params": [
        "107824",
        "107825"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "107824",
            "107825"
        ]
    },
    "id": 1
}
```

Fuente

CHostGroup::delete() en *ui/include/classes/api/services/CHostGroup.php*.

hostgroup.get

Descripción

`integer/array hostgroup.get(object parameters)`

El método permite recuperar grupos de hosts según los parámetros proporcionados.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
graphids	ID/array	Devolver solo los grupos de hosts que contienen hosts con los gráficos indicados.
groupids	ID/array	Devolver solo los grupos de hosts con los IDs de grupo de hosts indicados.
hostids	ID/array	Devolver solo los grupos de hosts que contienen los hosts indicados.
maintenanceids	ID/array	Devolver solo los grupos de hosts que están afectados por los mantenimientos indicados.
triggerids	ID/array	Devolver solo los grupos de hosts que contienen hosts con los triggers indicados.
with_graphs	boolean	Devolver solo los grupos de hosts que contienen hosts con gráficos.
with_graph_prototypes	boolean	Devolver solo los grupos de hosts que contienen hosts con prototipos de gráficos.
with_hosts	boolean	Devolver solo los grupos de hosts que contienen hosts.
with_httptests	boolean	Devolver solo los grupos de hosts que contienen hosts con comprobaciones web. Sobrescribe el parámetro `with_monitored_httptests`.
with_items	boolean	Devolver solo los grupos de hosts que contienen hosts con items. Sobrescribe los parámetros `with_monitored_items` y `with_simple_graph_items`.
with_item_prototypes	boolean	Devolver solo los grupos de hosts que contienen hosts con prototipos de item. Sobrescribe el parámetro `with_simple_graph_item_prototypes`.
with_simple_graph_item_prototypes	boolean	Devolver solo los grupos de hosts que contienen hosts con prototipos de item, que están habilitados para la creación y tienen tipo de información numérica.
with_monitored_httptests	boolean	Devolver solo los grupos de hosts que contienen hosts con comprobaciones web habilitadas.
with_monitored_hosts	boolean	Devolver solo los grupos de hosts que contienen hosts monitorizados.
with_monitored_items	boolean	Devolver solo los grupos de hosts que contienen hosts con items habilitados. Sobrescribe el parámetro `with_simple_graph_items`.
with_monitored_triggers	boolean	Devolver solo los grupos de hosts que contienen hosts con triggers habilitados. Todos los items utilizados en el trigger también deben estar habilitados.
with_simple_graph_items	boolean	Devolver solo los grupos de hosts que contienen hosts con items numéricos.

Parámetro	Tipo	Descripción
with_triggers	boolean	Devolver solo los grupos de hosts que contienen hosts con triggers. Sobrescribe el parámetro `with_monitored_triggers`.
selectDiscoveryRules	query	Devolver una propiedad `discoveryRules` con las reglas LLD que descubrieron el grupo de hosts.
selectDiscoveryData	query	Devolver una propiedad `discoveryData` con los objetos de descubrimiento de grupo de hosts. Cada objeto de descubrimiento de grupo de hosts es un prototipo de grupo de hosts vinculado al grupo de hosts descubierto y tiene las siguientes propiedades: `parent_group_prototypeid` - `(ID)` ID del prototipo de grupo de hosts desde el que se descubrió el grupo de hosts; `name` - `(string)` nombre del prototipo de grupo de hosts; `ts_delete` - `(timestamp)` hora en que se eliminará el grupo de hosts que ya no se descubre; `status` - `(int)` estado de descubrimiento del grupo de hosts: 0 - *(por defecto)* el grupo de hosts está descubierto, 1 - el grupo de hosts ya no se descubre.
selectHostPrototypes	query	Devolver una propiedad `hostPrototypes` con los prototipos de host que descubrieron este grupo de hosts.
selectHosts	query	Devolver una propiedad `hosts` con los hosts que pertenecen al grupo de hosts. Admite `count`.
limitSelects	integer	Limita el número de registros devueltos por subconsultas. Se aplica a las siguientes subconsultas: `selectHosts` - los resultados se ordenarán por `host`.
sortfield	string/array	Ordenar el resultado por las propiedades indicadas. Valores posibles: `groupid`, `name`.
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	
selectGroupDiscoveries	query	Devolver una propiedad `groupDiscoveries` con los objetos de descubrimiento de grupo de hosts. Cada objeto de descubrimiento de grupo de hosts es un prototipo de grupo de hosts vinculado al grupo de hosts descubierto. Esta consulta está **obsoleta**, por favor utilice `selectDiscoveryData` en su lugar.

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`

Ejemplos

Recuperar datos por nombre

Recuperar todos los datos sobre dos grupos de hosts llamados ”Zabbix servers” y ”Linux servers”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "hostgroup.get",
    "params": {
        "output": "extend",
        "filter": {
            "name": [
                "Zabbix servers",
                "Linux servers"
            ]
        }
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "groupid": "2",
            "name": "Linux servers",
            "flags": "0",
            "uuid": "dc579cd7a1a34222933f24f52a68bcd8"
        },
        {
            "groupid": "4",
            "name": "Zabbix servers",
            "flags": "0",
            "uuid": "6f6799aa69e844b4b3918f779f2abf08"
        }
    ],
    "id": 1
}
```

Véase también

- Equipo

Fuente

CHostGroup::get() en *ui/include/classes/api/services/CHostGroup.php*.

hostgroup.massadd

Descripción

`object hostgroup.massadd(object parameters)`

Este método le permite agregar simultáneamente múltiples objetos relacionados a todos los grupos de hosts especificados.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object)` Parámetros que contienen los IDs de los grupos de host a actualizar y los objetos a añadir a todos los grupos de host.

El método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
groups	object/array	Grupos de host a actualizar. Los grupos de host deben tener definida únicamente la propiedad `groupid`. Comportamiento del parámetro: - *obligatorio*
hosts	object/array	Hosts a añadir a todos los grupos de host. Los hosts deben tener definida únicamente la propiedad `hostid`.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los grupos de equipos actualizados bajo la propiedad `groupids`.

Ejemplos

Añadir equipos a grupos de equipos

Añadir dos equipos a los grupos de equipos con IDs 5 y 6.

Petición:

```
{
    "jsonrpc": "2.0",
    "method": "hostgroup.massadd",
    "params": {
        "groups": [
            {
                "groupid": "5"
            },
            {
                "groupid": "6"
            }
        ],
        "hosts": [
            {
                "hostid": "30050"
            },
            {
                "hostid": "30001"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "5",
            "6"
        ]
    },
    "id": 1
}
```

Véase también

- Equipo

Fuente

CHostGroup::massAdd() en *ui/include/classes/api/services/CHostGroup.php*.

hostgroup.massremove

Descripción

`object hostgroup.massremove(object parameters)`

Este método le permite eliminar objetos relacionados de varios grupos de hosts.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object)` Parámetros que contienen los IDs de los grupos de host a actualizar y los objetos que deben eliminarse.

Parámetro	Tipo	Descripción
groupids	ID/array	IDs de los grupos de host a actualizar. Comportamiento del parámetro: - *obligatorio*
hostids	ID/array	IDs de los hosts a eliminar de todos los grupos de host.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los grupos de equipos actualizados bajo la propiedad `groupids`.

Ejemplos

Eliminando equipos de los grupos de equipos

Elimine dos equipos de los grupos de equipos dados.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "hostgroup.massremove",
    "params": {
        "groupids": [
            "5",
            "6"
        ],
        "hostids": [
            "30050",
            "30001"
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "5",
            "6"
        ]
    },
    "id": 1
}
```

Fuente

CHostGroup::massRemove() en *ui/include/classes/api/services/CHostGroup.php*.

hostgroup.propagate

Descripción

`object hostgroup.propagate(object parameters)`

Este método permite aplicar permisos y filtros de etiquetas a todos los subgrupos de un host group.

> **Note:**
> Este método solo está disponible para tipos de usuario *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
groups	object/array	Grupos de hosts a propagar. Los grupos de hosts deben tener definida la propiedad `groupid`. Comportamiento del parámetro: - *obligatorio*
permissions	boolean	Establecer en ”true” para propagar los permisos. Comportamiento del parámetro: - *obligatorio* si `tag_filters` no está establecido
tag_filters	boolean	Establecer en ”true” para propagar los filtros de etiquetas. Comportamiento del parámetro: - *obligatorio* si `permissions` no está establecido

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los grupos de equipos propagados bajo la propiedad `groupids`.

Ejemplos

Propagando permisos de grupo de equipos y filtros de etiquetas a sus subgrupos.

Propaga los permisos de grupo de equipos y los filtros de etiquetas a sus subgrupos.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "hostgroup.propagate",
    "params": {
        "groups": [
            {
                "groupid": "6"
            }
        ],
        "permissions": true,
        "tag_filters": true
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "6",
        ]
    },
    "id": 1
}
```

Véase también

- hostgroup.update
- hostgroup.massadd
- Equipo

Fuente

CHostGroup::propagate() en *ui/include/classes/api/services/CHostGroup.php*.

hostgroup.update

Descripción

`object hostgroup.update(object/array hostGroups)`

Este método le permite actualizar grupos de hosts existentes.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object/array)` Propiedades del grupo de hosts que se actualizarán.

La propiedad `groupid` debe estar definida para cada grupo de hosts, todas las demás propiedades son opcionales. Sólo se actualizarán las propiedades indicadas, todas las demás permanecerán sin cambios.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los grupos de equipos actualizados bajo la propiedad `groupids`.

Ejemplos

Renombrar un grupo de equipos

Renombrar un grupo de equipos a ”Linux hosts”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "hostgroup.update",
    "params": {
        "groupid": "7",
        "name": "Linux hosts"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "7"
        ]
    },
```

```
    "id": 1
}
```

Renombrar varios grupos de hosts

Renombrar dos grupos de hosts para incluir información de región en sus nombres.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "hostgroup.update",
    "params": [
        {
            "groupid": "8",
            "name": "Europe/Linux hosts"
        },
        {
            "groupid": "9",
            "name": "Europe/Windows hosts"
        }
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "8",
            "9"
        ]
    },
    "id": 1
}
```

Fuente

CHostGroup::update() en *ui/include/classes/api/services/CHostGroup.php*.

Grupo de proxy

Esta clase está diseñada para trabajar con grupos de proxy.

Referencias de objetos:

- Grupo de proxy

Métodos disponibles:

- proxygroup.create - crear nuevos grupos de proxy
- proxygroup.delete - eliminar grupos de proxy
- proxygroup.get - obtener grupos de proxy
- proxygroup.update - actualizar grupos de proxy

Objeto de grupo de proxy

Los siguientes objetos están directamente relacionados con la API `proxygroup`.

Grupo de proxy

El objeto de grupo de proxy tiene las siguientes propiedades.

Property	Type	Description
proxy_groupid	ID	ID del grupo de proxy. Comportamiento de la propiedad: - *solo lectura* - *obligatoria* para las operaciones de actualización
name	string	Nombre del grupo de proxy. Comportamiento de la propiedad: - *obligatoria* para las operaciones de creación
description	text	Descripción del grupo de proxy.
failover_delay	string	Período durante el cual un proxy del grupo de proxy debe comunicarse con el server de Zabbix para considerarse en línea. Se admiten sufijos de tiempo (por ejemplo, 30s, 1m). Se admiten macros de usuario. Valores posibles: 10s-15m. Valor predeterminado: 1m.
min_online	string	Número mínimo de proxies en línea necesarios para mantener el grupo de proxy en línea. Se admiten macros de usuario. Rango de valores posibles: 1-1000. Valor predeterminado: 1.
state	integer	Estado del grupo de proxy. Valores posibles: 0 - Desconocido; 1 - Fuera de línea; 2 - En recuperación; 3 - En línea; 4 - En degradación. Comportamiento de la propiedad: - *solo lectura*

proxygroup.create

Descripción

`object proxygroup.create(object/array proxyGroups)`

Este método permite crear nuevos grupos de proxy.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object/array)` Grupos de proxy a crear.

El método acepta grupos de proxy con las propiedades estándar de grupo de proxy.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los grupos de proxy creados bajo la propiedad `proxy_groupids`. El orden de los IDs devueltos coincide con el orden de los grupos de proxy pasados.

Ejemplos

Crear un grupo de proxies

Cree un grupo de proxies con configuraciones personalizadas.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "proxygroup.create",
    "params": {
        "name": "Proxy group",
        "failover_delay": "5m",
        "min_online": "10"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxy_groupids": [
            "5"
        ]
    },
    "id": 1
}
```

Fuente

CProxyGroup::create() en *ui/include/classes/api/services/CProxyGroup.php*.

proxygroup.delete

Descripción

`object proxygroup.delete(array proxyGroupIds)`

Este método permite eliminar grupos de proxy.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(array)` IDs de los grupos de proxy a eliminar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los grupos de proxy eliminados bajo la propiedad `proxy_groupids`.

Ejemplos

Eliminar varios grupos de proxies

Eliminar dos grupos de proxies.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "proxygroup.delete",
    "params": [
        "5",
        "10"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxy_groupids": [
            "5",
            "10"
        ]
    },
    "id": 1
}
```

Fuente

CProxyGroup::delete() en *ui/include/classes/api/services/CProxyGroup.php*.

proxygroup.get

Descripción

`integer/array proxygroup.get(object parameters)`

El método permite recuperar grupos de proxy según los parámetros especificados.

Note:
Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método se pueden revocar en la configuración del rol de usuario. Consulte Roles de usuario para obtener más información. Solo se devuelven los grupos de proxy permitidos por la *Lista de acceso de proxy* de los grupos de usuarios del usuario.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
proxy_groupids	ID/array	Devolver solo los grupos de proxy con los IDs dados.
proxyids	ID/array	Devolver solo los grupos de proxy que contienen los proxies dados.
selectProxies	query	Devolver una propiedad `proxies` con los proxies que pertenecen al grupo de proxy. Admite `count`.
sortfield	string/array	Ordenar el resultado por las propiedades dadas. Valores posibles: `proxy_groupid`, `name`.
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperar todos los grupos de proxies

Recupera todos los grupos de proxies configurados con proxies.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "proxygroup.get",
    "params": {
        "output": "extend",
        "selectProxies": ["proxyid", "name"]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "proxy_groupid": "1",
            "name": "Proxy group 1",
            "failover_delay": "1m",
            "min_online": "3",
            "description": "",
            "state": "1",
            "proxies": [
                {
                    "proxyid": "1",
                    "name": "proxy 1"
                },
                {
                    "proxyid": "2",
                    "name": "proxy 2"
                }
            ]
        },
        {
            "proxy_groupid": "2",
            "name": "Proxy group 2",
            "failover_delay": "10m",
            "min_online": "3",
            "description": "",
            "state": "3",
            "proxies": [
                {
                    "proxyid": "3",
                    "name": "proxy 3"
                },
                {
                    "proxyid": "4",
                    "name": "proxy 4"
                },
                {
                    "proxyid": "5",
                    "name": "proxy 5"
                }
            ]
        }
    ],
    "id": 1
}
```

Véase también

- Proxy

Fuente

CProxyGroup::get() en *ui/include/classes/api/services/CProxyGroup.php*.

proxygroup.update

Descripción

`object proxygroup.update(object/array proxyGroups)`

Este método permite actualizar grupos de proxy existentes.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object/array)` Propiedades del grupo de proxy que se actualizarán.

La propiedad `proxy_groupid` debe definirse para cada grupo de proxy, todas las demás propiedades son opcionales. Sólo se actualizarán las propiedades pasadas, todas las demás permanecerán sin cambios.

El método acepta grupos de proxy con las propiedades estándar del grupo de proxy.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los grupos de proxy actualizados bajo la propiedad `proxy_groupids`.

Ejemplos

Cambiar el número mínimo de proxies en línea

Cambie el número mínimo de proxies en línea requeridos para que el grupo esté en línea.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "proxygroup.update",
    "params": {
        "proxy_groupid": "5",
        "min_online": "3"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxy_groupids": [
            "5"
        ]
    },
    "id": 1
}
```

Fuente

CProxyGroup::update() en *ui/include/classes/api/services/CProxyGroup.php*.

Grupo de templates

Esta clase está diseñada para trabajar con grupos de templates.

Referencias de objetos:

- Grupo de templates

Métodos disponibles:

- templategroup.create - crear nuevos grupos de templates
- templategroup.delete - eliminar grupos de templates
- templategroup.get - obtener grupos de templates
- templategroup.massadd - añadir objetos relacionados a grupos de templates
- templategroup.massremove - eliminar objetos relacionados de grupos de templates
- templategroup.propagate - propagar permisos a subgrupos de grupos de templates
- templategroup.update - actualizar grupos de templates

Objeto de grupo de plantillas

Los siguientes objetos están directamente relacionados con la API de `templategroup`.

Grupo de template

El objeto grupo de template tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
groupid	ID	ID del grupo de template. Comportamiento de la propiedad: - *solo lectura* - *requerido* para operaciones de actualización
name	string	Nombre del grupo de template. Comportamiento de la propiedad: - *requerido* para operaciones de creación
uuid	string	Identificador único universal, utilizado para vincular grupos de template importados con los ya existentes. Se genera automáticamente si no se proporciona.

templategroup.create

Descripción

`object templategroup.create(object/array templateGroups)`

Este método le permite crear nuevos grupos de templates.

Note:
Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object/array)` Grupos de templates a crear. El método acepta grupos de templates con las propiedades estándar de grupo de templates.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los grupos de plantillas creados bajo la propiedad `groupids`. El orden de los IDs devueltos coincide con el orden de los grupos de plantillas pasados.

Ejemplos

Creando un grupo de plantillas

Cree un grupo de plantillas llamado "Templates/Databases".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "templategroup.create",
    "params": {
        "name": "Templates/Databases"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "107820"
        ]
    },
    "id": 1
}
```

Fuente

CTemplateGroup::create() en *ui/include/classes/api/services/CTemplateGroup.php*.

templategroup.delete

Descripción

`object templategroup.delete(array templateGroupIds)`

Este método le permite eliminar grupos de templates.

No se puede eliminar un grupo de templates si contiene templates que pertenecen solo a este grupo.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(array)` IDs de los grupos de plantillas a eliminar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los grupos de plantillas eliminados bajo la propiedad `groupids`.

Ejemplos

Eliminando múltiples grupos de plantillas

Elimine dos grupos de plantillas.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "templategroup.delete",
    "params": [
        "107814",
        "107815"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
```

```
    "result": {
        "groupids": [
            "107814",
            "107815"
        ]
    },
    "id": 1
}
```

Fuente

CTemplateGroup::delete() en *ui/include/classes/api/services/CTemplateGroup.php*.

templategroup.get

Descripción

`integer/array templategroup.get(object parameters)`

El método permite recuperar grupos de templates según los parámetros proporcionados.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
graphids	ID/array	Devolver solo los grupos de templates que contienen templates con los gráficos indicados.
groupids	ID/array	Devolver solo los grupos de templates con los IDs de grupo de templates indicados.
templateids	ID/array	Devolver solo los grupos de templates que contienen los templates indicados.
triggerids	ID/array	Devolver solo los grupos de templates que contienen templates con los triggers indicados.
with_graphs	boolean	Devolver solo los grupos de templates que contienen templates con gráficos.
with_graph_prototypes	boolean	Devolver solo los grupos de templates que contienen templates con prototipos de gráficos.
with_httptests	boolean	Devolver solo los grupos de templates que contienen templates con comprobaciones web.
with_items	boolean	Devolver solo los grupos de templates que contienen templates con items. Sobrescribe los parámetros `with_simple_graph_items`.
with_item_prototypes	boolean	Devolver solo los grupos de templates que contienen templates con prototipos de items. Sobrescribe el parámetro `with_simple_graph_item_prototypes`.
with_simple_graph_item_prototypes	boolean	Devolver solo los grupos de templates que contienen templates con prototipos de items, que están habilitados para la creación y tienen tipo de información numérica.
with_simple_graph_items	boolean	Devolver solo los grupos de templates que contienen templates con items numéricos.
with_templates	boolean	Devolver solo los grupos de templates que contienen templates.
with_triggers	boolean	Devolver solo los grupos de templates que contienen templates con triggers.

Parámetro	Tipo	Descripción
selectTemplates	query	Devolver una propiedad `templates` con los templates que pertenecen al grupo de templates. Admite `count`.
limitSelects	integer	Limita el número de registros devueltos por las subselecciones. Se aplica a las siguientes subselecciones: `selectTemplates` - los resultados se ordenarán por `template`.
sortfield	string/array	Ordenar el resultado por las propiedades indicadas. Valores posibles: `groupid`, `name`.
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperar datos por nombre

Recuperar todos los datos sobre dos grupos de plantillas llamados ”Templates/Databases” y ”Templates/Modules”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "templategroup.get",
    "params": {
        "output": "extend",
        "filter": {
            "name": [
                "Templates/Databases",
                "Templates/Modules"
            ]
        }
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "groupid": "13",
            "name": "Templates/Databases",
            "uuid": "748ad4d098d447d492bb935c907f652f"
        },
        {
```

```
            "groupid": "8",
            "name": "Templates/Modules",
            "uuid": "57b7ae836ca64446ba2c296389c009b7"
        }
    ],
    "id": 1
}
```

Véase también

- Plantilla

Origen

CTemplateGroup::get() en *ui/include/classes/api/services/CTemplateGroup.php*.

templategroup.massadd

Descripción

`object templategroup.massadd(object parameters)`

Este método permite agregar simultáneamente varios objetos relacionados a todos los grupos de templates especificados.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object)` Parámetros que contienen los IDs de los grupos de templates a actualizar y los objetos a añadir a todos los grupos de templates.

El método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
groups	object/array	Grupos de templates a actualizar. Los grupos de templates deben tener definida únicamente la propiedad `groupid`. Comportamiento del parámetro: - *obligatorio*
templates	object/array	Templates a añadir a todos los grupos de templates. Los templates deben tener definida únicamente la propiedad `templateid`. Comportamiento del parámetro: - *obligatorio*

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los grupos de plantillas actualizados bajo la propiedad `groupids`.

Ejemplos

Añadir plantillas a grupos de plantillas

Añadir dos plantillas a los grupos de plantillas con IDs 12 y 13.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "templategroup.massadd",
    "params": {
```

```
        "groups": [
            {
                "groupid": "12"
            },
            {
                "groupid": "13"
            }
        ],
        "templates": [
            {
                "templateid": "10486"
            },
            {
                "templateid": "10487"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "12",
            "13"
        ]
    },
    "id": 1
}
```

Véase también

- Plantilla

Origen

CTemplateGroup::massAdd() en *ui/include/classes/api/services/CTemplateGroup.php*.

templategroup.massremove

Descripción

`object templategroup.massremove(object parameters)`

Este método permite eliminar objetos relacionados de varios grupos de templates.

Note:
Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object)` Parámetros que contienen los IDs de los grupos de templates a actualizar y los objetos que deben ser eliminados.

Parámetro	Tipo	Descripción
groupids	ID/array	IDs de los grupos de templates que se actualizarán. Comportamiento del parámetro: - *obligatorio*

Parámetro	Tipo	Descripción
templateids	ID/array	IDs de los templates que se eliminarán de todos los grupos de templates. Comportamiento del parámetro: - *obligatorio*

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los grupos de plantillas actualizados bajo la propiedad `groupids`.

Ejemplos

Eliminando plantillas de grupos de plantillas

Elimine dos plantillas de los grupos de plantillas dados.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "templategroup.massremove",
    "params": {
        "groupids": [
            "5",
            "6"
        ],
        "templateids": [
            "30050",
            "30001"
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "5",
            "6"
        ]
    },
    "id": 1
}
```

Origen

CTemplateGroup::massRemove() en *ui/include/classes/api/services/CTemplateGroup.php*.

templategroup.propagate

Descripción

`object templategroup.propagate(object parameters)`

Este método permite aplicar permisos a todos los subgrupos de los grupos de template.

> **Note:**
> Este método solo está disponible para tipos de usuario *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
groups	object/array	Grupos de templates a propagar. Los grupos de templates deben tener definida únicamente la propiedad `groupid`. Comportamiento del parámetro: - *obligatorio*
permissions	boolean	Establecer en `true` si es necesario propagar los permisos. Comportamiento del parámetro: - *obligatorio*

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los grupos de plantilla propagados bajo la propiedad `groupids`.

Ejemplos

Propagación de permisos de grupo de plantillas a sus subgrupos.

Propagar los permisos del grupo de plantillas a sus subgrupos.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "templategroup.propagate",
    "params": {
        "groups": [
            {
                "groupid": "15"
            }
        ],
        "permissions": true
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "15",
        ]
    },
    "id": 1
}
```

Véase también

- templategroup.update
- templategroup.massadd
- Plantilla

Fuente

CTemplateGroup::propagate() en *ui/include/classes/api/services/CTemplateGroup.php*.

templategroup.update

Descripción

`object templategroup.update(object/array templateGroups)`

Este método le permite actualizar grupos de templates existentes.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object/array)` Propiedades del grupo de templates que se van a actualizar.

La propiedad `groupid` debe estar definida para cada grupo de templates, todas las demás propiedades son opcionales. Sólo se actualizarán las propiedades indicadas, todas las demás permanecerán sin cambios.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los grupos de plantillas actualizados bajo la propiedad `groupids`.

Ejemplos

Renombrar un grupo de plantillas

Renombrar un grupo de plantillas a "Templates/Databases"

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "templategroup.update",
    "params": {
        "groupid": "7",
        "name": "Templates/Databases"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "7"
        ]
    },
    "id": 1
}
```

Fuente

CTemplateGroup::update() en *ui/include/classes/api/services/CTemplateGroup.php*.

Grupo de usuarios

Esta clase está diseñada para trabajar con grupos de usuarios.

Referencias de objetos:

- Grupo de usuarios
- Permiso
- Permiso basado en etiquetas

Métodos disponibles:

- usergroup.create - crear nuevos grupos de usuarios
- usergroup.delete - eliminar grupos de usuarios
- usergroup.get - obtener grupos de usuarios
- usergroup.update - actualizar grupos de usuarios

Objeto de grupo de usuarios

Los siguientes objetos están directamente relacionados con la API de `usergroup`.

Grupo de usuarios

El objeto de grupo de usuarios tiene las siguientes propiedades.

Property	Type	Description
usrgrpid	ID	ID del grupo de usuarios. Comportamiento de la propiedad: - *solo lectura* - *obligatoria* para las operaciones de actualización
name	string	Nombre del grupo de usuarios. Comportamiento de la propiedad: - *obligatoria* para las operaciones de creación
debug_mode	integer	Indica si el modo de depuración está habilitado o deshabilitado. Valores posibles: 0 - *(predeterminado)* deshabilitado; 1 - habilitado.
gui_access	integer	Método de autenticación del frontend para los usuarios del grupo. Valores posibles: 0 - *(predeterminado)* usar el método de autenticación predeterminado del sistema; 1 - usar la autenticación interna; 2 - usar la autenticación LDAP; 3 - deshabilitar el acceso al frontend.
mfa_status	integer	Indica si MFA está habilitado o deshabilitado para los usuarios del grupo. Valores posibles: 0 - deshabilitado (para todos los métodos MFA configurados); 1 - habilitado (para todos los métodos MFA configurados).
mfaid	ID	Método MFA utilizado para los usuarios del grupo. Comportamiento de la propiedad: - *compatible* con usuarios de tipo *Super admin* - *compatible* si `mfa_status` del objeto de autenticación está establecido en ”Habilitado”
proxy_mode	integer	Modo para gestionar el acceso a los proxies que no están incluidos en grupos de proxies. Valores posibles: 0 - Lista de denegación; 1 - *(predeterminado)* Lista de permitidos. Comportamiento de la propiedad: - *compatible* con usuarios de tipo *Super admin*.
proxy_group_mode	integer	Modo para gestionar el acceso a los grupos de proxies (y sus proxies). Valores posibles: 0 - Lista de denegación; 1 - *(predeterminado)* Lista de permitidos. Comportamiento de la propiedad: - *compatible* con usuarios de tipo *Super admin*.

Property	Type	Description
users_status	integer	Indica si el grupo de usuarios está habilitado o deshabilitado. Para los usuarios desaprovisionados, el grupo de usuarios no puede habilitarse. Valores posibles: 0 - *(predeterminado)* habilitado; 1 - deshabilitado.
userdirectoryid	ID	ID del directorio de usuarios utilizado para la autenticación. Comportamiento de la propiedad: - *compatible* con usuarios de tipo *Super admin* - *compatible* si `gui_access` está establecido en "usar el método de autenticación predeterminado del sistema" o "usar la autenticación LDAP"

Permiso

El objeto de permiso tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
id	ID	ID del grupo de hosts o grupo de templates al que añadir el permiso. Comportamiento de la propiedad: - *obligatorio*
permission	integer	Nivel de acceso al grupo de hosts o grupo de templates. Valores posibles: 0 - acceso denegado; 2 - acceso de solo lectura; 3 - acceso de lectura y escritura. Comportamiento de la propiedad: - *obligatorio* para operaciones de creación

Permiso basado en etiquetas

El objeto de permiso basado en etiquetas tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
groupid	ID	ID del grupo de hosts al que añadir el permiso. Comportamiento de la propiedad: - *obligatorio*
tag	string	Nombre de la etiqueta.
value	string	Valor de la etiqueta.

APM-based permission

The APM-based permission object has the following properties.

Property	Type	Description
key	string	APM resource attribute key. Property behavior: - *required*
value	string	APM resource attribute value.

Property	Type	Description
signal_type	integer	APM resource attribute category. Possible values: -1 - *(default)* All; 0 - Traces; 1 - Metrics; 2 - Logs.

usergroup.create

Descripción

`object usergroup.create(object/array userGroups)`

Este método permite crear nuevos grupos de usuarios.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object/array)` Grupos de usuarios que se crearán.

Además de las propiedades estándar de los grupos de usuarios, el método acepta los siguientes parámetros.

Parameter	Type	Description
hostgroup_rights	object/array	Permisos de los grupos de hosts que se asignarán al grupo de usuarios.
proxies	object/array	Proxies que se vincularán al grupo de usuarios. Los proxies deben tener definida únicamente la propiedad `proxyid`.
proxy_groups	object/array	Grupos de proxies que se vincularán al grupo de usuarios. Los grupos de proxies deben tener definida únicamente la propiedad `proxy_groupid`.
templategroup_rights	object/array	Permisos de los grupos de templates que se asignarán al grupo de usuarios.
tag_filters	array	Permisos basados en etiquetas que se asignarán al grupo de usuarios.
users	object/array	Usuarios que se añadirán al grupo de usuarios. El usuario debe tener definida únicamente la propiedad `userid`.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los grupos de usuarios creados bajo la propiedad `usrgrpids`. El orden de los IDs devueltos coincide con el orden de los grupos de usuarios pasados.

Ejemplos

Creando un grupo de usuarios

Cree un grupo de usuarios *Operation managers* con acceso denegado al grupo de equipos "2" y añada un usuario al mismo.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "usergroup.create",
    "params": {
        "name": "Operation managers",
        "hostgroup_rights": {
            "id": "2",
            "permission": 0
        },
        "users": [
```

```
            {
                "userid": "12"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "usrgrpids": [
            "20"
        ]
    },
    "id": 1
}
```

Véase también

- Permiso

Fuente

CUserGroup::create() en *ui/include/classes/api/services/CUserGroup.php*.

usergroup.delete

Descripción

`object usergroup.delete(array userGroupIds)`

Este método le permite eliminar grupos de usuarios.

Attention:
El grupo de usuarios Deprovisioned (el grupo de usuarios especificado para `disabled_usrgrpid` en Authentication) no se puede eliminar.

Note:
Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración del rol de usuario. Consulte User roles para obtener más información.

Parámetros

`(array)` IDs de los grupos de usuarios a eliminar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los grupos de usuarios eliminados bajo la propiedad `usrgrpids`.

Ejemplos

Eliminando varios grupos de usuarios

Eliminar dos grupos de usuarios.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "usergroup.delete",
    "params": [
        "20",
        "21"
    ],
```

```
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "usrgrpids": [
            "20",
            "21"
        ]
    },
    "id": 1
}
```

Fuente

CUserGroup::delete() en *ui/include/classes/api/services/CUserGroup.php*.

usergroup.get

Descripción

`integer/array usergroup.get(object parameters)`

El método permite recuperar grupos de usuarios según los parámetros proporcionados.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object)` Parámetros que definen el resultado deseado.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
mfaids	ID/array	Devuelve únicamente los grupos de usuarios con los métodos MFA indicados. Comportamiento del parámetro: - *compatible* con usuarios de tipo *Super admin*
mfa_status	integer	Devuelve únicamente los grupos de usuarios con el estado MFA indicado. Consulte la página de grupos de usuarios para ver la lista de estados compatibles.
proxyids	ID/array	Devuelve únicamente los grupos de usuarios que contienen los proxies indicados. Comportamiento del parámetro: - *compatible* con usuarios de tipo *Super admin*.
proxy_groupids	ID/array	Devuelve únicamente los grupos de usuarios que contienen los grupos de proxies indicados. Comportamiento del parámetro: - *compatible* con usuarios de tipo *Super admin*.
selectProxies	ID/array	Devuelve una propiedad `proxies` con los proxies del grupo de usuarios. Comportamiento del parámetro: - *compatible* con usuarios de tipo *Super admin*.

Parámetro	Tipo	Descripción
selectProxyGroups	ID/array	Devuelve una propiedad `proxy_groups` con los grupos de proxies del grupo de usuarios. Comportamiento del parámetro: - *compatible* con usuarios de tipo *Super admin*.
status	integer	Devuelve únicamente los grupos de usuarios con el estado indicado. Consulte la página de grupos de usuarios para ver la lista de estados compatibles.
userids	ID/array	Devuelve únicamente los grupos de usuarios que contienen a los usuarios indicados.
usrgrpids	ID/array	Devuelve únicamente los grupos de usuarios con los ID indicados.
selectTagFilters	query	Devuelve los permisos basados en etiquetas del grupo de usuarios en la propiedad `tag_filters`.
selectUsers	query	Devuelve los usuarios del grupo de usuarios en la propiedad `users`. Consulte `user.get` para conocer las restricciones según el tipo de usuario.
selectHostGroupRights	query	Devuelve los permisos del grupo de usuarios sobre los grupos de hosts en la propiedad `hostgroup_rights`. Consulte la página de grupos de usuarios para ver la lista de niveles de acceso a los grupos de hosts.
selectTemplateGroupRights	query	Devuelve los permisos del grupo de usuarios sobre los grupos de templates en la propiedad `templategroup_rights`. Consulte la página de grupos de usuarios para ver la lista de niveles de acceso a los grupos de templates.
limitSelects	integer	Limita el número de registros devueltos por las subselecciones.
output	query	Propiedades del objeto de grupo de usuarios que se devolverán. Los usuarios de tipo *Admin* y *User* solo pueden recuperar las siguientes propiedades: `usrgrpid`, `name`, `gui_access`, `users_status`, `debug_mode`, `mfa_status`. Predeterminado: `extend`.
sortfield	string/array	Ordena el resultado por las propiedades indicadas. Valores posibles: `usrgrpid`, `name`.
countOutput	boolean	Estos parámetros se describen en los comentarios de referencia.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperar grupos de usuarios habilitados

Recuperar todos los grupos de usuarios habilitados.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "usergroup.get",
    "params": {
        "output": "extend",
        "status": 0
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "usrgrpid": "7",
            "name": "Zabbix administrators",
            "gui_access": "0",
            "users_status": "0",
            "debug_mode": "1",
            "userdirectoryid": "0",
            "mfa_status": "0",
            "mfaid": "0"
        },
        {
            "usrgrpid": "8",
            "name": "Guests",
            "gui_access": "0",
            "users_status": "0",
            "debug_mode": "0",
            "userdirectoryid": "0",
            "mfa_status": "0",
            "mfaid": "0"
        },
        {
            "usrgrpid": "11",
            "name": "Enabled debug mode",
            "gui_access": "0",
            "users_status": "0",
            "debug_mode": "1",
            "userdirectoryid": "0",
            "mfa_status": "0",
            "mfaid": "0"
        },
        {
            "usrgrpid": "12",
            "name": "No access to the frontend",
            "gui_access": "2",
            "users_status": "0",
            "debug_mode": "0",
            "userdirectoryid": "0",
            "mfa_status": "0",
            "mfaid": "0"
        },
        {
            "usrgrpid": "14",
            "name": "Read only",
            "gui_access": "0",
            "users_status": "0",
            "debug_mode": "0",
            "userdirectoryid": "0",
```

```
            "mfa_status": "0",
            "mfaid": "0"
        },
        {
            "usrgrpid": "18",
            "name": "Deny",
            "gui_access": "0",
            "users_status": "0",
            "debug_mode": "0",
            "userdirectoryid": "0",
            "mfa_status": "0",
            "mfaid": "0"
        }
    ],
    "id": 1
}
```


Véase también

- Usuario

Fuente

CUserGroup::get() en *ui/include/classes/api/services/CUserGroup.php*.

usergroup.update

Descripción

`object usergroup.update(object/array userGroups)`

Este método le permite actualizar grupos de usuarios existentes.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración del rol de usuario. Consulte User roles para obtener más información.

Parámetros

`(object/array)` Propiedades del grupo de usuarios que se actualizarán.

La propiedad `usrgrpid` debe definirse para cada grupo de usuarios; todas las demás propiedades son opcionales. Solo se actualizarán las propiedades proporcionadas; las demás permanecerán sin cambios.

Además de las propiedades estándar del grupo de usuarios, el método acepta los siguientes parámetros.

Parameter	Type	Description
hostgroup_rights	object/array	Permisos de los grupos de hosts que reemplazarán los permisos actuales asignados al grupo de usuarios.
proxies	object/array	Proxies que se vincularán al grupo de usuarios. Los proxies solo deben tener definida la propiedad `proxyid`.
proxy_groups	object/array	Grupos de proxies que se vincularán al grupo de usuarios. Los grupos de proxies solo deben tener definida la propiedad `proxy_groupid`.
templategroup_rights	object/array	Permisos de los grupos de templates que reemplazarán los permisos actuales asignados al grupo de usuarios.
tag_filters	array	Permisos basados en etiquetas que reemplazarán los permisos actuales asignados al grupo de usuarios.
users	object/array	Usuarios que reemplazarán a los usuarios actuales asignados al grupo de usuarios. El usuario solo debe tener definida la propiedad `userid`.

Valores de retorno

(object) Devuelve un objeto que contiene los IDs de los grupos de usuarios actualizados bajo la propiedad usrgrpids.

Ejemplos

Habilitar un grupo de usuarios y actualizar permisos

Habilite un grupo de usuarios y proporcione acceso de lectura-escritura para él a los grupos de equipos ”2” y ”4”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "usergroup.update",
    "params": {
        "usrgrpid": "17",
        "users_status": "0",
        "hostgroup_rights": [
            {
                "id": "2",
                "permission": 3
            },
            {
                "id": "4",
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "usrgrpids": [
            "17"
        ]
    },
    "id": 1
}
```

Véase también

- Permiso

Fuente

CUserGroup::update() en *ui/include/classes/api/services/CUserGroup.php*.

Gráfico

Esta clase está diseñada para trabajar con gráficos.

Referencias de objetos:

- Gráfico

Métodos disponibles:

- graph.create - crear nuevos gráficos
- graph.delete - eliminar gráficos
- graph.get - obtener gráficos
- graph.update - actualizar gráficos

Objeto gráfico

Los siguientes objetos están directamente relacionados con la API de graph.

Gráfico

El objeto gráfico tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
graphid	ID	ID del gráfico. Comportamiento de la propiedad: - *solo lectura* - *requerido* para operaciones de actualización
height	integer	Altura del gráfico en píxeles. Comportamiento de la propiedad: - *requerido* para operaciones de creación
name	string	Nombre del gráfico. Comportamiento de la propiedad: - *requerido* para operaciones de creación
width	integer	Ancho del gráfico en píxeles. Comportamiento de la propiedad: - *requerido* para operaciones de creación
flags	integer	Origen del gráfico. Valores posibles: 0 - *(por defecto)* un gráfico simple; 4 - un gráfico convertido desde un prototipo. Comportamiento de la propiedad: - *solo lectura*
graphtype	integer	Tipo de diseño del gráfico. Valores posibles: 0 - *(por defecto)* normal; 1 - apilado; 2 - circular; 3 - circular expandido.
percent_left	float	Percentil izquierdo. Por defecto: 0.
percent_right	float	Percentil derecho. Por defecto: 0.
show_3d	integer	Si se muestran los gráficos circulares y circulares expandidos en 3D. Valores posibles: 0 - *(por defecto)* mostrar en 2D; 1 - mostrar en 3D.
show_legend	integer	Si se muestra la leyenda en el gráfico. Valores posibles: 0 - ocultar; 1 - *(por defecto)* mostrar.
show_work_period	integer	Si se muestra el tiempo de trabajo en el gráfico. Valores posibles: 0 - ocultar; 1 - *(por defecto)* mostrar.
show_triggers	integer	Si se muestra la línea de trigger en el gráfico. Valores posibles: 0 - ocultar; 1 - *(por defecto)* mostrar.

Propiedad	Tipo	Descripción
templateid	ID	ID del gráfico de la template padre. Comportamiento de la propiedad: - *solo lectura*
yaxismax	float	El valor máximo fijo para el eje Y. Por defecto: 100.
yaxismin	float	El valor mínimo fijo para el eje Y. Por defecto: 0.
ymax_itemid	ID	ID del item que se utiliza como valor máximo para el eje Y. Si un usuario no tiene acceso al item especificado, el gráfico se representa como si `ymax_type` estuviera establecido en ”calculado”.
ymax_type	integer	Método de cálculo del valor máximo para el eje Y. Valores posibles: 0 - *(por defecto)* calculado; 1 - fijo; 2 - item.
ymin_itemid	ID	ID del item que se utiliza como valor mínimo para el eje Y. Si un usuario no tiene acceso al item especificado, el gráfico se representa como si `ymin_type` estuviera establecido en ”calculado”.
ymin_type	integer	Método de cálculo del valor mínimo para el eje Y. Valores posibles: 0 - *(por defecto)* calculado; 1 - fijo; 2 - item.
uuid	string	Identificador único universal, utilizado para vincular gráficos importados con los ya existentes. Se genera automáticamente si no se proporciona. Comportamiento de la propiedad: - *soportado* si el gráfico pertenece a una template

graph.create

Descripción

`object graph.create(object/array graphs)`

Este método permite crear nuevos gráficos.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para invocar el método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(objeto/array)` Gráficas a crear.

Además de las propiedades estándar de la gráfica, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
gitems	array	Items de la gráfica que se crearán para la gráfica. Comportamiento del parámetro: - *requerido*

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los gráficos creados bajo la propiedad `graphids`. El orden de los IDs devueltos coincide con el orden de los gráficos enviados.

Ejemplos

Creando un gráfico

Cree un gráfico con dos métricas.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "graph.create",
    "params": {
        "name": "MySQL bandwidth",
        "width": 900,
        "height": 200,
        "gitems": [
            {
                "itemid": "22828",
                "color": "00AA00"
            },
            {
                "itemid": "22829",
                "color": "3333FF"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "graphids": [
            "652"
        ]
    },
    "id": 1
}
```

Ver también

- Item de gráfico

Fuente

CGraph::create() en *ui/include/classes/api/services/CGraph.php*.

graph.delete

Descripción

`object graph.delete(array graphIds)`

Este método le permite eliminar gráficos.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración del rol de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(array)` IDs de los gráficos a eliminar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los gráficos eliminados bajo la propiedad `graphids`.

Ejemplos

Eliminando varios gráficos

Eliminar dos gráficos.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "graph.delete",
    "params": [
        "652",
        "653"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "graphids": [
            "652",
            "653"
        ]
    },
    "id": 1
}
```

Fuente

CGraph::delete() en *ui/include/classes/api/services/CGraph.php*.

graph.get

Descripción

`integer/array graph.get(object parameters)`

El método permite recuperar gráficos según los parámetros proporcionados.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
graphids	ID/array	Devolver solo los gráficos con los IDs dados.
groupids	ID/array	Devolver solo los gráficos que pertenecen a hosts o templates en los grupos de hosts o grupos de templates dados.
templateids	ID/array	Devolver solo los gráficos que pertenecen a los templates dados.
hostids	ID/array	Devolver solo los gráficos que pertenecen a los hosts dados.
itemids	ID/array	Devolver solo los gráficos que contienen los items dados.
templated	boolean	Si se establece en `true`, devolver solo los gráficos que pertenecen a templates.
inherited	boolean	Si se establece en `true`, devolver solo los gráficos heredados de un template.

Parámetro	Tipo	Descripción
expandName	flag	Expandir macros en el nombre del gráfico.
selectDiscoveryData	query	Devolver una propiedad `discoveryData` con los datos del objeto de descubrimiento de gráficos. El objeto de descubrimiento de gráficos vincula un gráfico descubierto a una plantilla de gráfico de la que fue descubierto. Tiene las siguientes propiedades: `parent_graphid` - `(ID)` ID de la plantilla de gráfico de la que se ha creado el gráfico; `status` - `(int)` estado del descubrimiento del gráfico: 0 - *(por defecto)* el gráfico está descubierto, 1 - el gráfico ya no está descubierto; `ts_delete` - `(timestamp)` hora en la que se eliminará un gráfico que ya no está descubierto.
selectDiscoveryRule	query	Devolver una propiedad `discoveryRule` con la regla de descubrimiento de bajo nivel que creó el gráfico.
selectHostGroups	query	Devolver una propiedad `hostgroups` con los grupos de hosts a los que pertenece el gráfico.
selectTemplateGroups	query	Devolver una propiedad `templategroups` con los grupos de templates a los que pertenece el gráfico.
selectTemplates	query	Devolver una propiedad `templates` con los templates a los que pertenece el gráfico.
selectHosts	query	Devolver una propiedad `hosts` con los hosts a los que pertenece el gráfico.
selectItems	query	Devolver una propiedad `items` con los items utilizados en el gráfico.
selectGraphItems	query	Devolver una propiedad `gitems` con los items utilizados en el gráfico.
filter	object	Devolver solo aquellos resultados que coincidan exactamente con el filtro dado. Acepta un objeto, donde las claves son nombres de propiedades y los valores son un único valor o un array de valores con los que comparar. No admite propiedades del tipo de dato `text` data type. Admite propiedades adicionales: `host` - nombre técnico del host al que pertenece el gráfico; `hostid` - ID del host al que pertenece el gráfico.
sortfield	string/array	Ordenar el resultado por las propiedades dadas. Valores posibles: `graphid`, `name`, `graphtype`.
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
editable	boolean	
excludeSearch	boolean	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	
selectGraphDiscovery	query	Devolver una propiedad `graphDiscovery` con el objeto de descubrimiento de gráficos. Los objetos de descubrimiento de gráficos vinculan el gráfico a una plantilla de gráfico de la que fue creado. Esta consulta está **obsoleta**, por favor use `selectDiscoveryData` en su lugar.

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperar gráficos de equipos

Recuperar todos los gráficos del equipo "10107" y ordenarlos por nombre.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "graph.get",
    "params": {
        "output": "extend",
        "hostids": 10107,
        "sortfield": "name"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "graphid": "612",
            "name": "CPU jumps",
            "width": "900",
            "height": "200",
            "yaxismin": "0",
            "yaxismax": "100",
            "templateid": "439",
            "show_work_period": "1",
            "show_triggers": "1",
            "graphtype": "0",
            "show_legend": "1",
            "show_3d": "0",
            "percent_left": "0",
            "percent_right": "0",
            "ymin_type": "0",
            "ymax_type": "0",
            "ymin_itemid": "0",
            "ymax_itemid": "0",
            "flags": "0"
        },
        {
            "graphid": "613",
            "name": "Carga de CPU",
            "width": "900",
            "height": "200",
            "yaxismin": "0",
            "yaxismax": "100",
            "templateid": "433",
            "show_work_period": "1",
            "show_triggers": "1",
            "graphtype": "0",
            "show_legend": "1",
            "show_3d": "0",
            "percent_left": "0",
            "percent_right": "0",
            "ymin_type": "1",
            "ymax_type": "0",
            "ymin_itemid": "0",
```

```
            "ymax_itemid": "0",
            "flags": "0"
        },
        {
            "graphid": "614",
            "name": "Utilización de CPU",
            "width": "900",
            "height": "200",
            "yaxismin": "0",
            "yaxismax": "100",
            "templateid": "387",
            "show_work_period": "1",
            "show_triggers": "0",
            "graphtype": "1",
            "show_legend": "1",
            "show_3d": "0",
            "percent_left": "0",
            "percent_right": "0",
            "ymin_type": "1",
            "ymax_type": "1",
            "ymin_itemid": "0",
            "ymax_itemid": "0",
            "flags": "0"
        },
        {
            "graphid": "645",
            "name": "Uso de espacio en disco /",
            "width": "600",
            "height": "340",
            "yaxismin": "0",
            "yaxismax": "0",
            "templateid": "0",
            "show_work_period": "0",
            "show_triggers": "0",
            "graphtype": "2",
            "show_legend": "1",
            "show_3d": "1",
            "percent_left": "0",
            "percent_right": "0",
            "ymin_type": "0",
            "ymax_type": "0",
            "ymin_itemid": "0",
            "ymax_itemid": "0",
            "flags": "4"
        }
    ],
    "id": 1
}
```


Ver también

- Regla de descubrimiento
- Item de gráfico
- Item
- Host
- Grupo de hosts
- Template
- Grupo de templates

Fuente

CGraph::get() en *ui/include/classes/api/services/CGraph.php*.

graph.update

Descripción

`object graph.update(object/array graphs)`

Este método permite actualizar gráficos existentes.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(objeto/array)` Propiedades del gráfico a actualizar.

La propiedad `graphid` debe estar definida para cada gráfico, todas las demás propiedades son opcionales. Sólo se actualizarán las propiedades pasadas, todas las demás permanecerán sin cambios.

Además de las propiedades estándar del gráfico, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
gitems	array	Items del gráfico para reemplazar los items existentes del gráfico. Si un item del gráfico tiene definida la propiedad `gitemid`, se actualizará; de lo contrario, se creará un nuevo item del gráfico.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los gráficos actualizados bajo la propiedad `graphids`.

Ejemplos

Establecer el máximo para la escala Y

Establezca el máximo de la escala Y en un valor fijo de 100.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "graph.update",
    "params": {
        "graphid": "439",
        "ymax_type": 1,
        "yaxismax": 100
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "graphids": [
            "439"
        ]
    },
    "id": 1
}
```

Fuente

CGraph::update() en *ui/include/classes/api/services/CGraph.php*.

History

Esta clase está diseñada para trabajar con datos históricos.

Referencias de objetos:

- Historial de tipo flotante
- Historial de tipo entero
- Historial de tipo cadena
- Historial de tipo texto
- Historial de tipo log

Métodos disponibles:

- history.clear - borrar datos históricos
- history.get - recuperar datos históricos
- history.push - enviar datos históricos al servidor Zabbix

Objeto de historial

Los siguientes objetos están directamente relacionados con la API de `history`.

Note:
Los objetos de historial difieren según el tipo de información de la métrica. Son creados por el servidor Zabbix y no pueden ser modificados a través de la API.

Historial de valores flotantes

El objeto de historial de valores flotantes tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
clock	timestamp	Hora en que se recibió ese valor.
itemid	ID	ID del item relacionado.
ns	integer	Nanosegundos en que se recibió el valor.
value	float	Valor recibido.

Historial de enteros

El objeto de historial de enteros tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
clock	timestamp	Hora en que se recibió ese valor.
itemid	ID	ID del item relacionado.
ns	integer	Nanosegundos en que se recibió el valor.
value	integer	Valor recibido.

Historial de cadenas

El objeto de historial de cadenas tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
clock	timestamp	Hora en que se recibió ese valor.
itemid	ID	ID del item relacionado.
ns	integer	Nanosegundos en que se recibió el valor.
value	string	Valor recibido.

Historial de texto

El objeto de historial de texto tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
id	ID	ID de la entrada de historial.
clock	timestamp	Hora en la que se recibió ese valor.
itemid	ID	ID del item relacionado.
ns	integer	Nanosegundos en los que se recibió el valor.
value	text	Valor recibido.

Historial de logs

El objeto de historial de logs tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
id	ID	ID de la entrada de historial.
clock	timestamp	Hora en la que se recibió el valor.
itemid	ID	ID del item relacionado.
logeventid	integer	ID de la entrada del log de eventos de Windows.
ns	integer	Nanosegundos en los que se recibió el valor.
severity	integer	Nivel de la entrada del log de eventos de Windows.
source	string	Origen de la entrada del log de eventos de Windows.
timestamp	timestamp	Hora de la entrada del log de eventos de Windows.
value	text	Valor recibido.

Historial binario

El objeto de historial binario tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
clock	timestamp	Hora en la que se recibió ese valor.
itemid	ID	ID del item relacionado.
ns	integer	Nanosegundos en los que se recibió el valor.
value	text	Valor recibido.

Historial JSON

El objeto de historial JSON tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
clock	timestamp	Hora en la que se recibió ese valor.
itemid	ID	ID del item relacionado.
ns	integer	Nanosegundos en los que se recibió el valor.
value	text	Valor recibido.

history.clear

Descripción

`object history.clear(array itemids)`

Este método permite borrar el historial de item.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(array)` IDs de las métricas a limpiar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las métricas borradas bajo la propiedad `itemids`.

Ejemplos

Borrar historial

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "history.clear",
    "params": [
        "10325",
        "13205"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "10325",
            "13205"
        ]
    },
    "id": 1
}
```

Fuente

CHistory::clear() en *ui/include/classes/api/services/CHistory.php*.

history.get

Descripción

`integer/array history.get(object parameters)`

El método permite recuperar datos históricos según los parámetros proporcionados.

Attention:
Este método puede devolver datos históricos de una entidad eliminada si estos datos aún no han sido eliminados por el housekeeper.

Note:
Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parameter	Type	Description
history	integer	Tipos de objeto de historial que se devolverán. Valores posibles: 0 - flotante numérico; 1 - carácter; 2 - log; 3 - *(predeterminado)* entero sin signo numérico; 4 - texto; 5 - binario; 6 - JSON.

Parameter	Type	Description
hostids	ID/array	Devuelve solo el historial de los hosts dados.
itemids	ID/array	Devuelve solo el historial de los items dados.
maxValueSize	integer	Número máximo de bytes que se devolverán en la propiedad `value`. Rango de valores posibles: 1-134217728 (1 byte a 128MiB). Si se establece en `null`, no se aplica ningún límite. Predeterminado: 65536 (64KiB). Comportamiento del parámetro: - *compatible* si `history` se establece en ”binary” o ”JSON”
time_from	timestamp	Devuelve solo los valores que se han recibido después o en el momento indicado.
time_till	timestamp	Devuelve solo los valores que se han recibido antes o en el momento indicado.
sortfield	string/array	Ordena el resultado por las propiedades indicadas. Valores posibles: `itemid`, `clock`, `ns`. No admite ordenar por `ns` si los datos de historial se almacenan en ClickHouse.
search	object	Devuelve resultados que coinciden con el patrón indicado (sin distinción entre mayúsculas y minúsculas). Acepta un objeto, donde las claves son nombres de propiedades y los valores son cadenas de texto que se buscarán. Si no se proporcionan opciones adicionales, esto realizará una búsqueda `LIKE "%…%"`. No admite la búsqueda por `value` si `history` se establece en 6 (JSON).
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperar datos históricos de métricas

Devuelve los 10 últimos valores recibidos de una métrica numérica (flotante).

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "history.get",
    "params": {
        "output": "extend",
        "history": 0,
        "itemids": "23296",
        "sortfield": "clock",
```

```
        "sortorder": "DESC",
        "limit": 10
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "itemid": "23296",
            "clock": "1351090996",
            "value": "0.085",
            "ns": "563157632"
        },
        {
            "itemid": "23296",
            "clock": "1351090936",
            "value": "0.16",
            "ns": "549216402"
        },
        {
            "itemid": "23296",
            "clock": "1351090876",
            "value": "0.18",
            "ns": "537418114"
        },
        {
            "itemid": "23296",
            "clock": "1351090816",
            "value": "0.21",
            "ns": "522659528"
        },
        {
            "itemid": "23296",
            "clock": "1351090756",
            "value": "0.215",
            "ns": "507809457"
        },
        {
            "itemid": "23296",
            "clock": "1351090696",
            "value": "0.255",
            "ns": "495509699"
        },
        {
            "itemid": "23296",
            "clock": "1351090636",
            "value": "0.36",
            "ns": "477708209"
        },
        {
            "itemid": "23296",
            "clock": "1351090576",
            "value": "0.375",
            "ns": "463251343"
        },
        {
            "itemid": "23296",
            "clock": "1351090516",
            "value": "0.315",
```

```
            "ns": "447947017"
        },
        {
            "itemid": "23296",
            "clock": "1351090456",
            "value": "0.275",
            "ns": "435307141"
        }
    ],
    "id": 1
}
```


Fuente

CHistory::get() en *ui/include/classes/api/services/CHistory.php*.

history.push

Descripción

`object history.push(object/array itemHistoryData)`

Este método permite enviar datos históricos de métricas al servidor Zabbix.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden ser revocados en la configuración del rol de usuario. Consulte Roles de usuario para más información.

Parámetros

`(object/array)` Datos históricos del item a enviar.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
itemid	ID	ID del item relacionado. Comportamiento del parámetro: - *obligatorio* si `host` y `key` no están establecidos
host	string	Nombre técnico del host. Comportamiento del parámetro: - *obligatorio* si `itemid` no está establecido
key	string	Clave del item. Comportamiento del parámetro: - *obligatorio* si `itemid` no está establecido
value	mixed	Valor del item. Comportamiento del parámetro: - *obligatorio*
clock	timestamp	Hora en que se recibió el valor.
ns	integer	Nanosegundos en que se recibió el valor.

Valores de retorno

`(object)` Devuelve el resultado de la operación de envío de datos.

Ejemplos

Enviar datos históricos de métricas

Enviar datos históricos de métricas al servidor Zabbix para las métricas ”10600”, ”10601” y ”999999”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "history.push",
    "params": [
        {
            "itemid": 10600,
            "value": 0.5,
            "clock": 1690891294,
            "ns": 45440940
        },
        {
            "itemid": 10600,
            "value": 0.6,
            "clock": 1690891295,
            "ns": 312431
        },
        {
            "itemid": 10601,
            "value": "[Tue Aug 01 15:01:35 2023] [error] [client 1.2.3.4] File does not exist: /var/www/ht
        },
        {
            "itemid": 999999,
            "value": 123
        }
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "response": "success",
        "data": [
            {
                "itemid": "10600"
            },
            {
                "itemid": "10600"
            },
            {
                "itemid": "10601",
                "error": "La métrica está deshabilitada."
            },
            {
                "error": "Sin permisos para el objeto referido o no existe."
            }
        ]
    },
    "id": 1
}
```

Véase también

- Métricas trapper
- Métricas HTTP agent
- Equipo
- Métrica

Fuente

CHistory::push() en *ui/include/classes/api/services/CHistory.php*.

Host

Esta clase está diseñada para trabajar con hosts.

Referencias de objetos:

- Host
- Inventario de host
- Etiqueta de host

Métodos disponibles:

- host.create - crear nuevos hosts
- host.delete - eliminar hosts
- host.get - obtener hosts
- host.massadd - añadir objetos relacionados a los hosts
- host.massremove - eliminar objetos relacionados de los hosts
- host.update - actualizar hosts

Objeto equipo

Los siguientes objetos están directamente relacionados con la API de equipos.

Host

El objeto host tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
hostid	ID	ID del host. Comportamiento de la propiedad: - *solo lectura* - *obligatoria* para operaciones de actualización
host	string	Nombre técnico del host. Comportamiento de la propiedad: - *obligatoria* para operaciones de creación
description	text	Descripción del host.
flags	integer	Origen del host. Valores posibles: 0 - un host normal; 4 - un host convertido a partir de un prototipo. Comportamiento de la propiedad: - *solo lectura*
inventory_mode	integer	Modo de población del inventario del host. Valores posibles: -1 - *(predeterminado)* deshabilitado; 0 - manual; 1 - automático.
ipmi_authtype	integer	Algoritmo de autenticación IPMI. Valores posibles: -1 - *(predeterminado)* predeterminado; 0 - ninguno; 1 - MD2; 2 - MD5 4 - directo; 5 - OEM; 6 - RMCP+.
ipmi_password	string	Contraseña de IPMI.

Propiedad	Tipo	Descripción
ipmi_privilege	integer	Nivel de privilegios de IPMI. Valores posibles: 1 - devolución de llamada; 2 - *(predeterminado)* usuario; 3 - operador; 4 - administrador; 5 - OEM.
ipmi_username	string	Nombre de usuario de IPMI.
maintenance_from	timestamp	Hora de inicio del mantenimiento efectivo. Comportamiento de la propiedad: - *solo lectura*
maintenance_status	integer	Estado del mantenimiento efectivo. Valores posibles: 0 - *(predeterminado)* sin mantenimiento; 1 - mantenimiento activo. Comportamiento de la propiedad: - *solo lectura*
maintenance_type	integer	Tipo de mantenimiento efectivo. Valores posibles: 0 - *(predeterminado)* mantenimiento con recopilación de datos; 1 - mantenimiento sin recopilación de datos. Comportamiento de la propiedad: - *solo lectura*
maintenanceid	ID	ID del mantenimiento actualmente activo en el host. Comportamiento de la propiedad: - *solo lectura*
name	string	Nombre visible del host. Predeterminado: valor de la propiedad `host`.
monitored_by	integer	Origen utilizado para monitorizar el host. Valores posibles: 0 - *(predeterminado)* Zabbix server; 1 - proxy; 2 - grupo de proxies. Comportamiento de la propiedad: - *obligatoria* si `select_server_for_monitoring` está establecido en "Disabled".
proxyid	ID	ID del proxy utilizado para monitorizar el host. Comportamiento de la propiedad: - *obligatoria* si `monitored_by` está establecido en "Proxy"
proxy_groupid	ID	ID del grupo de proxies utilizado para monitorizar el host. Comportamiento de la propiedad: - *obligatoria* si `monitored_by` está establecido en "Proxy group"
status	integer	Estado y función del host. Valores posibles: 0 - *(predeterminado)* habilitado; 1 - deshabilitado.

Propiedad	Tipo	Descripción
tls_connect	integer	Conexiones al host. Valores posibles: 1 - *(predeterminado)* sin cifrado; 2 - PSK; 4 - certificado.
tls_accept	integer	Conexiones desde el host. Valores de bitmap posibles: 1 - *(predeterminado)* sin cifrado; 2 - PSK; 4 - certificado. Este es un campo de máscara de bits; se acepta cualquier suma de los valores de bitmap posibles (por ejemplo, 6 para PSK y certificado).
tls_issuer	string	Emisor del certificado.
tls_subject	string	Asunto del certificado.
tls_psk_identity	string	Identidad de PSK; debe asociarse con una sola PSK (entre autoregistro, hosts y proxies). No incluya información confidencial en la identidad de PSK, ya que se envía sin cifrar a través de la red para informar al receptor de qué PSK debe utilizar. Comportamiento de la propiedad: - *solo escritura* - *obligatoria* si `tls_connect` está establecido en ”PSK” o si `tls_accept` contiene el bit ”PSK”
tls_psk	string	Clave precompartida (PSK); debe contener al menos 32 dígitos hexadecimales. Comportamiento de la propiedad: - *solo escritura* - *obligatoria* si `tls_connect` está establecido en ”PSK” o si `tls_accept` contiene el bit ”PSK”
active_available	integer	Estado de disponibilidad de la interfaz activa del host. Valores posibles: 0 - se desconoce el estado de la interfaz; 1 - la interfaz está disponible; 2 - la interfaz no está disponible. Comportamiento de la propiedad: - *solo lectura*
assigned_proxyid	ID	ID del proxy asignado por Zabbix server, si el host está monitorizado por un grupo de proxies. Comportamiento de la propiedad: - *solo lectura*

Inventario de host

El objeto de inventario de host tiene las siguientes propiedades.

Note:
Cada propiedad tiene su propio número de ID único, que se utiliza para asociar los campos de inventario de host con los items.

ID	Propiedad	Tipo	Descripción	Longitud máxima
4	alias	string	Alias.	128 caracteres

ID	Propiedad	Tipo	Descripción	Longitud máxima
11	asset_tag	string	Etiqueta de activo.	64 caracteres
28	chassis	string	Chasis.	64 caracteres
23	contact	string	Persona de contacto.	65535 caracteres para bases de datos SQL
32	contract_number	string	Número de contrato.	64 caracteres
47	date_hw_decomm	string	Fecha de baja del hardware.	64 caracteres
46	date_hw_expiry	string	Fecha de vencimiento del mantenimiento del hardware.	64 caracteres
45	date_hw_install	string	Fecha de instalación del hardware.	64 caracteres
44	date_hw_purchase	string	Fecha de compra del hardware.	64 caracteres
34	deployment_status	string	Estado de despliegue.	64 caracteres
14	hardware	string	Hardware.	255 caracteres
15	hardware_full	string	Hardware detallado.	65535 caracteres para bases de datos SQL
39	host_netmask	string	Máscara de subred del host.	39 caracteres
38	host_networks	string	Redes del host.	65535 caracteres para bases de datos SQL
40	host_router	string	Router del host.	39 caracteres
30	hw_arch	string	Arquitectura de hardware.	32 caracteres
33	installer_name	string	Nombre del instalador.	64 caracteres
24	location	string	Ubicación.	65535 caracteres para bases de datos SQL
25	location_lat	string	Latitud de la ubicación.	16 caracteres
26	location_lon	string	Longitud de la ubicación.	16 caracteres
12	macaddress_a	string	Dirección MAC A.	64 caracteres
13	macaddress_b	string	Dirección MAC B.	64 caracteres
29	model	string	Modelo.	64 caracteres
3	name	string	Nombre.	128 caracteres
27	notes	string	Notas.	65535 caracteres para bases de datos SQL
41	oob_ip	string	Dirección IP OOB.	39 caracteres
42	oob_netmask	string	Máscara de subred OOB.	39 caracteres
43	oob_router	string	Router OOB.	39 caracteres
5	os	string	Nombre del sistema operativo.	128 caracteres
6	os_full	string	Nombre detallado del sistema operativo.	255 caracteres
7	os_short	string	Nombre corto del sistema operativo.	128 caracteres
61	poc_1_cell	string	Número móvil del POC primario.	64 caracteres
58	poc_1_email	string	Correo electrónico primario.	128 caracteres
57	poc_1_name	string	Nombre del POC primario.	128 caracteres
63	poc_1_notes	string	Notas del POC primario.	65535 caracteres para bases de datos SQL
59	poc_1_phone_a	string	Teléfono A del POC primario.	64 caracteres
60	poc_1_phone_b	string	Teléfono B del POC primario.	64 caracteres
62	poc_1_screen	string	Nombre de pantalla del POC primario.	64 caracteres
68	poc_2_cell	string	Número móvil del POC secundario.	64 caracteres
65	poc_2_email	string	Correo electrónico secundario del POC.	128 caracteres
64	poc_2_name	string	Nombre del POC secundario.	128 caracteres
70	poc_2_notes	string	Notas del POC secundario.	65535 caracteres para bases de datos SQL
66	poc_2_phone_a	string	Teléfono A del POC secundario.	64 caracteres
67	poc_2_phone_b	string	Teléfono B del POC secundario.	64 caracteres
69	poc_2_screen	string	Nombre de pantalla del POC secundario.	64 caracteres
8	serialno_a	string	Número de serie A.	64 caracteres
9	serialno_b	string	Número de serie B.	64 caracteres
48	site_address_a	string	Dirección del sitio A.	128 caracteres
49	site_address_b	string	Dirección del sitio B.	128 caracteres
50	site_address_c	string	Dirección del sitio C.	128 caracteres
51	site_city	string	Ciudad del sitio.	128 caracteres
53	site_country	string	País del sitio.	64 caracteres
56	site_notes	string	Notas del sitio.	65535 caracteres para bases de datos SQL

ID	Propiedad	Tipo	Descripción	Longitud máxima
55	site_rack	string	Ubicación del rack del sitio.	128 caracteres
52	site_state	string	Estado del sitio.	64 caracteres
54	site_zip	string	Código postal del sitio.	64 caracteres
16	software	string	Software.	255 caracteres
18	software_app_a	string	Aplicación de software A.	64 caracteres
19	software_app_b	string	Aplicación de software B.	64 caracteres
20	software_app_c	string	Aplicación de software C.	64 caracteres
21	software_app_d	string	Aplicación de software D.	64 caracteres
22	software_app_e	string	Aplicación de software E.	64 caracteres
17	software_full	string	Detalles del software.	65535 caracteres para bases de datos SQL
10	tag	string	Etiqueta.	64 caracteres
1	type	string	Tipo.	64 caracteres
2	type_full	string	Detalles del tipo.	64 caracteres
35	url_a	string	URL A.	2048 caracteres
36	url_b	string	URL B.	2048 caracteres
37	url_c	string	URL C.	2048 caracteres
31	vendor	string	Proveedor.	64 caracteres

Etiqueta de host

El objeto de etiqueta de host tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
tag	string	Nombre de la etiqueta de host. Comportamiento de la propiedad: - *requerido*
value	string	Valor de la etiqueta de host. Comportamiento de la propiedad: - *solo lectura*.
automatic	integer	Tipo de etiqueta de host. Valores posibles: 0 - *(por defecto)* manual (etiqueta creada por el usuario); 1 - automática (etiqueta creada por el descubrimiento de bajo nivel) Comportamiento de la propiedad: - *solo lectura*
object	integer	Tipo de objeto del que se heredó la etiqueta. Valores posibles: 0 - template. Comportamiento de la propiedad: - *solo lectura*.
objectid	ID	ID del objeto del que se heredó la etiqueta. Comportamiento de la propiedad: - *solo lectura*.

host.create

Descripción

`object host.create(object/array hosts)`

Este método le permite crear nuevos hosts.

Note:
Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(objeto/array)` Hosts a crear.

Además de las propiedades estándar de host, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
groups	objeto/array	Grupos de host a los que añadir el host. Los grupos de host deben tener definida únicamente la propiedad `groupid`. Comportamiento del parámetro: - *obligatorio*
interfaces	objeto/array	Interfaces que se crearán para el host.
tags	objeto/array	Etiquetas de host.
templates	objeto/array	Templates que se enlazarán al host. Los templates deben tener definida únicamente la propiedad `templateid`.
macros	objeto/array	Macros de usuario que se crearán para el host.
inventory	objeto	Propiedades del inventario de host.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los equipos creados bajo la propiedad `hostids`. El orden de los IDs devueltos coincide con el orden de los equipos enviados.

Ejemplos

Creando un equipo

Cree un equipo llamado "Linux server" con una interfaz IP y etiquetas, agréguelo a un grupo, vincule una plantilla y establezca las direcciones MAC en el inventario del equipo.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "host.create",
    "params": {
        "host": "Linux server",
        "interfaces": [
            {
                "type": 1,
                "main": 1,
                "useip": 1,
                "ip": "192.168.3.1",
                "dns": "",
                "port": "10050"
            }
        ],
        "groups": [
            {
                "groupid": "50"
            }
        ],
        "tags": [
            {
                "tag": "host-name",
                "value": "linux-server"
```

```
            }
        ],
        "templates": [
            {
                "templateid": "20045"
            }
        ],
        "macros": [
            {
                "macro": "{$USER_ID}",
                "value": "123321"
            },
            {
                "macro": "{$USER_LOCATION}",
                "value": "0:0:0",
                "description": "coordenadas de latitud, longitud y altitud"
            }
        ],
        "inventory_mode": 0,
        "inventory": {
            "macaddress_a": "01234",
            "macaddress_b": "56768"
        }
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "107819"
        ]
    },
    "id": 1
}
```

Creando un equipo con interfaz SNMP

Cree un equipo llamado ”SNMP host” con una interfaz SNMPv3 con detalles.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "host.create",
    "params": {
        "host": "SNMP host",
        "interfaces": [
            {
                "type": 2,
                "main": 1,
                "useip": 1,
                "ip": "127.0.0.1",
                "dns": "",
                "port": "161",
                "details": {
                    "version": 3,
                    "bulk": 0,
                    "securityname": "mysecurityname",
                    "contextname": "",
                    "securitylevel": 1
                }
```

```
            }
        ],
        "groups": [
            {
                "groupid": "4"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10658"
        ]
    },
    "id": 1
}
```

Creación de un host con cifrado PSK

Cree un host llamado "PSK host" con el cifrado PSK configurado. Tenga en cuenta que el host debe estar preconfigurado para usar PSK.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "host.create",
    "params": {
        "host": "PSK host",
        "interfaces": [
            {
                "type": 1,
                "ip": "192.168.3.1",
                "dns": "",
                "port": "10050",
                "useip": 1,
                "main": 1
            }
        ],
        "groups": [
            {
                "groupid": "2"
            }
        ],
        "tls_accept": 2,
        "tls_connect": 2,
        "tls_psk_identity": "PSK 001",
        "tls_psk": "1f87b595725ac58dd977beef14b97461a7c1045b9a1c963065002c5473194952"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10590"
        ]
```

```
    },
    "id": 1
}
```

Creación de un equipo monitorizado por un proxy

Cree un equipo que será monitorizado por el proxy con ID ”1”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "host.create",
    "params": {
        "host": "Equipo monitorizado por proxy",
        "groups": [
            {
                "groupid": "2"
            }
        ],
        "monitored_by": 1,
        "proxyid": 1
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10591"
        ]
    },
    "id": 1
}
```

Creando un equipo monitorizado por un grupo de proxies

Cree un equipo que será monitorizado por el grupo de proxies con ID ”1”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "host.create",
    "params": {
        "host": "Equipo monitorizado por grupo de proxies",
        "groups": [
            {
                "groupid": "2"
            }
        ],
        "monitored_by": 2,
        "proxy_groupid": 1
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10592"
        ]
```

```
    },
    "id": 1
}
```

Ver también

- Grupo de hosts
- Template
- Macro de usuario
- Interfaz de host
- Inventario de host
- Etiqueta de host
- Proxy
- Grupo de proxies

Fuente

CHost::create() en *ui/include/classes/api/services/CHost.php*.

host.delete

Descripción

`object host.delete(array hosts)`

Este método permite eliminar hosts.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(array)` IDs de los equipos a eliminar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los equipos eliminados bajo la propiedad `hostids`.

Ejemplos

Eliminando varios equipos

Eliminar dos equipos.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "host.delete",
    "params": [
        "13",
        "32"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "13",
            "32"
        ]
    },
    "id": 1
}
```

Fuente

CHost::delete() en *ui/include/classes/api/services/CHost.php*.

host.get

Descripción

`integer/array host.get(object parameters)`

El método permite recuperar hosts según los parámetros proporcionados.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
groupids	ID/array	Devolver solo los hosts que pertenecen a los grupos indicados.
dserviceids	ID/array	Devolver solo los hosts que están relacionados con los servicios descubiertos indicados.
graphids	ID/array	Devolver solo los hosts que tienen los gráficos indicados.
hostids	ID/array	Devolver solo los hosts con los IDs de host indicados.
httptestids	ID/array	Devolver solo los hosts que tienen las comprobaciones web indicadas.
interfaceids	ID/array	Devolver solo los hosts que usan las interfaces indicadas.
itemids	ID/array	Devolver solo los hosts que tienen los items indicados.
maintenanceids	ID/array	Devolver solo los hosts que están afectados por los mantenimientos indicados.
monitored_hosts	flag	Devolver solo los hosts monitorizados.
proxyids	ID/array	Devolver solo los hosts que son monitorizados por los proxies indicados.
proxy_groupids	ID/array	Devolver solo los hosts que son monitorizados por los grupos de proxy indicados.
templated_hosts	flag	Devolver tanto hosts como templates.
templateids	ID/array	Devolver solo los hosts que están vinculados a los templates indicados.
triggerids	ID/array	Devolver solo los hosts que tienen los triggers indicados.
with_items	flag	Devolver solo los hosts que tienen items. Sobrescribe los parámetros `with_monitored_items` y `with_simple_graph_items`.
with_item_prototypes	flag	Devolver solo los hosts que tienen prototipos de item. Sobrescribe el parámetro `with_simple_graph_item_prototypes`.
with_simple_graph_item_prototypes	flag	Devolver solo los hosts que tienen prototipos de item, que están habilitados para la creación y tienen tipo de información numérica.
with_graphs	flag	Devolver solo los hosts que tienen gráficos.
with_graph_prototypes	flag	Devolver solo los hosts que tienen prototipos de gráfico.
with_httptests	flag	Devolver solo los hosts que tienen comprobaciones web. Sobrescribe el parámetro `with_monitored_httptests`.
with_monitored_httptests	flag	Devolver solo los hosts que tienen comprobaciones web habilitadas.
with_monitored_items	flag	Devolver solo los hosts que tienen items habilitados. Sobrescribe el parámetro `with_simple_graph_items`.
with_monitored_triggers	flag	Devolver solo los hosts que tienen triggers habilitados. Todos los items utilizados en el trigger también deben estar habilitados.
with_simple_graph_items	flag	Devolver solo los hosts que tienen items con tipo de información numérica.

Parámetro	Tipo	Descripción
with_triggers	flag	Devolver solo los hosts que tienen triggers. Sobrescribe el parámetro `with_monitored_triggers`.
withProblemsSuppressed	boolean	Si se establece en `true`, devolver solo los hosts con problemas suprimidos.
evaltype	integer	Método de evaluación de la etiqueta. Valores posibles: 0 - *(por defecto)* And/Or; 2 - Or.
severities	integer/array	Devolver hosts que solo tengan problemas con las severidades indicadas. Solo se aplica si el objeto de problema es un trigger.
tags	object/array	Devolver solo los hosts con las etiquetas indicadas. Formato: `[{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]`. Un array vacío devuelve todos los hosts. Valores posibles de operator: 0 - *(por defecto)* Contiene; 1 - Igual a; 2 - No contiene; 3 - No igual a; 4 - Existe; 5 - No existe.
inheritedTags	boolean	Devolver hosts que tengan las `tags` indicadas también en todos sus templates vinculados. Valores posibles: `true` - los templates vinculados también deben tener las `tags` indicadas; `false` - *(por defecto)* se ignoran las etiquetas de los templates vinculados.
selectDiscoveryData	query	Devolver una propiedad `discoveryData` con los datos del objeto de descubrimiento de host. El objeto de descubrimiento de host vincula un host descubierto a un prototipo de host del que se descubrió. Tiene las siguientes propiedades: `host` - `(string)` ID del host; `parent_hostid` - `(string)` ID del prototipo de host del que se ha creado el host; `status` - `(int)` estado de descubrimiento del host: 0 - *(por defecto)* el host está descubierto, 1 - el host ya no está descubierto; `ts_delete` - `(timestamp)` hora en la que se eliminará un host que ya no está descubierto; `ts_disable` - `(timestamp)` hora en la que se deshabilitará un host que ya no está descubierto; `disable_source` - `(int)` indicador de si el host fue deshabilitado por una regla LLD o manualmente: 0 - *(por defecto)* deshabilitado automáticamente, 1 - deshabilitado por una regla LLD.
selectDiscoveryRule	query	Devolver una propiedad `discoveryRule` con la regla de descubrimiento de bajo nivel que creó el host (a partir del prototipo de host en la monitorización de VMware).
selectDiscoveryRules	query	Devolver una propiedad `discoveryRules` con las reglas LLD del host. Admite `count`.
selectGraphs	query	Devolver una propiedad `graphs` con los gráficos del host. Admite `count`.
selectHostGroups	query	Devolver una propiedad `hostgroups` con los datos de los grupos de host a los que pertenece el host.

Parámetro	Tipo	Descripción
selectHttpTests	query	Devolver una propiedad `httpTests` con los escenarios web del host. Admite `count`.
selectInterfaces	query	Devolver una propiedad `interfaces` con las interfaces del host. Admite `count`.
selectInventory	query	Devolver una propiedad `inventory` con los datos de inventario del host.
selectItems	query	Devolver una propiedad `items` con los items del host. Admite `count`.
selectMacros	query	Devolver una propiedad `macros` con las macros del host.
selectParentTemplates	query	Devolver una propiedad `parentTemplates` con los templates a los que está vinculado el host. Además de los campos del objeto Template, contiene `link_type` - `(integer)` la forma en que el template está vinculado al host. Valores posibles: 0 - *(por defecto)* vinculado manualmente; 1 - vinculado automáticamente por LLD. Admite `count`.
selectDashboards	query	Devolver una propiedad `dashboards`. Admite `count`.
selectTags	query	Devolver una propiedad `tags` con las etiquetas del host.
selectInheritedTags	query	Devolver una propiedad `inheritedTags` con las etiquetas que están en todos los templates que están vinculados al host.
selectTriggers	query	Devolver una propiedad `triggers` con los triggers del host. Admite `count`.
selectValueMaps	query	Devolver una propiedad `valuemaps` con los mapas de valores del host.
filter	object	Devolver solo aquellos resultados que coincidan exactamente con el filtro indicado. Acepta un objeto, donde las claves son nombres de propiedades y los valores son un único valor o un array de valores con los que comparar. No admite propiedades del tipo de datos `text` data type. Admite propiedades adicionales: propiedades de Host interface.
limitSelects	integer	Limita el número de registros devueltos por subconsultas. Se aplica a las siguientes subconsultas: `selectParentTemplates` - los resultados se ordenarán por `host`; `selectInterfaces`; `selectItems` - ordenados por `name`; `selectDiscoveryRules` - ordenados por `name`; `selectTriggers` - ordenados por `description`; `selectGraphs` - ordenados por `name`; `selectDashboards` - ordenados por `name`.
search	object	Devolver resultados que coincidan con el patrón indicado (no distingue mayúsculas/minúsculas). Acepta un objeto, donde las claves son nombres de propiedades y los valores son cadenas a buscar. Si no se dan opciones adicionales, esto realizará una búsqueda `LIKE "%…%"`. Solo admite propiedades de tipo de datos `string` y `text` data type. Admite propiedades adicionales: propiedades de Host interface.

Parámetro	Tipo	Descripción
searchInventory	object	Devolver hosts que tengan datos de inventario que coincidan con el patrón indicado (no distingue mayúsculas/minúsculas). Acepta un objeto, donde las claves son nombres de propiedades y los valores son cadenas a buscar. Si no se dan opciones adicionales, esto realizará una búsqueda `LIKE "%…%"`. Solo admite propiedades de tipo de datos `string` y `text` data type.
sortfield	string/array	Ordenar el resultado por las propiedades indicadas. Valores posibles: `hostid`, `host`, `name`, `status`.
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
editable	boolean	
excludeSearch	boolean	
limit	integer	
output	query	
preservekeys	boolean	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	
selectDiscoveries	query	Devolver una propiedad `discoveries` con las reglas de descubrimiento de bajo nivel del host. Admite `count`. Esta consulta está **obsoleta**, utilice `selectDiscoveryRules` en su lugar.
selectHostDiscovery	query	Devolver una propiedad `hostDiscovery` con los datos del objeto de descubrimiento de host. Esta consulta está **obsoleta**, utilice `selectDiscoveryData` en su lugar.

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperar datos por nombre

Recuperar todos los datos sobre dos equipos llamados ”Zabbix server” y ”Linux server”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "filter": {
            "host": [
                "Zabbix server",
                "Linux server"
            ]
        }
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "hostid": "10160",
            "proxyid": "0",
            "host": "Zabbix server",
            "status": "0",
            "ipmi_authtype": "-1",
            "ipmi_privilege": "2",
            "ipmi_username": "",
            "ipmi_password": "",
            "maintenanceid": "0",
            "maintenance_status": "0",
            "maintenance_type": "0",
            "maintenance_from": "0",
            "name": "Zabbix server",
            "flags": "0",
            "description": "The Zabbix monitoring server.",
            "tls_connect": "1",
            "tls_accept": "1",
            "tls_issuer": "",
            "tls_subject": "",
            "proxy_groupid": "0",
            "monitored_by": "0",
            "inventory_mode": "1",
            "active_available": "1",
            "assigned_proxyid": "0"
        },
        {
            "hostid": "10167",
            "proxyid": "0",
            "host": "Linux server",
            "status": "0",
            "ipmi_authtype": "-1",
            "ipmi_privilege": "2",
            "ipmi_username": "",
            "ipmi_password": "",
            "maintenanceid": "0",
            "maintenance_status": "0",
            "maintenance_type": "0",
            "maintenance_from": "0",
            "name": "Linux server",
            "flags": "0",
            "description": "",
            "tls_connect": "1",
            "tls_accept": "1",
            "tls_issuer": "",
            "tls_subject": "",
            "proxy_groupid": "0",
            "monitored_by": "0",
            "inventory_mode": "1",
            "active_available": "1",
            "assigned_proxyid": "0"
        }
    ],
    "id": 1
}
```


Recuperar grupos de equipos

Recuperar los grupos de equipos a los que pertenece el equipo ”Zabbix server”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": ["hostid"],
        "selectHostGroups": "extend",
        "filter": {
            "host": [
                "Zabbix server"
            ]
        }
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "hostid": "10085",
            "hostgroups": [
                {
                    "groupid": "2",
                    "name": "Linux servers",
                    "flags": "0",
                    "uuid": "dc579cd7a1a34222933f24f52a68bcd8"
                },
                {
                    "groupid": "4",
                    "name": "Zabbix servers",
                    "flags": "0",
                    "uuid": "6f6799aa69e844b4b3918f779f2abf08"
                }
            ]
        }
    ],
    "id": 1
}
```

Recuperar plantillas vinculadas

Recuperar los IDs y nombres de las plantillas vinculadas al equipo ”10084”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": ["hostid"],
        "selectParentTemplates": [
            "templateid",
            "name"
        ],
        "hostids": "10084"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
```

```
        {
            "hostid": "10084",
            "parentTemplates": [
                {
                    "name": "Linux",
                    "templateid": "10001"
                },
                {
                    "name": "Zabbix Server",
                    "templateid": "10047"
                }
            ]
        }
    ],
    "id": 1
}
```

Recuperar equipos por plantilla

Recuperar los equipos que tienen la plantilla ”10001” (*Linux por Zabbix agent*) vinculada a ellos.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": ["hostid", "name"],
        "templateids": "10001"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "templateid": "10001",
            "hosts": [
                {
                    "hostid": "10084",
                    "name": "Zabbix server"
                },
                {
                    "hostid": "10603",
                    "name": "Host 1"
                },
                {
                    "hostid": "10604",
                    "name": "Host 2"
                }
            ]
        }
    ],
    "id": 1
}
```

Búsqueda por datos de inventario de equipos

Recuperar equipos que contengan ”Linux” en el campo ”SO” del inventario de equipos.

Solicitud:

```
{
    "jsonrpc": "2.0",
```

```
    "method": "host.get",
    "params": {
        "output": [
            "host"
        ],
        "selectInventory": [
            "os"
        ],
        "searchInventory": {
            "os": "Linux"
        }
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "hostid": "10084",
            "host": "Zabbix server",
            "inventory": {
                "os": "Linux Ubuntu"
            }
        },
        {
            "hostid": "10107",
            "host": "Linux server",
            "inventory": {
                "os": "Linux Mint"
            }
        }
    ],
    "id": 1
}
```

Búsqueda por etiquetas de host

Recuperar hosts que tengan la etiqueta ”host-name” igual a ”linux-server”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": ["hostid"],
        "selectTags": "extend",
        "evaltype": 0,
        "tags": [
            {
                "tag": "host-name",
                "value": "linux-server",
                "operator": 1
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
```

```
    "result": [
        {
            "hostid": "10085",
            "tags": [
                {
                    "tag": "host-name",
                    "value": "linux-server",
                    "automatic": "0"
                },
                {
                    "tag": "os",
                    "value": "rhel-7",
                    "automatic": "0"
                }
            ]
        }
    ],
    "id": 1
}
```

Recuperar hosts que tengan estas etiquetas no solo a nivel de host sino también en sus templates padre vinculados.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": ["name"],
        "tags": [
            {
                "tag": "os",
                "value": "rhel-7",
                "operator": 1
            }
        ],
        "inheritedTags": true
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "hostid": "10623",
            "name": "PC room 1"
        },
        {
            "hostid": "10601",
            "name": "Office"
        }
    ],
    "id": 1
}
```

Recuperar hosts excluyendo aquellos con la etiqueta ”component”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
```

```
        "output": ["hostid"],
        "selectTags": "extend",
        "evaltype": 0,
        "tags": [
            {
                "tag": "component",
                "value": "",
                "operator": 5
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "hostid": "10084",
            "tags": [
                {
                    "tag": "class",
                    "value": "os",
                    "automatic": "0"
                },
                {
                    "tag": "target",
                    "value": "linux",
                    "automatic": "0"
                }
            ]
        },
        {
            "hostid": "10629",
            "tags": [
                {
                    "tag": "class",
                    "value": "application",
                    "automatic": "0"
                },
                {
                    "tag": "target",
                    "value": "browser",
                    "automatic": "0"
                }
            ]
        }
    ],
    "id": 1
}
```

Búsqueda de equipo con etiquetas y etiquetas de plantilla

Recuperar un equipo con etiquetas y todas las etiquetas que están vinculadas a las plantillas padre.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": ["name"],
        "hostids": 10502,
```

```
        "selectTags": ["tag", "value"],
        "selectInheritedTags": ["tag", "value"]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "hostid": "10502",
            "name": "Desktop",
            "tags": [
                {
                    "tag": "host-name",
                    "value": "linux-server"
                },
                {
                    "tag": "os",
                    "value": "rhel-7"
                }
            ],
            "inheritedTags": [
                {
                    "tag": "class",
                    "value": "os"
                },
                {
                    "tag": "target",
                    "value": "linux"
                },
                {
                    "tag": "os",
                    "value": "rhel-7"
                }
            ]
        }
    ],
    "id": 1
}
```

Buscar equipos por gravedad del problema

Recuperar equipos que tienen problemas de tipo ”Desastre”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": ["name"],
        "severities": 5
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "hostid": "10160",
```

```
            "name": "Zabbix server"
        }
    ],
    "id": 1
}
```

Recuperar equipos que tienen problemas de tipo ”Media” y ”Alta”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": ["name"],
        "severities": [3, 4]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "hostid": "20170",
            "name": "Database"
        },
        {
            "hostid": "20183",
            "name": "workstation"
        }
    ],
    "id": 1
}
```

Recuperar los items del host

Recuperar los items para el host ”Zabbix server”. La solicitud se limita a las propiedades del item `itemid`, `name` y `status`. Debido al gran tamaño de la respuesta, solo se muestra un subconjunto de items en el ejemplo.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": ["hostid"],
        "selectItems": ["itemid","name","status"],
        "filter": {
            "host": [
                "Zabbix server"
            ]
        }
    },
    "id": 1
}
```

Respuesta:

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "hostid": "10084",
      "items": [
```

```
    {
      "itemid": "42227",
      "name": "Host name of Zabbix agent running",
      "status": "0"
    },
    {
      "itemid": "42237",
      "name": "Zabbix agent ping",
      "status": "0"
    },
    {
      "itemid": "42250",
      "name": "Version of Zabbix agent running",
      "status": "0"
    },
    {
      "itemid": "42251",
      "name": "Maximum number of open file descriptors",
      "status": "0"
    },
    {
      "itemid": "42252",
      "name": "Maximum number of processes",
      "status": "0"
    },
    {
      "itemid": "69869",
      "name": "Interface enp0s3: Inbound packets discarded",
      "status": "0"
    },
    {
      "itemid": "69870",
      "name": "Interface enp0s3: Inbound packets with errors",
      "status": "0"
    },
    {
      "itemid": "69871",
      "name": "Interface enp0s3: Bits received",
      "status": "0"
    },
    {
      "itemid": "69872",
      "name": "Interface enp0s3: Outbound packets discarded",
      "status": "0"
    },
    {
      "itemid": "69873",
      "name": "Interface enp0s3: Outbound packets with errors",
      "status": "0"
    },
    {
      "itemid": "69874",
      "name": "Interface enp0s3: Bits sent",
      "status": "0"
    },
    {
      "itemid": "42253",
      "name": "Number of processes",
      "status": "0"
    },
    {
      "itemid": "42254",
```

```
            "name": "Number of running processes",
            "status": "0"
          },
          {
            "itemid": "42255",
            "name": "System boot time",
            "status": "0"
          }
        ]
      }
    ],
    "id": 1
}
```

Ver también

- Grupo de hosts
- Template
- Macro de usuario
- Interfaz de host
- Proxy
- Grupo de proxies

Fuente

CHost::get() en *ui/include/classes/api/services/CHost.php*.

host.massadd

Descripción

`object host.massadd(object parameters)`

Este método permite agregar simultáneamente múltiples objetos relacionados a todos los hosts especificados.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object)` Parámetros que contienen los IDs de los hosts a actualizar y los objetos a añadir a todos los hosts.

El método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
hosts	object/array	Hosts que se actualizarán. Los hosts deben tener definida únicamente la propiedad `hostid`. Comportamiento del parámetro: - *obligatorio*
groups	object/array	Grupos de hosts a añadir a los hosts indicados. Los grupos de hosts deben tener definida únicamente la propiedad `groupid`.
interfaces	object/array	Interfaces de host que se crearán para los hosts indicados.
macros	object/array	Macros de usuario que se crearán para los hosts indicados.
templates	object/array	Templates a enlazar con los hosts indicados. Los templates deben tener definida únicamente la propiedad `templateid`.

Valores de retorno

(object) Devuelve un objeto que contiene los IDs de los equipos actualizados bajo la propiedad hostids.

Ejemplos

Añadir macros

Añadir dos nuevas macros a dos equipos.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "host.massadd",
    "params": {
        "hosts": [
            {
                "hostid": "10160"
            },
            {
                "hostid": "10167"
            }
        ],
        "macros": [
            {
                "macro": "{$TEST1}",
                "value": "MACROTEST1"
            },
            {
                "macro": "{$TEST2}",
                "value": "MACROTEST2",
                "description": "Descripción de prueba"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10160",
            "10167"
        ]
    },
    "id": 1
}
```

Ver también

- host.update
- Grupo de hosts
- Template
- Macro de usuario
- Interfaz de host

Fuente

CHost::massAdd() en *ui/include/classes/api/services/CHost.php*.

host.massremove

Descripción

`object host.massremove(object parameters)`

Este método permite eliminar objetos relacionados de varios host.

Note:
Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(objeto)` Parámetros que contienen los IDs de los hosts a actualizar y los objetos que deben ser eliminados.

Parámetro	Tipo	Descripción
hostids	ID/array	IDs de los hosts a actualizar. Comportamiento del parámetro: *- requerido*
groupids	ID/array	IDs de los grupos de hosts de los que se eliminarán los hosts indicados.
interfaces	objeto/array	Interfaces de host a eliminar de los hosts indicados. El objeto de interfaz de host debe tener definidas únicamente las propiedades `ip`, `dns` y `port`.
macros	string/array	Macros de usuario a eliminar de los hosts indicados.
templateids	ID/array	IDs de las templates a desvincular de los hosts indicados.
templateids_clear	ID/array	IDs de las templates a desvincular y limpiar de los hosts indicados.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los equipos actualizados bajo la propiedad `hostids`.

Ejemplos

Desvinculación de plantillas

Desvincular una plantilla de dos equipos y eliminar todas las entidades de la plantilla.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "host.massremove",
    "params": {
        "hostids": ["69665", "69666"],
        "templateids_clear": "325"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "69665",
            "69666"
        ]
    },
    "id": 1
}
```

Ver también

- host.update
- Macro de usuario
- Interfaz de host

Fuente

CHost::massRemove() en *ui/include/classes/api/services/CHost.php*.

host.update

Descripción

`object host.update(object/array hosts)`

Este método permite actualizar hosts existentes.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object/array)` Propiedades del host que se actualizarán.

La propiedad `hostid` debe estar definida para cada host, todas las demás propiedades son opcionales. Sólo se actualizarán las propiedades indicadas, todas las demás permanecerán sin cambios.

Tenga en cuenta, sin embargo, que al actualizar el nombre técnico del host también se actualizará el nombre visible del host (si no se indica por separado) por el valor del nombre técnico del host.

Además de las propiedades estándar del host, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
groups	object/array	Grupos de hosts para reemplazar los grupos de hosts actuales a los que pertenece el host. Todos los grupos de hosts que no estén listados en la solicitud serán desvinculados. Los grupos de hosts deben tener definida únicamente la propiedad `groupid`.
interfaces	object/array	Interfaces de host para reemplazar las interfaces de host actuales. Todas las interfaces que no estén listadas en la solicitud serán eliminadas.
tags	object/array	Etiquetas de host para reemplazar las etiquetas de host actuales. Todas las etiquetas que no estén listadas en la solicitud serán eliminadas.
inventory	object	Propiedades del inventario de host.
macros	object/array	Macros de usuario para reemplazar las macros de usuario actuales. Todas las macros que no estén listadas en la solicitud serán eliminadas.
templates	object/array	Templates para reemplazar los templates actualmente vinculados. Todos los templates que no estén listados en la solicitud sólo serán desvinculados. Los templates deben tener definida únicamente la propiedad `templateid`.
templates_clear	object/array	Templates para desvincular y limpiar del host. Los templates deben tener definida únicamente la propiedad `templateid`.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los equipos actualizados bajo la propiedad `hostids`.

Ejemplos

Habilitar un equipo

Habilite la monitorización del equipo, es decir, establezca su estado en "0".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "host.update",
```

```
    "params": {
        "hostid": "10126",
        "status": 0
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10126"
        ]
    },
    "id": 1
}
```

Habilitar varios hosts

Habilite la monitorización de dos hosts, es decir, establezca su estado en ”0”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "host.update",
    "params": [
        {
            "hostid": "10127",
            "status": 0
        },
        {
            "hostid": "10128",
            "status": 0
        }
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10127",
            "10128"
        ]
    },
    "id": 1
}
```

Desvincular plantillas

Desvincular y limpiar dos plantillas de un equipo.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "host.update",
    "params": {
        "hostid": "10126",
        "templates_clear": [
            {
                "templateid": "10124"
```

```
            },
            {
                "templateid": "10125"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10126"
        ]
    },
    "id": 1
}
```

Actualización de macros de equipo

Reemplace todas las macros de equipo por dos nuevas.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "host.update",
    "params": {
        "hostid": "10126",
        "macros": [
            {
                "macro": "{$PASS}",
                "value": "password"
            },
            {
                "macro": "{$DISC}",
                "value": "sda",
                "description": "Descripción actualizada"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10126"
        ]
    },
    "id": 1
}
```

Actualizando el inventario del equipo

Cambiar el modo de inventario y añadir ubicación

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "host.update",
```

```
    "params": {
        "hostid": "10387",
        "inventory_mode": 0,
        "inventory": {
            "location": "Latvia, Riga"
        }
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10387"
        ]
    },
    "id": 1
}
```

Actualización de etiquetas de equipo

Reemplazar todas las etiquetas de equipo por una nueva.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "host.update",
    "params": {
        "hostid": "10387",
        "tags": {
            "tag": "os",
            "value": "rhel-7"
        }
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10387"
        ]
    },
    "id": 1
}
```

Actualización de macros de equipo descubiertas

Convertir la macro ”automática” creada por la regla de descubrimiento en ”manual” y cambiar su valor a ”nuevo-valor”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "host.update",
    "params": {
        "hostid": "10387",
        "macros": {
            "hostmacroid": "5541",
            "value": "nuevo-valor",
            "automatic": "0"
```

```
        }
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10387"
        ]
    },
    "id": 1
}
```

Actualización del cifrado del host

Actualice el host ”10590” para usar cifrado PSK solo para las conexiones desde el host al server de Zabbix, y cambie la identidad PSK y la clave PSK. Tenga en cuenta que el host debe estar preconfigurado para usar PSK.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "host.update",
    "params": {
        "hostid": "10590",
        "tls_connect": 1,
        "tls_accept": 2,
        "tls_psk_identity": "PSK 002",
        "tls_psk": "e560cb0d918d26d31b4f642181f5f570ad89a390931102e5391d08327ba434e9"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10590"
        ]
    },
    "id": 1
}
```

Ver también

- host.massadd
- host.massremove
- Grupo de hosts
- Template
- Macro de usuario
- Interfaz de host
- Inventario de host
- Etiqueta de host
- Proxy
- Grupo de proxies

Fuente

CHost::update() en *ui/include/classes/api/services/CHost.php*.

Host descubierto

Esta clase está diseñada para trabajar con hosts descubiertos.

Referencias de objetos:

- Host descubierto

Métodos disponibles:

- dhost.get - recuperar hosts descubiertos

Objeto de equipo descubierto

Los siguientes objetos están directamente relacionados con la API `dhost`.

Host descubierto

> **Note:**
> Los host descubiertos son creados por el servidor Zabbix y no pueden ser modificados a través de la API.

El objeto host descubierto contiene información sobre un host descubierto por una regla de descubrimiento de red. Tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
dhostid	ID	ID del host descubierto.
druleid	ID	ID de la regla de descubrimiento que detectó el host.
lastdown	timestamp	Hora en la que el host descubierto se cayó por última vez.
lastup	timestamp	Hora en la que el host descubierto se levantó por última vez.
status	integer	Indica si el host descubierto está activo o caído. Un host está activo si tiene al menos un servicio descubierto activo. Valores posibles: 0 - host activo; 1 - host caído.

dhost.get

Descripción

`integer/array dhost.get(object parameters)`

El método permite recuperar hosts descubiertos según los parámetros proporcionados.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
dhostids	ID/array	Devolver solo los hosts descubiertos con los IDs dados.
druleids	ID/array	Devolver solo los hosts descubiertos que han sido creados por las reglas de descubrimiento indicadas.
dserviceids	ID/array	Devolver solo los hosts descubiertos que están ejecutando los servicios indicados.
selectDRules	query	Devolver una propiedad `drules` con un array de las reglas de descubrimiento que detectaron el host.
selectDServices	query	Devolver una propiedad `dservices` con los servicios descubiertos que se ejecutan en el host. Admite `count`.

Parámetro	Tipo	Descripción
limitSelects	integer	Limita el número de registros devueltos por las subselecciones. Se aplica a las siguientes subselecciones: `selectDServices` - los resultados se ordenarán por `dserviceid`.
sortfield	string/array	Ordenar el resultado por las propiedades indicadas. Valores posibles: `dhostid`, `druleid`.
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperar equipos descubiertos por regla de descubrimiento

Recupera todos los equipos y los servicios descubiertos que están ejecutando y que han sido detectados por la regla de descubrimiento "4".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "dhost.get",
    "params": {
        "output": "extend",
        "selectDServices": "extend",
        "druleids": "4"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "dservices": [
                {
                    "dserviceid": "1",
                    "dhostid": "1",
                    "type": "4",
                    "key_": "",
                    "value": "",
                    "port": "80",
                    "status": "0",
                    "lastup": "1337697227",
                    "lastdown": "0",
                    "dcheckid": "5",
```

```
                "ip": "192.168.1.1",
                "dns": "station.company.lan"
            }
        ],
        "dhostid": "1",
        "druleid": "4",
        "status": "0",
        "lastup": "1337697227",
        "lastdown": "0"
    },
    {
        "dservices": [
            {
                "dserviceid": "2",
                "dhostid": "2",
                "type": "4",
                "key_": "",
                "value": "",
                "port": "80",
                "status": "0",
                "lastup": "1337697234",
                "lastdown": "0",
                "dcheckid": "5",
                "ip": "192.168.1.4",
                "dns": "john.company.lan"
            }
        ],
        "dhostid": "2",
        "druleid": "4",
        "status": "0",
        "lastup": "1337697234",
        "lastdown": "0"
    },
    {
        "dservices": [
            {
                "dserviceid": "3",
                "dhostid": "3",
                "type": "4",
                "key_": "",
                "value": "",
                "port": "80",
                "status": "0",
                "lastup": "1337697234",
                "lastdown": "0",
                "dcheckid": "5",
                "ip": "192.168.1.26",
                "dns": "printer.company.lan"
            }
        ],
        "dhostid": "3",
        "druleid": "4",
        "status": "0",
        "lastup": "1337697234",
        "lastdown": "0"
    },
    {
        "dservices": [
            {
                "dserviceid": "4",
                "dhostid": "4",
                "type": "4",
```

```
                    "key_": "",
                    "value": "",
                    "port": "80",
                    "status": "0",
                    "lastup": "1337697234",
                    "lastdown": "0",
                    "dcheckid": "5",
                    "ip": "192.168.1.7",
                    "dns": "mail.company.lan"
                }
            ],
            "dhostid": "4",
            "druleid": "4",
            "status": "0",
            "lastup": "1337697234",
            "lastdown": "0"
        }
    ],
    "id": 1
}
```


Ver también

- Servicio descubierto
- Regla de descubrimiento

Fuente

CDHost::get() en *ui/include/classes/api/services/CDHost.php*.

Housekeeping

Esta clase está diseñada para trabajar con housekeeping.

Referencias de objetos:

- Housekeeping

Métodos disponibles:

- housekeeping.get - recuperar housekeeping
- housekeeping.update - actualizar housekeeping

Objeto de mantenimiento

Los siguientes objetos están directamente relacionados con la API de `housekeeping`.

Mantenimiento

El objeto de configuración tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
hk_events_mode	integer	Habilitar el mantenimiento interno para eventos y alertas. Valores posibles: 0 - Deshabilitar; 1 - *(predeterminado)* Habilitar.
hk_events_trigger	string	Período de almacenamiento de los datos de los trigger. Acepta un número de segundos (por ejemplo, 1, 60, 300) o un valor con un sufijo de tiempo (por ejemplo, `30s`, `1m`, `2h`, `1d`). Predeterminado: `365d`.

Propiedad	Tipo	Descripción
hk_events_service	string	Período de almacenamiento de los datos de los servicios. Acepta un número de segundos (por ejemplo, 1, 60, 300) o un valor con un sufijo de tiempo (por ejemplo, 30s, 1m, 2h, 1d). Predeterminado: 1d.
hk_events_internal	string	Período de almacenamiento de los datos internos. Acepta un número de segundos (por ejemplo, 1, 60, 300) o un valor con un sufijo de tiempo (por ejemplo, 30s, 1m, 2h, 1d). Predeterminado: 1d.
hk_events_discovery	string	Período de almacenamiento de los datos de descubrimiento de red. Acepta un número de segundos (por ejemplo, 1, 60, 300) o un valor con un sufijo de tiempo (por ejemplo, 30s, 1m, 2h, 1d). Predeterminado: 1d.
hk_events_autoreg	string	Período de almacenamiento de los datos de autorregistro. Acepta un número de segundos (por ejemplo, 1, 60, 300) o un valor con un sufijo de tiempo (por ejemplo, 30s, 1m, 2h, 1d). Predeterminado: 1d.
hk_services_mode	integer	Habilitar el mantenimiento interno para los servicios. Valores posibles: 0 - Deshabilitar; 1 - *(predeterminado)* Habilitar.
hk_services	string	Período de almacenamiento de los datos de los servicios. Acepta un número de segundos (por ejemplo, 1, 60, 300) o un valor con un sufijo de tiempo (por ejemplo, 30s, 1m, 2h, 1d). Predeterminado: 365d.
hk_audit_mode	integer	Habilitar el mantenimiento interno para la auditoría. Valores posibles: 0 - Deshabilitar; 1 - *(predeterminado)* Habilitar.
hk_audit	string	Período de almacenamiento de los datos de auditoría. Acepta un número de segundos (por ejemplo, 1, 60, 300) o un valor con un sufijo de tiempo (por ejemplo, 30s, 1m, 2h, 1d). Predeterminado: 31d.
hk_sessions_mode	integer	Habilitar el mantenimiento interno para las sesiones. Valores posibles: 0 - Deshabilitar; 1 - *(predeterminado)* Habilitar.
hk_sessions	string	Período de almacenamiento de los datos de las sesiones. Acepta un número de segundos (por ejemplo, 1, 60, 300) o un valor con un sufijo de tiempo (por ejemplo, 30s, 1m, 2h, 1d). Predeterminado: 31d.
hk_history_mode	integer	Habilitar el mantenimiento interno para el historial. Valores posibles: 0 - Deshabilitar; 1 - *(predeterminado)* Habilitar.
hk_history_global	integer	Anular el período del historial del item. Valores posibles: 0 - No anular; 1 - *(predeterminado)* Anular.

Propiedad	Tipo	Descripción
hk_history	string	Período de almacenamiento de los datos del historial. Acepta un número de segundos (por ejemplo, 1, 60, 300) o un valor con un sufijo de tiempo (por ejemplo, 30s, 1m, 2h, 1d). Predeterminado: 31d.
hk_trends_mode	integer	Habilitar el mantenimiento interno para las tendencias. Valores posibles: 0 - Deshabilitar; 1 - *(predeterminado)* Habilitar.
hk_trends_global	integer	Anular el período de tendencias del item. Valores posibles: 0 - No anular; 1 - *(predeterminado)* Anular.
hk_trends	string	Período de almacenamiento de los datos de tendencias. Acepta un número de segundos (por ejemplo, 1, 60, 300) o un valor con un sufijo de tiempo (por ejemplo, 30s, 1m, 2h, 1d). Predeterminado: 365d.
db_extension	string	Indicador de configuración de la extensión de la base de datos. Si este indicador se establece en ”timescaledb”, el server cambia su comportamiento para el mantenimiento y la eliminación de items. Comportamiento de la propiedad: - *solo lectura*
compression_availability	integer	Indica si la base de datos (o su extensión) admite la compresión de datos. Valores posibles: 0 - No disponible; 1 - Disponible. Comportamiento de la propiedad: - *solo lectura*
compression_status	integer	Habilitar la compresión de TimescaleDB para el historial y las tendencias. Valores posibles: 0 - *(predeterminado)* Desactivada; 1 - Activada.
compress_older	string	Comprimir los registros del historial y las tendencias que sean anteriores al período especificado. Acepta un número de segundos (por ejemplo, 1, 60, 300) o un valor con un sufijo de tiempo (por ejemplo, 30s, 1m, 2h, 1d). Predeterminado: 7d.

housekeeping.get

Descripción

`object housekeeping.get(object parameters)`

El método permite recuperar el objeto de housekeeping según los parámetros proporcionados.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método solo admite un parámetro.

Parámetro	Tipo	Descripción
output	query	Este parámetro se describe en el comentario de referencia.

Valores de retorno

`(object)` Devuelve el objeto de limpieza.

Ejemplos

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "housekeeping.get",
    "params": {
        "output": "extend"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hk_events_mode": "1",
        "hk_events_trigger": "365d",
        "hk_events_service": "1d",
        "hk_events_internal": "1d",
        "hk_events_discovery": "1d",
        "hk_events_autoreg": "1d",
        "hk_services_mode": "1",
        "hk_services": "365d",
        "hk_audit_mode": "1",
        "hk_audit": "31d",
        "hk_sessions_mode": "1",
        "hk_sessions": "365d",
        "hk_history_mode": "1",
        "hk_history_global": "0",
        "hk_history": "31d",
        "hk_trends_mode": "1",
        "hk_trends_global": "0",
        "hk_trends": "365d",
        "db_extension": "",
        "compression_status": "0",
        "compress_older": "7d"
    },
    "id": 1
}
```

Fuente

CHousekeeping ::get() en *ui/include/classes/api/services/CHousekeeping.php*.

housekeeping.update

Descripción

`object housekeeping.update(object housekeeping)`

Este método le permite actualizar la configuración de housekeeping existente.

Note:
Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración del rol de usuario. Consulte User roles para obtener más información.

Parámetros

`(objeto)` Propiedades de limpieza a actualizar.

Valores de retorno

`(array)` Devuelve un array con los nombres de los parámetros actualizados.

Ejemplos

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "housekeeping.update",
    "params": {
        "hk_events_mode": "1",
        "hk_events_trigger": "200d",
        "hk_events_internal": "2d",
        "hk_events_discovery": "2d"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        "hk_events_mode",
        "hk_events_trigger",
        "hk_events_internal",
        "hk_events_discovery"
    ],
    "id": 1
}
```

Fuente

CHousekeeping::update() en *ui/include/classes/api/services/CHousekeeping.php*.

Imagen

Esta clase está diseñada para trabajar con imágenes.

Referencias de objetos:

- Imagen

Métodos disponibles:

- image.create - crear nuevas imágenes
- image.delete - eliminar imágenes
- image.get - recuperar imágenes
- image.update - actualizar imágenes

Objeto de imagen

Los siguientes objetos están directamente relacionados con la API de `image`.

Imagen

El objeto imagen tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
imageid	ID	ID de la imagen. Comportamiento de la propiedad: - *solo lectura* - *requerido* para operaciones de actualización
name	string	Nombre de la imagen. Comportamiento de la propiedad: - *requerido* para operaciones de creación
imagetype	integer	Tipo de imagen. Valores posibles: 1 - *(por defecto)* icono; 2 - imagen de fondo. Comportamiento de la propiedad: - *constante* - *requerido* para operaciones de creación
image	string	Imagen codificada en Base64. El tamaño máximo de la imagen codificada es de 1 MB. El tamaño máximo puede ajustarse cambiando el valor de la constante `ZBX_MAX_IMAGE_SIZE`. Formatos de imagen soportados: PNG, JPEG, GIF y WebP. Comportamiento de la propiedad: - *requerido* para operaciones de creación

image.create

Descripción

`object image.create(object/array images)`

Este método le permite crear nuevas imágenes.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(objeto/array)` Imágenes a crear.

El método acepta imágenes con las propiedades estándar de imagen.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las imágenes creadas bajo la propiedad `imageids`. El orden de los IDs devueltos coincide con el orden de las imágenes enviadas.

Ejemplos

Crear una imagen de icono

Cree un icono de nube.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "image.create",
    "params": {
        "imagetype": 1,
        "name": "Cloud_(24)",
        "image": "iVBORw0KGgoAAAANSUhEUgAAABgAAAANCAYAAACzbK7QAAAABHNCSVQICAgIfAhkiAAAAAlwSFlzAAACmAAAApgB
```

```
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "imageids": [
            "188"
        ]
    },
    "id": 1
}
```

Crear una imagen de fondo

Cree una pequeña imagen de fondo para usarla como patrón.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "image.create",
    "params": {
        "imagetype": 2,
        "name": "background_image",
        "image": "iVBORw0KGgoAAAANSUhEUgAAABgAAAAYCAIAAABvFaqvAAABhWlDQ1BJQ0MgcHJvZmlsZQAAKJF9kb1Lw0AYxp+m
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "imageids": [
            "189"
        ]
    },
    "id": 1
}
```

Origen

CImage::create() en *ui/include/classes/api/services/CImage.php*.

image.delete

Descripción

`object image.delete(array imageIds)`

Este método le permite eliminar imágenes.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración del rol de usuario. Consulte User roles para obtener más información.

Parámetros

`(array)` IDs de las imágenes a eliminar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las imágenes eliminadas bajo la propiedad `imageids`.

Ejemplos

Eliminar varias imágenes

Eliminar dos imágenes.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "image.delete",
    "params": [
        "188",
        "192"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "imageids": [
            "188",
            "192"
        ]
    },
    "id": 1
}
```

Fuente

CImage::delete() en *ui/include/classes/api/services/CImage.php*.

image.get

Descripción

`integer/array image.get(object parameters)`

El método permite recuperar imágenes según los parámetros proporcionados.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para invocar el método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
imageids	ID/array	Devolver solo las imágenes con los IDs indicados.
sysmapids	ID/array	Devolver imágenes que se utilizan en los mapas indicados.
select_image	flag	Devolver una propiedad `image` con la imagen codificada en Base64.
sortfield	string/array	Ordenar el resultado por las propiedades indicadas. Valores posibles: `imageid`, `name`.
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	

Parámetro	Tipo	Descripción
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperar una imagen

Recuperar todos los datos de la imagen con ID "2".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "image.get",
    "params": {
        "output": "extend",
        "select_image": true,
        "imageids": "2"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "imageid": "2",
            "imagetype": "1",
            "name": "Cloud_(24)",
            "image": "iVBORw0KGgoAAAANSUhEUgAAABgAAAANCAYAAACzbK7QAAAABHNCSVQICAgIfAhkiAAAAAlwSFlzAAACmAAA
        }
    ],
    "id": 1
}
```

Fuente

CImage::get() en *ui/include/classes/api/services/CImage.php*.

image.update

Descripción

`object image.update(object/array images)`

Este método permite actualizar imágenes existentes.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(objeto/array)` Propiedades de la imagen que se van a actualizar.

La propiedad `imageid` debe estar definida para cada imagen, todas las demás propiedades son opcionales. Solo se actualizarán las propiedades pasadas, todas las demás permanecerán sin cambios.

El método acepta imágenes con las propiedades estándar de imagen.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las imágenes actualizadas bajo la propiedad `imageids`.

Ejemplos

Renombrar imagen

Renombrar imagen a "Cloud icon".

Petición:

```
{
    "jsonrpc": "2.0",
    "method": "image.update",
    "params": {
        "imageid": "2",
        "name": "Cloud icon"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "imageids": [
            "2"
        ]
    },
    "id": 1
}
```

Fuente

CImage::update() en *ui/include/classes/api/services/CImage.php*.

Información de la API

Esta clase está diseñada para recuperar metainformación sobre la API.

Métodos disponibles:

- apiinfo.version - recuperar la versión de la API de Zabbix

apiinfo.version

Descripción

`string apiinfo.version(array)`

Este método permite recuperar la versión de la API de Zabbix.

> **Attention:**
> Este método solo está disponible para usuarios no autenticados.

Parámetros

`(array)` El método acepta un array vacío.

Valores de retorno

`(string)` Devuelve la versión de la API de Zabbix.

Note:
A partir de Zabbix 2.0.4, la versión de la API coincide con la versión de Zabbix.

Ejemplos

Recuperar la versión de la API

Recupera la versión de la API de Zabbix.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "apiinfo.version",
    "params": [],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": "8.0.0",
    "id": 1
}
```

Fuente

CAPIInfo::version() en *ui/include/classes/api/services/CAPIInfo.php*.

Información del server

Esta clase está diseñada para recuperar información sobre el server de Zabbix.

Métodos disponibles:

- serverinfo.get - recuperar la hora de la última actualización del server de Zabbix

serverinfo.get

Este método permite recuperar la hora de la última actualización del server de Zabbix indicado.

El método está disponible para *User*, *Admin*, *Super admin* y usuarios no autenticados.

Firma del método: `object serverinfo.get(object parameters)`

Origen del método: `CServerInfo::get()` en `ui/include/classes/api/services/CServerInfo.php`

Parámetros

Parameter	Type	Description
serverid	ID	ID del server de Zabbix. Comportamiento del parámetro: - Obligatorio
output	query	Este parámetro se describe en los comentarios de referencia.

Valores devueltos

Property	Type	Description
serverid	ID	ID del server de Zabbix.
lastaccess	timestamp	Hora de la última actualización del server de Zabbix. En un clúster de alta disponibilidad (HA), muestra el valor de `lastaccess` del nodo activo.

Ejemplos

Obtener información del server

Obtiene información sobre el server de Zabbix especificado.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "serverinfo.get",
    "params": {
        "serverid": "019fb1f9-22bd-74ea-bc03-51efb8b76aba",
        "output": "extend"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "serverid": "019fb1f9-22bd-74ea-bc03-51efb8b76aba",
        "lastaccess": 1785337245
    },
    "id": 1
}
```

Interfaz del host

Esta clase proporciona acceso a las interfaces del host.

Referencias de objetos:

- Interfaz del host
  - Detalles

Método disponible:

- hostinterface.get: recuperar interfaces del host

Las interfaces del host se crean, actualizan y eliminan mediante los métodos del host.

Objeto de interfaz de equipo

Los siguientes objetos están directamente relacionados con la API `hostinterface`.

Interfaz del host

El objeto de interfaz del host tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
interfaceid	ID	ID de la interfaz. Comportamiento de la propiedad: - *solo lectura* - *obligatoria* para operaciones de actualización
available	integer	Disponibilidad de la interfaz del host. Valores posibles: 0 - *(predeterminado)* desconocida; 1 - disponible; 2 - no disponible. Comportamiento de la propiedad: - *solo lectura*

Propiedad	Tipo	Descripción
hostid	ID	ID del host al que pertenece la interfaz. Comportamiento de la propiedad: - *constante* - *obligatoria* para operaciones de creación
type	integer	Tipo de interfaz. Valores posibles: 1 - Agent; 2 - SNMP; 3 - IPMI; 4 - JMX. Comportamiento de la propiedad: - *obligatoria* para operaciones de creación
ip	string	Dirección IP utilizada por la interfaz. Puede estar vacía si la conexión se realiza mediante DNS. Comportamiento de la propiedad: - *obligatoria* para operaciones de creación
dns	string	Nombre DNS utilizado por la interfaz. Puede estar vacío si la conexión se realiza mediante IP. Comportamiento de la propiedad: - *obligatoria* para operaciones de creación
port	string	Número de puerto utilizado por la interfaz. Puede contener macros de usuario. Comportamiento de la propiedad: - *obligatoria* para operaciones de creación
useip	integer	Indica si la conexión debe realizarse mediante IP. Valores posibles: 0 - conectarse utilizando el nombre DNS del host; 1 - conectarse utilizando la dirección IP del host. Comportamiento de la propiedad: - *obligatoria* para operaciones de creación
main	integer	Indica si la interfaz se utiliza como predeterminada en el host. Solo una interfaz de cada tipo puede establecerse como predeterminada en un host. Valores posibles: 0 - no predeterminada; 1 - predeterminada. Comportamiento de la propiedad: - *obligatoria* para operaciones de creación
details	object	Objeto details adicional para la interfaz. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en "SNMP"
disable_until	timestamp	Próximo momento de sondeo de una interfaz de host no disponible. Comportamiento de la propiedad: - *solo lectura*
error	string	Texto del error si la interfaz del host no está disponible. Comportamiento de la propiedad: - *solo lectura*

Propiedad	Tipo	Descripción
errors_from	timestamp	Momento en que la interfaz del host dejó de estar disponible. Comportamiento de la propiedad: - *solo lectura*

Detalles

El objeto details tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
version	integer	Versión de la interfaz SNMP. Valores posibles: 1 - SNMPv1; 2 - SNMPv2c; 3 - SNMPv3. Comportamiento de la propiedad: - *requerido*
bulk	integer	Si se deben usar solicitudes SNMP bulk. Valores posibles: 0 - no usar solicitudes bulk; 1 - (por defecto) - usar solicitudes bulk.
community	string	Comunidad SNMP. Se usa solo en las interfaces SNMPv1 y SNMPv2. Comportamiento de la propiedad: - *requerido* si `version` está configurado como ”SNMPv1” o ”SNMPv2c”
max_repetitions	integer	Valor máximo de repetición para solicitudes SNMP bulk nativas (GetBulkRequest-PDUs). Se usa solo para items `discovery[]` y `walk[]` en SNMPv2 y v3. Por defecto: 10.
securityname	string	Nombre de seguridad SNMPv3. Se usa solo en las interfaces SNMPv3.
securitylevel	integer	Nivel de seguridad SNMPv3. Se usa solo en las interfaces SNMPv3. Valores posibles: 0 - (por defecto) - noAuthNoPriv; 1 - authNoPriv; 2 - authPriv.
authpassphrase	string	Frase de autenticación SNMPv3. Se usa solo en las interfaces SNMPv3.
privpassphrase	string	Frase de privacidad SNMPv3. Se usa solo en las interfaces SNMPv3.
authprotocol	integer	Protocolo de autenticación SNMPv3. Se usa solo en las interfaces SNMPv3. Valores posibles: 0 - (por defecto) - MD5; 1 - SHA1; 2 - SHA224; 3 - SHA256; 4 - SHA384; 5 - SHA512.

Propiedad	Tipo	Descripción
privprotocol	integer	Protocolo de privacidad SNMPv3. Se usa solo en las interfaces SNMPv3. Valores posibles: 0 - (por defecto) - DES; 1 - AES128; 2 - AES192; 3 - AES256; 4 - AES192C; 5 - AES256C.
contextname	string	Nombre de contexto SNMPv3. Se usa solo en las interfaces SNMPv3.

hostinterface.get

Descripción

`integer/array hostinterface.get(object parameters)`

El método permite recuperar las interfaces de host según los parámetros especificados.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración del rol de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(objeto)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
hostids	ID/array	Devolver solo las interfaces de host utilizadas por los hosts indicados.
interfaceids	ID/array	Devolver solo las interfaces de host con los IDs indicados.
itemids	ID/array	Devolver solo las interfaces de host utilizadas por los items indicados.
triggerids	ID/array	Devolver solo las interfaces de host utilizadas por los items en los triggers indicados.
selectItems	query	Devolver una propiedad `items` con los items que utilizan la interfaz. Admite `count`.
selectHosts	query	Devolver una propiedad `hosts` con un array de hosts que utilizan la interfaz.
limitSelects	integer	Limita el número de registros devueltos por subconsultas. Se aplica a las siguientes subconsultas: `selectItems`.
sortfield	string/array	Ordenar el resultado por las propiedades indicadas. Valores posibles: `interfaceid`, `dns`, `ip`.
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

(integer/array) Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperar interfaces de equipo

Recuperar todos los datos sobre las interfaces utilizadas por el equipo "30057".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "hostinterface.get",
    "params": {
        "output": "extend",
        "hostids": "30057"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "interfaceid": "50039",
            "hostid": "30057",
            "main": "1",
            "type": "1",
            "useip": "0",
            "ip": "",
            "dns": "localhost",
            "port": "10050",
            "available": "0",
            "error": "",
            "errors_from": "0",
            "disable_until": "0",
            "details": []
        },
        {
            "interfaceid": "55082",
            "hostid": "30057",
            "main": "1",
            "type": "2",
            "useip": "1",
            "ip": "127.0.0.1",
            "dns": "",
            "port": "161",
            "available": "0",
            "error": "",
            "errors_from": "0",
            "disable_until": "0",
            "details": {
                "version": "2",
                "bulk": "0",
                "community": "{$SNMP_COMMUNITY}",
                "max_repetitions": "10"
            }
        }
    ],
    "id": 1
}
```

Véase también

- Equipo
- Métrica

Fuente

CHostInterface::get() en *ui/include/classes/api/services/CHostInterface.php*.

Item

Esta clase está diseñada para trabajar con items.

Referencias de objetos:

- Item
  - HTTP header
  - HTTP query field
- Etiqueta de item
- Preprocesamiento de item

Métodos disponibles:

- item.create - crear nuevos items
- item.delete - eliminar items
- item.get - obtener items
- item.update - actualizar items

Objeto de métrica

Los siguientes objetos están directamente relacionados con la API de `métrica`.

Item

> **Note:**
> Los items web no se pueden crear, actualizar ni eliminar directamente mediante la API de Zabbix.

El objeto item tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
itemid	ID	ID del item. Comportamiento de la propiedad: - *solo lectura* - *obligatoria* para las operaciones de actualización

Propiedad	Tipo	Descripción
delay	string	Intervalo de actualización del item. Acepta un número de segundos (por ejemplo, 1, 60, 300), un valor con un sufijo de tiempo (por ejemplo, 30s, 1m, 2h, 1d) o, de forma opcional, uno o varios intervalos personalizados, todos separados por punto y coma. Los intervalos personalizados pueden ser una combinación de intervalos flexibles y de planificación. Acepta macros de usuario. Si se utiliza una macro, el valor debe ser una única macro. No se admiten varias macros ni macros combinadas con texto. Los intervalos flexibles se pueden escribir como dos macros separadas por una barra diagonal (por ejemplo, {$FLEX_INTERVAL}/{$FLEX_PERIOD}). Ejemplo: 1h;wd1-5h9-18;{$Macro1}/1-7,00:00-24:00;0/6-7,12:00-24:00;{$Macr Comportamiento de la propiedad: - *obligatoria* si type está establecido en ”Zabbix agent” (0), ”Simple check” (3), ”Zabbix internal” (5), ”External check” (10), ”Database monitor” (11), ”IPMI agent” (12), ”SSH agent” (13), ”TELNET agent” (14), ”Calculated” (15), ”JMX agent” (16), ”HTTP agent” (19), ”SNMP agent” (20), ”Script” (21), ”Browser” (22), ”Telemetry query” (24), o si type está establecido en ”Zabbix agent (active)” (7) y key_ no contiene ”mqtt.get”
hostid	ID	ID del host o template al que pertenece el item. Comportamiento de la propiedad: - *constante* - *obligatoria* para las operaciones de creación
interfaceid	ID	ID de la interfaz del host del item. Comportamiento de la propiedad: - *obligatoria* si el item pertenece a un host y type está establecido en ”Zabbix agent”, ”IPMI agent”, ”JMX agent”, ”SNMP trap” o ”SNMP agent” - *compatible* si el item pertenece a un host y type está establecido en ”Simple check”, ”External check”, ”SSH agent”, ”TELNET agent” o ”HTTP agent” - *solo lectura* para objetos descubiertos
key_	string	Clave del item. Comportamiento de la propiedad: - *obligatoria* para las operaciones de creación - *solo lectura* para objetos heredados o descubiertos
name	string	Nombre del item. Admite macros de usuario. Comportamiento de la propiedad: - *obligatoria* para las operaciones de creación - *solo lectura* para objetos heredados o descubiertos
name_resolved	string	Nombre del item con las macros de usuario resueltas. Comportamiento de la propiedad: - *solo lectura*

Propiedad	Tipo	Descripción
type	integer	Tipo del item. Valores posibles: 0 - Zabbix agent; 2 - Zabbix trapper; 3 - Simple check; 5 - Zabbix internal; 7 - Zabbix agent (active); 9 - Web item; 10 - External check; 11 - Database monitor; 12 - IPMI agent; 13 - SSH agent; 14 - TELNET agent; 15 - Calculated; 16 - JMX agent; 17 - SNMP trap; 18 - Dependent item; 19 - HTTP agent; 20 - SNMP agent; 21 - Script; 22 - Browser; 24 - Telemetry query. Comportamiento de la propiedad: - *obligatoria* para las operaciones de creación - *solo lectura* para objetos heredados o descubiertos
url	string	Cadena de URL. Admite macros de usuario, {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {HOST.PORT}, {ITEM.ID}, {ITEM.KEY}. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados o descubiertos
value_type	integer	Tipo de información del item. Valores posibles: 0 - numérico flotante; 1 - carácter; 2 - registro; 3 - numérico sin signo; 4 - texto; 5 - binario; 6 - JSON. Comportamiento de la propiedad: - *obligatoria* para las operaciones de creación - *solo lectura* para objetos heredados o descubiertos
allow_traps	integer	Permite rellenar el valor de forma similar al item trapper. 0 - *(predeterminado)* No aceptar datos entrantes; 1 - Aceptar datos entrantes. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos descubiertos

Propiedad	Tipo	Descripción
authtype	integer	Método de autenticación. Valores posibles si `type` está establecido en ”SSH agent”: 0 - *(predeterminado)* contraseña; 1 - clave pública. Valores posibles si `type` está establecido en ”HTTP agent”: 0 - *(predeterminado)* ninguno; 1 - básico; 2 - NTLM; 3 - Kerberos; 4 - Digest. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”SSH agent” o ”HTTP agent” - *solo lectura* para objetos heredados (si `type` está establecido en ”HTTP agent”) u objetos descubiertos
description	string	Descripción del item. Comportamiento de la propiedad: - *solo lectura* para objetos descubiertos
error	string	Texto del error si hay problemas al actualizar el valor del item. Comportamiento de la propiedad: - *solo lectura*
flags	integer	Origen del item. Valores posibles: 0 - un item normal; 4 - un item convertido a partir de un prototipo. Comportamiento de la propiedad: - *solo lectura*
follow_redirects	integer	Seguir las redirecciones de respuesta durante el sondeo de datos. Valores posibles: 0 - No seguir redirecciones; 1 - *(predeterminado)* Seguir redirecciones. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados o descubiertos
headers	array	Matriz de encabezados que se enviarán al realizar una solicitud HTTP. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados o descubiertos
history	string	Unidad de tiempo durante la que se deben conservar los datos del historial. También acepta una macro de usuario. Predeterminado: 31d. Comportamiento de la propiedad: - *solo lectura* para objetos descubiertos
http_proxy	string	Cadena de conexión del proxy HTTP(S). Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados o descubiertos

Propiedad	Tipo	Descripción
inventory_link	integer	ID del campo del inventario del host que rellena el item. Consulte la página del inventario del host para obtener una lista de los campos de inventario del host compatibles y sus ID. Predeterminado: 0. Comportamiento de la propiedad: - *compatible* si `value_type` está establecido en ”numeric float”, ”character”, ”numeric unsigned” o ”text” - *solo lectura* para objetos descubiertos
ipmi_sensor	string	Sensor IPMI. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”IPMI agent” y `key_` no está establecido en ”ipmi.get” - *compatible* si `type` está establecido en ”IPMI agent” - *solo lectura* para objetos heredados o descubiertos
jmx_endpoint	string	Cadena de conexión personalizada del JMX agent. Valor predeterminado: service:jmx:rmi:///jndi/rmi://{HOST.CONN}:{HOST.PORT}/jmxrmi Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”JMX agent” - *solo lectura* para objetos descubiertos
lastclock	timestamp	Momento en que se actualizó por última vez el valor del item. De forma predeterminada, solo se muestran los valores correspondientes a las últimas 24 horas. Puede ampliar este período cambiando el valor del parámetro *Max history display period* en la sección de menú *Administration > General*. Comportamiento de la propiedad: - *solo lectura*
lastns	integer	Nanosegundos del momento en que se actualizó por última vez el valor del item. De forma predeterminada, solo se muestran los valores correspondientes a las últimas 24 horas. Puede ampliar este período cambiando el valor del parámetro *Max history display period* en la sección de menú *Administration > General*. Comportamiento de la propiedad: - *solo lectura*
lastvalue	string	Último valor del item. De forma predeterminada, solo se muestran los valores correspondientes a las últimas 24 horas. Puede ampliar este período cambiando el valor del parámetro *Max history display period* en la sección de menú *Administration > General*. Comportamiento de la propiedad: - *solo lectura*
logtimefmt	string	Formato de la hora en las entradas del registro. Comportamiento de la propiedad: - *compatible* si `value_type` está establecido en ”log” - *solo lectura* para objetos heredados o descubiertos

Propiedad	Tipo	Descripción
master_itemid	ID	ID del item maestro. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”Dependent item” - *solo lectura* para objetos heredados o descubiertos
output_format	integer	Indica si la respuesta debe convertirse a JSON. 0 - *(predeterminado)* Almacenar sin procesar; 1 - Convertir a JSON. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados o descubiertos
params	string	Parámetros adicionales según el tipo de item: - script ejecutado para items SSH agent y TELNET agent; - consulta SQL para items Database monitor; - fórmula para items Calculated; - script para items Script y Browser. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”Database monitor”, ”SSH agent”, ”TELNET agent”, ”Calculated”, ”Script” o ”Browser” - *solo lectura* para objetos heredados (si `type` está establecido en ”Script” o ”Browser”) u objetos descubiertos
parameters	object/array	Parámetros adicionales si `type` está establecido en ”Script” o ”Browser”. Matriz de objetos con las propiedades `name` y `value`, donde `name` debe ser único. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”Script” o ”Browser” - *solo lectura* para objetos heredados o descubiertos
password	string	Contraseña para la autenticación. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”JMX agent” y `username` está establecido - *compatible* si `type` está establecido en ”Simple check”, ”SSH agent”, ”TELNET agent”, ”Database monitor” o ”HTTP agent” - *solo lectura* para objetos heredados (si `type` está establecido en ”HTTP agent”) u objetos descubiertos
post_type	integer	Tipo del cuerpo de datos de publicación almacenado en la propiedad `posts`. Valores posibles: 0 - *(predeterminado)* Datos sin procesar; 2 - Datos JSON; 3 - Datos XML. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados o descubiertos
posts	string	Datos del cuerpo de la solicitud HTTP(S). Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”HTTP agent” y `post_type` está establecido en ”JSON data” o ”XML data” - *compatible* si `type` está establecido en ”HTTP agent” y `post_type` está establecido en ”Raw data” - *solo lectura* para objetos heredados o descubiertos

Propiedad	Tipo	Descripción
prevvalue	string	Valor anterior del item. De forma predeterminada, solo se muestran los valores correspondientes a las últimas 24 horas. Puede ampliar este período cambiando el valor del parámetro *Max history display period* en la sección de menú *Administration > General*. Comportamiento de la propiedad: - *solo lectura*
privatekey	string	Nombre del archivo de clave privada. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”SSH agent” y `authtype` está establecido en ”public key” - *solo lectura* para objetos descubiertos
publickey	string	Nombre del archivo de clave pública. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”SSH agent” y `authtype` está establecido en ”public key” - *solo lectura* para objetos descubiertos
query_fields	array	Matriz de campos de consulta que se enviarán al realizar una solicitud HTTP. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados o descubiertos
request_method	integer	Tipo de método de solicitud. Valores posibles: 0 - *(predeterminado)* GET; 1 - POST; 2 - PUT; 3 - HEAD. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados o descubiertos
retrieve_mode	integer	Parte de la respuesta que se debe almacenar. Valores posibles si `request_method` está establecido en ”GET”, ”POST” o ”PUT”: 0 - *(predeterminado)* Cuerpo; 1 - Encabezados; 2 - Se almacenarán el cuerpo y los encabezados. Valores posibles si `request_method` está establecido en ”HEAD”: 1 - Encabezados. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados o descubiertos
snmp_oid	string	OID de SNMP. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”SNMP agent” - *solo lectura* para objetos heredados o descubiertos
ssl_cert_file	string	Ruta del archivo de clave pública SSL. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados o descubiertos

Propiedad	Tipo	Descripción
ssl_key_file	string	Ruta del archivo de clave privada SSL. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados o descubiertos
ssl_key_password	string	Contraseña del archivo de clave SSL. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados o descubiertos
state	integer	Estado del item. Valores posibles: 0 - *(predeterminado)* normal; 1 - no compatible. Comportamiento de la propiedad: - *solo lectura*
status	integer	Estado del item. Valores posibles: 0 - *(predeterminado)* item habilitado; 1 - item deshabilitado.
status_codes	string	Rangos de códigos de estado HTTP requeridos, separados por comas. También admite macros de usuario como parte de una lista separada por comas. Ejemplo: 200,200-{$M},{$M},200-400 Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados o descubiertos
templateid	ID	ID del item del template principal. *Sugerencia*: Utilice la propiedad `hostid` para especificar el template al que pertenece el item. Comportamiento de la propiedad: - *solo lectura*
timeout	string	Tiempo de espera de la solicitud de sondeo de datos del item. Acepta un número de segundos (por ejemplo, 1, 60, 300) o un valor con un sufijo de tiempo (por ejemplo, 300s, 1m). También acepta macros de usuario. El rango de valores posibles es: 1-600s. Predeterminado: ”” - utilizar la configuración del proxy/global. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”Zabbix agent” (0), ”Simple check” (3) y `key_` no comienza por ”`vmware.`” ni ”`icmpping`”, ”Zabbix agent (active)” (7), ”External check” (10), ”Database monitor” (11), ”SSH agent” (13), ”TELNET agent” (14), ”HTTP agent” (19), ”SNMP agent” (20) y `snmp_oid` comienza por ”`walk[`” o ”`get[`”, ”Script” (21), ”Browser” (22) - *solo lectura* para objetos heredados y descubiertos, - si se aplica el valor predeterminado y el item pertenece a un host supervisado por un proxy o grupo de proxies inaccesible, se utiliza el tiempo de espera global.

Propiedad	Tipo	Descripción
trapper_hosts	string	Hosts permitidos. Comportamiento de la propiedad: - *solo lectura* para objetos descubiertos - *compatible* si `type` está establecido en ”Zabbix trapper”, o si `type` está establecido en ”HTTP agent” y `allow_traps` está establecido en ”Allow incoming data”. Si se omite la propiedad, se establece automáticamente en la macro de usuario `{$TRAPPER.ALLOWED_HOSTS}` (valor predeterminado). De forma predeterminada, `{$TRAPPER.ALLOWED_HOSTS}` es `127.0.0.1,::1`. Si la propiedad se establece en una cadena vacía, se denegarán las conexiones entrantes de todos los hosts.
trends	string	Unidad de tiempo durante la que se deben conservar los datos de tendencias. También acepta una macro de usuario. Predeterminado: `365d`. Comportamiento de la propiedad: - *compatible* si `value_type` está establecido en ”numeric float” o ”numeric unsigned” - *solo lectura* para objetos descubiertos
units	string	Unidades del valor. Comportamiento de la propiedad: - *compatible* si `value_type` está establecido en ”numeric float” o ”numeric unsigned” - *solo lectura* para objetos heredados o descubiertos
username	string	Nombre de usuario para la autenticación. Comportamiento de la propiedad: - *obligatorio* si `type` está establecido en ”SSH agent”, ”TELNET agent”, o si `type` está establecido en ”JMX agent” y `password` está establecido - *compatible* si `type` está establecido en ”Simple check”, ”Database monitor” o ”HTTP agent” - *solo lectura* para objetos heredados (si `type` está establecido en ”HTTP agent”) u objetos descubiertos
uuid	string	Identificador único universal, utilizado para vincular el item importado con otros ya existentes. Se genera automáticamente si no se proporciona. Comportamiento de la propiedad: - *compatible* si el item pertenece a un template
valuemapid	ID	ID del mapa de valores asociado. Comportamiento de la propiedad: - *compatible* si `value_type` está establecido en ”numeric float”, ”character” o ”numeric unsigned” - *solo lectura* para objetos heredados o descubiertos
verify_host	integer	Indica si se debe validar que el nombre del host de la conexión coincida con el del certificado del host. Valores posibles: 0 - *(predeterminado)* No validar; 1 - Validar. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados o descubiertos

Propiedad	Tipo	Descripción
verify_peer	integer	Indica si se debe validar que el certificado del host sea auténtico. Valores posibles: 0 - *(predeterminado)* No validar; 1 - Validar. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados o descubiertos
query	object	Configuración de la consulta de telemetría. Define qué consulta el item en el origen de datos de telemetría, cómo filtra los datos y cómo agrega el resultado. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”Telemetry query” - *solo lectura* para objetos heredados o descubiertos
time_shift	string	Hasta qué punto hacia atrás (antes de la hora actual) termina el intervalo de tiempo que consulta el item. Acepta un número de segundos (por ejemplo, 1, 60, 300) o un valor con un sufijo de tiempo (por ejemplo, 30s, 1m, 2h, 1d). También acepta macros de usuario. El rango de valores posibles es: 0-86400s. Predeterminado: 15s. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”Telemetry query”
lookback_limit	string	Hasta qué punto hacia atrás (antes de la hora actual) consulta el item los datos cuando no existe una consulta anterior desde la que continuar (por ejemplo, en la primera ejecución del item), o cuando la consulta anterior es más antigua que `delay` (por ejemplo, porque Zabbix server estaba inactivo). Acepta un número de segundos (por ejemplo, 1, 60, 300) o un valor con un sufijo de tiempo (por ejemplo, 30s, 1m, 2h, 1d). También acepta macros de usuario. El rango de valores posibles es: 1-259200s. Predeterminado: 10m. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”Telemetry query”
granularity	string	Duración de cada segmento en el que se divide el intervalo de tiempo completo de la consulta. Acepta un número de segundos (por ejemplo, 1, 60, 300) o un valor con un sufijo de tiempo (por ejemplo, 30s, 1m, 2h, 1d). También acepta macros de usuario. El rango de valores posibles es: 1-86400s. Predeterminado: 15s. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”Telemetry query”

Encabezado HTTP

El objeto header tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
name	string	Nombre del encabezado HTTP. Comportamiento de la propiedad: - *obligatorio*
value	string	Valor del encabezado. Comportamiento de la propiedad: - *obligatorio*

Campo de consulta HTTP

El objeto de campo de consulta define un nombre y un valor que se utilizan para especificar un parámetro de URL. Tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
name	string	Nombre del parámetro. Comportamiento de la propiedad: - *obligatorio*
value	string	Valor del parámetro. Comportamiento de la propiedad: - *obligatorio*

Consulta de telemetría

El objeto de consulta de telemetría tiene las siguientes propiedades.

Property	Type	Description
signal_type	integer	Tipo de señal que consulta el item. Valores posibles: 0 - Trazas; 1 - Métricas; 2 - Registros. Comportamiento de la propiedad: - *required*
metric_point_type	integer	Tipo de punto de métrica que consulta el item. Valores posibles: 0 - Suma; 1 - Medidor; 2 - Histograma; 3 - Histograma exponencial. Comportamiento de la propiedad: - *required* si `signal_type` está establecido en ”Métricas”

Property	Type	Description
columns	array	Matriz de columnas de datos de telemetría que consulta el item. Puede estar vacía. Comportamiento de la propiedad: - *required*
aggregated_columns	array	Matriz de columnas agregadas de datos de telemetría que consulta el item y de cómo calcula un valor a partir de los registros de las columnas. Comportamiento de la propiedad: - *required*
filter	object	Filtro de consulta de telemetría para los registros de columnas de la consulta. Comportamiento de la propiedad: - *required*

Columnas de datos de telemetría

El objeto de columnas de datos de telemetría tiene las siguientes propiedades.

Property	Type	Description
column	string	Nombre de la columna de datos de telemetría que consulta el item. Consulte columnas de datos de telemetría compatibles. Para varias columnas de valor único, cada valor de `column` debe ser único. Para varias columnas que contienen pares clave-valor, cada par de valores `column` y `attribute_key` debe ser único.
attribute_key	string	Nombre de la clave que consulta el item dentro de la columna que contiene pares clave-valor. Para varias columnas que contienen pares clave-valor, cada par de valores `column` y `attribute_key` debe ser único. Comportamiento de la propiedad: - *obligatorio* si `column` está establecido en una columna que contiene pares clave-valor

Columnas agregadas de datos de telemetría

El objeto de columnas agregadas de datos de telemetría tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
column	string	Nombre de la columna de datos de telemetría que consulta el item. Consulte columnas de datos de telemetría compatibles. Para varias columnas, cada valor de `column` debe ser único. Esta propiedad no es compatible si `function` está establecido en "Count". Comportamiento de la propiedad: - *obligatoria* si `function` está establecido en "Min", "Max", "Average", "Sum" o "Percentile"

Propiedad	Tipo	Descripción
function	integer	Función que el item aplica a los registros de la columna. Valores posibles: 1 - Min; 2 - Max; 3 - Average; 4 - Count; 5 - Sum; 8 - Percentile. Comportamiento de la propiedad: - *obligatoria*
parameters	array	Percentil que el item calcula a partir de los registros de la columna. Valor posible: un número del 0 al 100, indicado como una cadena. Comportamiento de la propiedad: - *obligatoria* si `function` está establecido en "Percentile"
alias	string	Nombre único del valor calculado, utilizado para identificarlo en el valor del item. Cada valor de `alias` debe ser único. Comportamiento de la propiedad: - *obligatoria*

Filtro de datos de telemetría

El objeto de filtro de datos de telemetría tiene las siguientes propiedades.

Property	Type	Description
conditions	array	Condiciones del filtro de datos de telemetría que debe cumplir un registro de una columna de datos de telemetría. Comportamiento de la propiedad: - *required*
evaltype	integer	Método de evaluación de la condición del filtro. Valores posibles: 0 - And/Or; 1 - And; 2 - Or; 3 - Expresión personalizada. Comportamiento de la propiedad: - *required*
eval_formula	string	Expresión generada que se utilizará para evaluar las condiciones del filtro. La expresión contiene identificadores que hacen referencia a condiciones de filtro específicas mediante su `formulaid`. El valor de `eval_formula` es igual al valor de `formula` para los filtros con una expresión personalizada. Comportamiento de la propiedad: - *read-only*

Property	Type	Description
formula	string	Expresión definida por el usuario que se utilizará para evaluar las condiciones de los filtros con una expresión personalizada. La expresión debe contener identificadores que hagan referencia a condiciones de filtro específicas mediante su `formulaid`. Los identificadores utilizados en la expresión deben coincidir exactamente con los definidos en las condiciones del filtro: no puede quedar ninguna condición sin utilizar ni omitirse ninguna. Comportamiento de la propiedad: - *required* si `evaltype` está establecido en "Expresión personalizada"

Condiciones de filtrado de datos de telemetría

El objeto de condiciones de filtrado de datos de telemetría tiene las siguientes propiedades.

Property	Type	Description
column	string	Nombre de la columna de datos de telemetría a la que se aplica la condición de filtrado. Consulte las columnas de datos de telemetría compatibles. Comportamiento de la propiedad: - *obligatoria*
attribute_key	string	Nombre de la clave a la que se aplica la condición, dentro de la columna que contiene pares clave-valor. Comportamiento de la propiedad: - *obligatoria* si `column` está configurada con una columna que contiene pares clave-valor
operator	integer	Operador de la condición de filtrado. Valores posibles para columnas de valor único: 0 - Es igual a; 1 - No es igual a; 2 - Contiene; 3 - No contiene. Valores posibles para columnas que contienen pares clave-valor: 0 - Es igual a; 1 - No es igual a; 12 - Existe. Comportamiento de la propiedad: - *obligatoria*
value	string	Valor con el que se debe comparar. Comportamiento de la propiedad: - *obligatoria* si `operator` está configurado como "Es igual a", "No es igual a", "Contiene" o "No contiene"
formulaid	string	ID único arbitrario que se utiliza para hacer referencia a la condición desde una expresión personalizada. Solo puede contener letras mayúsculas. El usuario debe definir el ID al modificar las condiciones de filtrado, pero se generará uno nuevo al solicitarlas posteriormente. Comportamiento de la propiedad: - *obligatoria* si `evaltype` del objeto Filtro de consulta de telemetría está configurado como "Expresión personalizada"

Etiqueta de item

El objeto de etiqueta de item tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
tag	string	Nombre de la etiqueta del item. Comportamiento de la propiedad: - *obligatorio*
value	string	Valor de la etiqueta del item. Comportamiento de la propiedad: - *solo lectura*.
object	integer	Tipo de objeto del que se heredó la etiqueta. Valores posibles: 0 - template; 1 - host. Comportamiento de la propiedad: - *solo lectura*.
objectid	ID	ID del objeto del que se heredó la etiqueta. Comportamiento de la propiedad: - *solo lectura*.

Preprocesamiento de item

El objeto de preprocesamiento de item tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
type	integer	El tipo de opción de preprocesamiento. Valores posibles: 1 - Multiplicador personalizado; 2 - Recortar por la derecha; 3 - Recortar por la izquierda; 4 - Recortar; 5 - Expresión regular; 6 - Booleano a decimal; 7 - Octal a decimal; 8 - Hexadecimal a decimal; 9 - Cambio simple; 10 - Cambio por segundo; 11 - XPath XML; 12 - JSONPath; 13 - En rango; 14 - Coincide con la expresión regular; 15 - No coincide con la expresión regular; 16 - Comprobar errores en JSON; 17 - Comprobar errores en XML; 18 - Comprobar errores mediante una expresión regular; 19 - Descartar sin cambios; 20 - Descartar sin cambios con heartbeat; 21 - JavaScript; 22 - Patrón de Prometheus; 23 - Prometheus a JSON; 24 - CSV a JSON; 25 - Reemplazar; 26 - Comprobar no compatible; 27 - XML a JSON; 28 - Valor de recorrido SNMP; 29 - Recorrido SNMP a JSON; 30 - Obtener valor mediante SNMP. Comportamiento de la propiedad: - *obligatoria*
params	string	Parámetros adicionales utilizados por la opción de preprocesamiento. Varios parámetros se separan mediante el carácter de nueva línea (\n). Si `type` se establece en ”Comprobar no compatible”, los parámetros siguen la sintaxis `<scope>[\n<pattern>]`, donde *pattern* es una expresión regular y *scope* es uno de los siguientes: -1 - coincide con cualquier error; 0 - comprobar si el mensaje de error coincide con *pattern*; 1 - comprobar si el mensaje de error no coincide con *pattern*. Comportamiento de la propiedad: - *obligatoria* si `type` se establece en ”Multiplicador personalizado” (1), ”Recortar por la derecha” (2), ”Recortar por la izquierda” (3), ”Recortar” (4), ”Expresión regular” (5), ”XPath XML” (11), ”JSONPath” (12), ”En rango” (13), ”Coincide con la expresión regular” (14), ”No coincide con la expresión regular” (15), ”Comprobar errores en JSON” (16), ”Comprobar errores en XML” (17), ”Comprobar errores mediante una expresión regular” (18), ”Descartar sin cambios con heartbeat” (20), ”JavaScript” (21), ”Patrón de Prometheus” (22), ”Prometheus a JSON” (23), ”CSV a JSON” (24), ”Reemplazar” (25), Comprobar no compatible (26), ”Valor de recorrido SNMP” (28), ”Recorrido SNMP a JSON” (29) o ”Obtener valor mediante SNMP” (30)

Propiedad	Tipo	Descripción
error_handler	integer	Tipo de acción utilizado en caso de fallo del paso de preprocesamiento. Valores posibles: 0 - El mensaje de error lo establece Zabbix server; 1 - Descartar el valor; 2 - Establecer un valor personalizado; 3 - Establecer un mensaje de error personalizado. Valores posibles si `type` se establece en ”Comprobar no compatible”: 1 - Descartar el valor; 2 - Establecer un valor personalizado; 3 - Establecer un mensaje de error personalizado. Comportamiento de la propiedad: - *obligatoria* si `type` se establece en ”Multiplicador personalizado” (1), ”Expresión regular” (5), ”Booleano a decimal” (6), ”Octal a decimal” (7), ”Hexadecimal a decimal” (8), ”Cambio simple” (9), ”Cambio por segundo” (10), ”XPath XML” (11), ”JSONPath” (12), ”En rango” (13), ”Coincide con la expresión regular” (14), ”No coincide con la expresión regular” (15), ”Comprobar errores en JSON” (16), ”Comprobar errores en XML” (17), ”Comprobar errores mediante una expresión regular” (18), ”Patrón de Prometheus” (22), ”Prometheus a JSON” (23), ”CSV a JSON” (24), ”Comprobar no compatible” (26), ”XML a JSON” (27), ”Valor de recorrido SNMP” (28), ”Recorrido SNMP a JSON” (29) u ”Obtener valor mediante SNMP” (30)
error_handler_params	string	Parámetros del gestor de errores. Comportamiento de la propiedad: - *obligatoria* si `error_handler` se establece en ”Establecer un valor personalizado” o ”Establecer un mensaje de error personalizado”

Los siguientes parámetros y gestores de errores son compatibles con cada tipo de preprocesamiento.

Tipo de preprocesamiento	Nombre	Parámetro 1	Parámetro 2	Parámetro 3	Gestores de errores compatibles
1	Multiplicador personalizado	número^1, 6^			0, 1, 2, 3
2	Recortar por la derecha	lista de caracteres[2]			
3	Recortar por la izquierda	lista de caracteres[2]			
4	Recortar	lista de caracteres[2]			
5	Expresión regular	patrón[3]	salida[2]		0, 1, 2, 3
6	Booleano a decimal				0, 1, 2, 3

Tipo de preprocesamiento	Nombre	Parámetro 1	Parámetro 2	Parámetro 3	Gestores de errores compatibles
7	Octal a decimal				0, 1, 2, 3
8	Hexadecimal a decimal				0, 1, 2, 3
9	Cambio simple				0, 1, 2, 3
10	Cambio por segundo				0, 1, 2, 3
11	XPath XML	ruta[4]			0, 1, 2, 3
12	JSONPath	ruta[4]			0, 1, 2, 3
13	En rango	mín.^1, 6^	máx.^1, 6^		0, 1, 2, 3
14	Coincide con la expresión regular	patrón[3]			0, 1, 2, 3
15	No coincide con la expresión regular	patrón[3]			0, 1, 2, 3
16	Comprobar errores en JSON	ruta[4]			0, 1, 2, 3
17	Comprobar errores en XML	ruta[4]			0, 1, 2, 3
18	Comprobar errores mediante una expresión regular	patrón[3]	salida[2]		0, 1, 2, 3

Tipo de preprocesamiento	Nombre	Parámetro 1	Parámetro 2	Parámetro 3	Gestores de errores compatibles
19	Descartar sin cambios				
20	Descartar sin cambios con heartbeat	segundos^5, 6^			
21	JavaScript	script[2]			
22	Patrón de Prometheus	patrón^6, 7^	`value`, `label`, `function`	salida^8, 9^	0, 1, 2, 3
23	Prometheus a JSON	patrón^6, 7^			0, 1, 2, 3
24	CSV a JSON	carácter[2]	carácter[2]	0,1	0, 1, 2, 3
25	Reemplazar	cadena de búsqueda[2]	reemplazo[2]		
26	Comprobar no compatible	ámbito[1]	patrón^3, 6^		1, 2, 3
27	XML a JSON				0, 1, 2, 3
28	Valor de recorrido SNMP	OID[2]	Formato: 0 - Sin cambios 1 - UTF-8 desde Hex-STRING 2 - MAC desde Hex-STRING 3 - Entero desde BITS		0, 1, 2, 3
29	Recorrido SNMP a JSON[10]	Nombre del campo[2]	Prefijo OID[2]	Formato: 0 - Sin cambios 1 - UTF-8 desde Hex-STRING 2 - MAC desde Hex-STRING 3 - Entero desde BITS	0, 1, 2, 3
30	Obtener valor mediante SNMP	Formato: 1 - UTF-8 desde Hex-STRING 2 - MAC desde Hex-STRING 3 - Entero desde BITS			0, 1, 2, 3

[1] número de coma flotante (los enteros se convierten implícitamente en valores de coma flotante)
 [2] cadena
 [3] expresión regular
 [4] JSONPath o XPath XML
 [5] entero positivo (compatible con sufijos de tiempo, por ejemplo, `30s`, `1m`, `2h`, `1d`)
 [6] macro de usuario
 [7] patrón de Prometheus que sigue la sintaxis: `<metric name>{<label name>="<label value>", ...} == <value>`. Cada componente del patrón de Prometheus (métrica, nombre de etiqueta, valor de etiqueta y valor de métrica) puede ser una macro de usuario.
 [8] salida de Prometheus que sigue la sintaxis: `<label name>` (puede ser una macro

de usuario) si `label` está seleccionado como segundo parámetro.
 [9] Una de las funciones de agregación: `sum`, `min`, `max`, `avg`, `count` si `function` está seleccionado como segundo parámetro.
 [10] Admite varios registros de "Nombre del campo,Prefijo OID,Formato" delimitados por un carácter de nueva línea.

item.create

Descripción

`object item.create(object/array items)`

Este método le permite crear nuevos items.

> **Note:**
> Los web items no se pueden crear mediante la API de Zabbix.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(objeto/array)` Items a crear.

Además de las propiedades estándar del item, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
preprocessing	array	Opciones de preprocesamiento del item.
tags	array	Etiquetas del item.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los items creados bajo la propiedad `itemids`. El orden de los IDs devueltos coincide con el orden de los items pasados.

Ejemplos

Creando una métrica

Cree una métrica numérica de Zabbix agent con 2 etiquetas de métrica para monitorizar el espacio libre en disco en el equipo con ID "30074".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "item.create",
    "params": {
        "name": "Espacio libre en disco en /home/joe/",
        "key_": "vfs.fs.size[/home/joe/,free]",
        "hostid": "30074",
        "type": 0,
        "value_type": 3,
        "interfaceid": "30084",
        "tags": [
            {
                "tag": "component",
                "value": "storage"
            },
            {
                "tag": "equipment",
                "value": "workstation"
            }
        ],
        "delay": "30s"
    },
```

```
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "24758"
        ]
    },
    "id": 1
}
```

Creando una métrica de inventario de equipo

Cree una métrica de agente Zabbix para rellenar el campo de inventario "SO" del equipo.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "item.create",
    "params": {
        "name": "uname",
        "key_": "system.uname",
        "hostid": "30021",
        "type": 0,
        "interfaceid": "30007",
        "value_type": 1,
        "delay": "10s",
        "inventory_link": 5
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "24759"
        ]
    },
    "id": 1
}
```

Crear un item con preprocesamiento

Crear un item usando un multiplicador personalizado.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "item.create",
    "params": {
        "name": "Device uptime",
        "key_": "sysUpTime",
        "hostid": "10084",
        "type": 20,
        "snmp_oid": "SNMPv2-MIB::sysUpTime.0",
        "value_type": 1,
        "delay": "60s",
        "interfaceid": "83",
        "preprocessing": [
```

```
            {
                "type": 1,
                "params": "0.01",
                "error_handler": 1,
                "error_handler_params": ""
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "44210"
        ]
    },
    "id": 1
}
```

Creación de un item dependiente

Cree un item dependiente para el item maestro con ID 24759. Solo se permiten dependencias en el mismo host, por lo tanto, el item maestro y el dependiente deben tener el mismo hostid.

Petición:

```
{
    "jsonrpc": "2.0",
    "method": "item.create",
    "params": {
      "hostid": "30074",
      "name": "Dependent test item",
      "key_": "dependent.item",
      "type": 18,
      "master_itemid": "24759",
      "value_type": 2
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "44211"
        ]
    },
    "id": 1
}
```

Crear métrica de agente HTTP

Crear una métrica de método de solicitud POST con preprocesamiento de respuesta JSON.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "item.create",
    "params": {
        "url":"http://127.0.0.1/http.php",
        "query_fields": [
```

```
            {
                "name": "mode",
                "value": "json"
            },
            {
                "name": "min",
                "value": "10"
            },
            {
                "name": "max",
                "value": "100"
            }
        ],
        "interfaceid": "1",
        "type": 19,
        "hostid": "10254",
        "delay": "5s",
        "key_": "json",
        "name": "Ejemplo de agente HTTP JSON",
        "value_type": 0,
        "output_format": 1,
        "preprocessing": [
            {
                "type": 12,
                "params": "$.random",
                "error_handler": 0,
                "error_handler_params": ""
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "23865"
        ]
    },
    "id": 1
}
```

Crear métrica de script

Cree una recopilación de datos simple utilizando una métrica de script.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "item.create",
    "params": {
        "name": "Ejemplo de script",
        "key_": "custom.script.item",
        "hostid": "12345",
        "type": 21,
        "value_type": 4,
        "params": "var request = new HttpRequest();\nreturn request.post(\"https://postman-echo.com/post\"
        "parameters": [
            {
                "name": "host",
                "value": "{HOST.CONN}"
```

```
            }
        ],
        "timeout": "6s",
        "delay": "30s"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "23865"
        ]
    },
    "id": 1
}
```

Fuente

CItem::create() en *ui/include/classes/api/services/CItem.php*.

item.delete

Descripción

`object item.delete(array itemIds)`

Este método le permite eliminar items.

Note:
Los web items no se pueden eliminar a través de la API de Zabbix.

Note:
Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(array)` IDs de las métricas a eliminar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las métricas eliminadas bajo la propiedad `itemids`.

Ejemplos

Eliminando múltiples items

Elimine dos items.
 Los items dependientes y los prototipos de items se eliminan automáticamente si se elimina el item principal.

Request:

```
{
    "jsonrpc": "2.0",
    "method": "item.delete",
    "params": [
        "22982",
        "22986"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "22982",
            "22986"
        ]
    },
    "id": 1
}
```

Fuente

CItem::delete() en *ui/include/classes/api/services/CItem.php*.

item.get

Descripción

`integer/array item.get(object parameters)`

El método permite recuperar items según los parámetros proporcionados.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
itemids	ID/array	Devolver solo los items con los IDs dados.
groupids	ID/array	Devolver solo los items que pertenecen a los hosts de los grupos dados.
templateids	ID/array	Devolver solo los items que pertenecen a las templates dadas.
hostids	ID/array	Devolver solo los items que pertenecen a los hosts dados.
proxyids	ID/array	Devolver solo los items que son monitorizados por los proxies dados.
interfaceids	ID/array	Devolver solo los items que usan las interfaces de host dadas.
graphids	ID/array	Devolver solo los items que se usan en los gráficos dados.
triggerids	ID/array	Devolver solo los items que se usan en los triggers dados.
webitems	flag	Incluir items web en el resultado.
inherited	boolean	Si se establece en `true`, devuelve solo los items heredados de una template.
inheritedTags	boolean	Devolver items que tengan las `tags` dadas también en host/template/templates enlazadas. Valores posibles: `true` - la template/host/templates enlazadas también deben tener las tags dadas; `false` - *(por defecto)* se ignoran las tags de template/host/templates enlazadas.
templated	boolean	Si se establece en `true`, devuelve solo los items que pertenecen a templates.
monitored	boolean	Si se establece en `true`, devuelve solo los items habilitados que pertenecen a hosts monitorizados.
group	string	Devolver solo los items que pertenecen a un grupo con el nombre dado.
host	string	Devolver solo los items que pertenecen a un host con el nombre dado.

Parámetro	Tipo	Descripción
evaltype	integer	Método de evaluación de tags. Valores posibles: 0 - *(por defecto)* And/Or; 2 - Or.
tags	array	Devolver solo los items con las tags dadas. Formato: `[{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]`. Un array vacío devuelve todos los items. Valores posibles de operator: 0 - *(por defecto)* Contiene; 1 - Igual a; 2 - No contiene; 3 - No es igual a; 4 - Existe; 5 - No existe.
with_triggers	boolean	Si se establece en `true`, devuelve solo los items que se usan en triggers.
selectHosts	query	Devuelve una propiedad `hosts` con un array de hosts a los que pertenece el item.
selectInterfaces	query	Devuelve una propiedad `interfaces` con un array de interfaces de host usadas por el item.
selectTriggers	query	Devuelve una propiedad `triggers` con los triggers en los que se usa el item. Soporta `count`.
selectGraphs	query	Devuelve una propiedad `graphs` con los gráficos que contienen el item. Soporta `count`.
selectDiscoveryData	query	Devuelve una propiedad `discoveryData` con los datos del objeto de descubrimiento de item. El objeto de descubrimiento de item enlaza un item descubierto con una plantilla de item de la que se descubrió. Tiene las siguientes propiedades: `parent_itemid` - `(string)` ID de la plantilla de item de la que se ha creado el item; `key_` - `(string)` clave de la plantilla de item; `status` - `(int)` estado de descubrimiento del item: 0 - *(por defecto)* el item está descubierto, 1 - el item ya no está descubierto; `ts_delete` - `(timestamp)` hora en la que se eliminará un item que ya no se descubre; `ts_disable` - `(timestamp)` hora en la que se deshabilitará un item que ya no se descubre; `disable_source` - `(int)` indicador de si el item fue deshabilitado por una regla LLD o manualmente: 0 - *(por defecto)* deshabilitado automáticamente, 1 - deshabilitado por una regla LLD.
selectDiscoveryRule	query	Devuelve una propiedad `discoveryRule` con la regla LLD que creó el item.
selectInheritedTags	query	Devuelve una propiedad `inheritedTags` con las tags que están en template/host/templates enlazadas.
selectPreprocessing	query	Devuelve una propiedad `preprocessing` con las opciones de preprocesamiento del item.
selectTags	query	Devuelve las tags del item en la propiedad `tags`.
selectValueMap	query	Devuelve una propiedad `valuemap` con el mapa de valores del item.

Parámetro	Tipo	Descripción
filter	object	Devuelve solo aquellos resultados que coincidan exactamente con el filtro dado. Acepta un objeto, donde las claves son nombres de propiedades y los valores son un único valor o un array de valores con los que comparar. No admite propiedades del tipo de datos `text` data type. Admite propiedades adicionales: `host` - nombre técnico del host al que pertenece el item.
limitSelects	integer	Limita el número de registros devueltos por subconsultas. Se aplica a las siguientes subconsultas: `selectGraphs` - los resultados se ordenarán por `name`; `selectTriggers` - los resultados se ordenarán por `description`.
sortfield	string/array	Ordena el resultado por las propiedades dadas. Valores posibles: `itemid`, `name`, `key_`, `delay`, `history`, `trends`, `type`, `status`.
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
editable	boolean	
excludeSearch	boolean	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	
selectItemDiscovery	query	Devuelve una propiedad `itemDiscovery` con el objeto de descubrimiento de item. El objeto de descubrimiento de item enlaza el item con una plantilla de item de la que se creó. Esta consulta está **obsoleta**, por favor use `selectDiscoveryData` en su lugar.

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Buscar items por clave

Recuperar todos los items utilizados en triggers para un ID de host específico que tengan la palabra ”system.cpu” en la clave del item y ordenar los resultados por nombre.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "item.get",
    "params": {
        "output": "extend",
        "hostids": "10084",
        "with_triggers": true,
        "search": {
            "key_": "system.cpu"
        },
        "sortfield": "name"
```

```
    },
    "id": 1
}
```

Respuesta:


```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "itemid": "42269",
            "type": "18",
            "snmp_oid": "",
            "hostid": "10084",
            "name": "CPU utilization",
            "key_": "system.cpu.util",
            "delay": "0",
            "history": "7d",
            "trends": "365d",
            "status": "0",
            "value_type": "0",
            "trapper_hosts": "",
            "units": "%",
            "formula": "",
            "logtimefmt": "",
            "templateid": "42267",
            "valuemapid": "0",
            "params": "",
            "ipmi_sensor": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "flags": "0",
            "interfaceid": "0",
            "description": "CPU utilization in %.",
            "inventory_link": "0",
            "lifetime": "7d",
            "evaltype": "0",
            "jmx_endpoint": "",
            "master_itemid": "42264",
            "timeout": "",
            "url": "",
            "query_fields": [],
            "posts": "",
            "status_codes": "200",
            "follow_redirects": "1",
            "post_type": "0",
            "http_proxy": "",
            "headers": [],
            "retrieve_mode": "0",
            "request_method": "0",
            "output_format": "0",
            "ssl_cert_file": "",
            "ssl_key_file": "",
            "ssl_key_password": "",
            "verify_peer": "0",
            "verify_host": "0",
            "allow_traps": "0",
            "uuid": "",
            "lifetime_type": "0",
            "enabled_lifetime_type": "2",
```

```
            "enabled_lifetime": "0",
            "state": "0",
            "error": "",
            "name_resolved": "CPU utilization",
            "parameters": [],
            "lastclock": "0",
            "lastns": "0",
            "lastvalue": "0",
            "prevvalue": "0"
        },
        {
            "itemid": "42259",
            "type": "0",
            "snmp_oid": "",
            "hostid": "10084",
            "name": "Load average (15m avg)",
            "key_": "system.cpu.load[all,avg15]",
            "delay": "1m",
            "history": "7d",
            "trends": "365d",
            "status": "0",
            "value_type": "0",
            "trapper_hosts": "",
            "units": "",
            "formula": "",
            "logtimefmt": "",
            "templateid": "42219",
            "valuemapid": "0",
            "params": "",
            "ipmi_sensor": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "flags": "0",
            "interfaceid": "1",
            "description": "",
            "inventory_link": "0",
            "lifetime": "7d",
            "evaltype": "0",
            "jmx_endpoint": "",
            "master_itemid": "0",
            "timeout": "",
            "url": "",
            "query_fields": [],
            "posts": "",
            "status_codes": "200",
            "follow_redirects": "1",
            "post_type": "0",
            "http_proxy": "",
            "headers": [],
            "retrieve_mode": "0",
            "request_method": "0",
            "output_format": "0",
            "ssl_cert_file": "",
            "ssl_key_file": "",
            "ssl_key_password": "",
            "verify_peer": "0",
            "verify_host": "0",
            "allow_traps": "0",
            "uuid": "",
```

```
            "lifetime_type": "0",
            "enabled_lifetime_type": "2",
            "enabled_lifetime": "0",
            "state": "0",
            "error": "",
            "name_resolved": "Load average (15m avg)",
            "parameters": [],
            "lastclock": "0",
            "lastns": "0",
            "lastvalue": "0",
            "prevvalue": "0"
        },
        {
            "itemid": "42249",
            "type": "0",
            "snmp_oid": "",
            "hostid": "10084",
            "name": "Load average (1m avg)",
            "key_": "system.cpu.load[all,avg1]",
            "delay": "1m",
            "history": "7d",
            "trends": "365d",
            "status": "0",
            "value_type": "0",
            "trapper_hosts": "",
            "units": "",
            "formula": "",
            "logtimefmt": "",
            "templateid": "42209",
            "valuemapid": "0",
            "params": "",
            "ipmi_sensor": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "flags": "0",
            "interfaceid": "1",
            "description": "",
            "inventory_link": "0",
            "lifetime": "7d",
            "evaltype": "0",
            "jmx_endpoint": "",
            "master_itemid": "0",
            "timeout": "",
            "url": "",
            "query_fields": [],
            "posts": "",
            "status_codes": "200",
            "follow_redirects": "1",
            "post_type": "0",
            "http_proxy": "",
            "headers": [],
            "retrieve_mode": "0",
            "request_method": "0",
            "output_format": "0",
            "ssl_cert_file": "",
            "ssl_key_file": "",
            "ssl_key_password": "",
            "verify_peer": "0",
            "verify_host": "0",
```

```
            "allow_traps": "0",
            "uuid": "",
            "lifetime_type": "0",
            "enabled_lifetime_type": "2",
            "enabled_lifetime": "0",
            "state": "0",
            "error": "",
            "name_resolved": "Load average (1m avg)",
            "parameters": [],
            "lastclock": "0",
            "lastns": "0",
            "lastvalue": "0",
            "prevvalue": "0"
        },
        {
            "itemid": "42257",
            "type": "0",
            "snmp_oid": "",
            "hostid": "10084",
            "name": "Load average (5m avg)",
            "key_": "system.cpu.load[all,avg5]",
            "delay": "1m",
            "history": "7d",
            "trends": "365d",
            "status": "0",
            "value_type": "0",
            "trapper_hosts": "",
            "units": "",
            "formula": "",
            "logtimefmt": "",
            "templateid": "42217",
            "valuemapid": "0",
            "params": "",
            "ipmi_sensor": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "flags": "0",
            "interfaceid": "1",
            "description": "",
            "inventory_link": "0",
            "lifetime": "7d",
            "evaltype": "0",
            "jmx_endpoint": "",
            "master_itemid": "0",
            "timeout": "",
            "url": "",
            "query_fields": [],
            "posts": "",
            "status_codes": "200",
            "follow_redirects": "1",
            "post_type": "0",
            "http_proxy": "",
            "headers": [],
            "retrieve_mode": "0",
            "request_method": "0",
            "output_format": "0",
            "ssl_cert_file": "",
            "ssl_key_file": "",
            "ssl_key_password": "",
```

```
            "verify_peer": "0",
            "verify_host": "0",
            "allow_traps": "0",
            "uuid": "",
            "lifetime_type": "0",
            "enabled_lifetime_type": "2",
            "enabled_lifetime": "0",
            "state": "0",
            "error": "",
            "name_resolved": "Load average (5m avg)",
            "parameters": [],
            "lastclock": "0",
            "lastns": "0",
            "lastvalue": "0",
            "prevvalue": "0"
        },
        {
            "itemid": "42260",
            "type": "0",
            "snmp_oid": "",
            "hostid": "10084",
            "name": "Number of CPUs",
            "key_": "system.cpu.num",
            "delay": "1m",
            "history": "7d",
            "trends": "365d",
            "status": "0",
            "value_type": "3",
            "trapper_hosts": "",
            "units": "",
            "formula": "",
            "logtimefmt": "",
            "templateid": "42220",
            "valuemapid": "0",
            "params": "",
            "ipmi_sensor": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "flags": "0",
            "interfaceid": "1",
            "description": "",
            "inventory_link": "0",
            "lifetime": "7d",
            "evaltype": "0",
            "jmx_endpoint": "",
            "master_itemid": "0",
            "timeout": "",
            "url": "",
            "query_fields": [],
            "posts": "",
            "status_codes": "200",
            "follow_redirects": "1",
            "post_type": "0",
            "http_proxy": "",
            "headers": [],
            "retrieve_mode": "0",
            "request_method": "0",
            "output_format": "0",
            "ssl_cert_file": "",
```

```
            "ssl_key_file": "",
            "ssl_key_password": "",
            "verify_peer": "0",
            "verify_host": "0",
            "allow_traps": "0",
            "uuid": "",
            "lifetime_type": "0",
            "enabled_lifetime_type": "2",
            "enabled_lifetime": "0",
            "state": "0",
            "error": "",
            "name_resolved": "Number of CPUs",
            "parameters": [],
            "lastclock": "0",
            "lastns": "0",
            "lastvalue": "0",
            "prevvalue": "0"
        }
    ],
    "id": 1
}
```


Buscar items dependientes por clave

Recuperar todos los items dependientes del host con ID ”10116” que tengan la palabra ”apache” en la clave.

Solicitud:


```
{
    "jsonrpc": "2.0",
    "method": "item.get",
    "params": {
        "output": "extend",
        "hostids": "10116",
        "search": {
            "key_": "apache"
        },
        "filter": {
            "type": 18
        }
    },
    "id": 1
}
```


Respuesta:


```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "itemid": "25550",
            "type": "18",
            "snmp_oid": "",
            "hostid": "10116",
            "name": "Days",
            "key_": "apache.status.uptime.days",
            "delay": "0",
            "history": "90d",
            "trends": "365d",
            "status": "0",
            "value_type": "3",
            "trapper_hosts": "",
            "units": "",
            "formula": "",
            "logtimefmt": "",
```

```
            "templateid": "0",
            "valuemapid": "0",
            "params": "",
            "ipmi_sensor": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "flags": "0",
            "interfaceid": "0",
            "description": "",
            "inventory_link": "0",
            "lifetime": "7d",
            "evaltype": "0",
            "jmx_endpoint": "",
            "master_itemid": "25545",
            "timeout": "",
            "url": "",
            "query_fields": [],
            "posts": "",
            "status_codes": "200",
            "follow_redirects": "1",
            "post_type": "0",
            "http_proxy": "",
            "headers": [],
            "retrieve_mode": "0",
            "request_method": "0",
            "output_format": "0",
            "ssl_cert_file": "",
            "ssl_key_file": "",
            "ssl_key_password": "",
            "verify_peer": "0",
            "verify_host": "0",
            "allow_traps": "0",
            "uuid": "",
            "lifetime_type": "0",
            "enabled_lifetime_type": "2",
            "enabled_lifetime": "0",
            "state": "0",
            "error": "",
            "name_resolved": "Days",
            "parameters": [],
            "lastclock": "0",
            "lastns": "0",
            "lastvalue": "0",
            "prevvalue": "0"
        },
        {
            "itemid": "25555",
            "type": "18",
            "snmp_oid": "",
            "hostid": "10116",
            "name": "Hours",
            "key_": "apache.status.uptime.hours",
            "delay": "0",
            "history": "90d",
            "trends": "365d",
            "status": "0",
            "value_type": "3",
            "trapper_hosts": "",
            "units": "",
```

```
            "formula": "",
            "logtimefmt": "",
            "templateid": "0",
            "valuemapid": "0",
            "params": "",
            "ipmi_sensor": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "flags": "0",
            "interfaceid": "0",
            "description": "",
            "inventory_link": "0",
            "lifetime": "7d",
            "evaltype": "0",
            "jmx_endpoint": "",
            "master_itemid": "25545",
            "timeout": "",
            "url": "",
            "query_fields": [],
            "posts": "",
            "status_codes": "200",
            "follow_redirects": "1",
            "post_type": "0",
            "http_proxy": "",
            "headers": [],
            "retrieve_mode": "0",
            "request_method": "0",
            "output_format": "0",
            "ssl_cert_file": "",
            "ssl_key_file": "",
            "ssl_key_password": "",
            "verify_peer": "0",
            "verify_host": "0",
            "allow_traps": "0",
            "uuid": "",
            "lifetime_type": "0",
            "enabled_lifetime_type": "2",
            "enabled_lifetime": "0",
            "state": "0",
            "error": "",
            "name_resolved": "Hours",
            "parameters": [],
            "lastclock": "0",
            "lastns": "0",
            "lastvalue": "0",
            "prevvalue": "0"
        }
    ],
    "id": 1
}
```


Buscar item HTTP agent

Buscar item HTTP agent con tipo de cuerpo post XML para un ID de host específico.

Solicitud:


```
{
    "jsonrpc": "2.0",
    "method": "item.get",
    "params": {
```

```
        "hostids": "10255",
        "filter": {
            "type": 19,
            "post_type": 3
        }
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "itemid": "28252",
            "type": "19",
            "snmp_oid": "",
            "hostid": "10255",
            "name": "template item",
            "key_": "ti",
            "delay": "30s",
            "history": "90d",
            "trends": "365d",
            "status": "0",
            "value_type": "3",
            "trapper_hosts": "",
            "units": "",
            "formula": "",
            "logtimefmt": "",
            "templateid": "0",
            "valuemapid": "0",
            "params": "",
            "ipmi_sensor": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "flags": "0",
            "interfaceid": "0",
            "description": "",
            "inventory_link": "0",
            "lifetime": "7d",
            "evaltype": "0",
            "jmx_endpoint": "",
            "master_itemid": "0",
            "timeout": "",
            "url": "localhost",
            "query_fields": [
                {
                    "name": "mode",
                    "value": "xml"
                }
            ],
            "posts": "<body>\r\n<![CDATA[{$MACRO}<foo></bar>]]>\r\n</body>",
            "status_codes": "200",
            "follow_redirects": "0",
            "post_type": "3",
            "http_proxy": "",
            "headers": [],
            "retrieve_mode": "1",
            "request_method": "3",
```

```
            "output_format": "0",
            "ssl_cert_file": "",
            "ssl_key_file": "",
            "ssl_key_password": "",
            "verify_peer": "0",
            "verify_host": "0",
            "allow_traps": "0",
            "uuid": "",
            "lifetime_type": "0",
            "enabled_lifetime_type": "2",
            "enabled_lifetime": "0",
            "state": "0",
            "error": "",
            "name_resolved": "template item",
            "parameters": [],
            "lastclock": "0",
            "lastns": "0",
            "lastvalue": "",
            "prevvalue": ""
        }
    ],
    "id": 1
}
```

Recuperar métricas con reglas de preprocesamiento

Recuperar todas las métricas y sus reglas de preprocesamiento para un ID de equipo específico.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "item.get",
    "params": {
        "output": ["itemid", "name", "key_"],
        "selectPreprocessing": "extend",
        "hostids": "10254"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemid": "23865",
        "name": "HTTP agent example JSON",
        "key_": "json",
        "preprocessing": [
            {
                "type": "12",
                "params": "$.random",
                "error_handler": "1",
                "error_handler_params": ""
            }
        ]
    },
    "id": 1
}
```

Ver también

- Regla de descubrimiento
- Gráfico
- Host

- Interfaz de host
- Trigger

Fuente

CItem::get() en *ui/include/classes/api/services/CItem.php*.

item.update

Descripción

`object item.update(object/array items)`

Este método le permite actualizar items existentes.

Note:
Los web items no se pueden actualizar a través de la API de Zabbix.

Note:
Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object/array)` Propiedades del item que se van a actualizar.

La propiedad `itemid` debe estar definida para cada item, todas las demás propiedades son opcionales. Sólo se actualizarán las propiedades pasadas, todas las demás permanecerán sin cambios.

Además de las propiedades estándar del item, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
preprocessing	array	Opciones de preprocesamiento del item para reemplazar las opciones de preprocesamiento actuales. Comportamiento del parámetro: - *sólo lectura* para objetos heredados u objetos descubiertos
tags	array	Etiquetas del item. Comportamiento del parámetro: - *sólo lectura* para objetos descubiertos

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las métricas actualizadas bajo la propiedad `itemids`.

Ejemplos

Habilitar una métrica

Habilite una métrica, es decir, establezca su estado en "0".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "item.update",
    "params": {
        "itemid": "10092",
        "status": 0
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "10092"
        ]
    },
    "id": 1
}
```

Actualizar métrica dependiente

Actualiza el nombre de la métrica dependiente y el ID de la métrica principal. Solo se permiten dependencias en el mismo equipo, por lo tanto, la métrica principal y la dependiente deben tener el mismo hostid.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "item.update",
    "params": {
        "name": "Nombre actualizado de la métrica dependiente",
        "master_itemid": "25562",
        "itemid": "189019"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "189019"
        ]
    },
    "id": 1
}
```

Actualizar métrica de agente HTTP

Habilitar la captura de valores de la métrica.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "item.update",
    "params": {
        "itemid": "23856",
        "allow_traps": 1
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "23856"
        ]
    },
    "id": 1
}
```

Actualizar una métrica con preprocesamiento

Actualizar una métrica con la regla de preprocesamiento "En rango".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "item.update",
    "params": {
        "itemid": "23856",
        "preprocessing": [
            {
                "type": 13,
                "params": "\n100",
                "error_handler": 1,
                "error_handler_params": ""
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "23856"
        ]
    },
    "id": 1
}
```

Actualizar un item de script

Actualice un item de script con un script diferente y elimine los parámetros innecesarios que fueron utilizados por el script anterior.

Request:

```
{
    "jsonrpc": "2.0",
    "method": "item.update",
    "params": {
        "itemid": "23865",
        "parameters": [],
        "params": "Zabbix.log(3, 'Log test');\nreturn 1;"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "23865"
        ]
    },
    "id": 1
}
```

Fuente

CItem::update() en *ui/include/classes/api/services/CItem.php*.

Item de gráfico

Esta clase está diseñada para trabajar con items de gráficos.

Referencias de objetos:

- Item de gráfico

Métodos disponibles:

- graphitem.get - recuperar items de gráficos

Objeto de métrica de gráfico

Los siguientes objetos están directamente relacionados con la API `graphitem`.

Item de gráfico

> **Note:**
> Los items de gráfico solo pueden modificarse a través de la API `graph`.

El objeto item de gráfico tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
gitemid	ID	ID del item de gráfico. Comportamiento de la propiedad: - *solo lectura*
color	string	Color de dibujo del item de gráfico como un código de color hexadecimal. Comportamiento de la propiedad: - *requerido* para operaciones de creación
itemid	ID	ID del item. Comportamiento de la propiedad: - *requerido* para operaciones de creación
calc_fnc	integer	Valor del item que se mostrará. Valores posibles: 1 - valor mínimo; 2 - *(por defecto)* valor promedio; 4 - valor máximo; 7 - todos los valores; 9 - último valor, utilizado solo por gráficos de tipo pastel y explotados.
drawtype	integer	Estilo de dibujo del item de gráfico. Valores posibles: 0 - *(por defecto)* línea; 1 - región rellena; 2 - línea en negrita; 3 - punto; 4 - línea discontinua; 5 - línea de degradado.
graphid	ID	ID del gráfico al que pertenece el item de gráfico.
sortorder	integer	Posición del item en el gráfico. Por defecto: comienza en ”0” y aumenta en uno con cada entrada.
type	integer	Tipo de item de gráfico. Valores posibles: 0 - *(por defecto)* simple; 2 - suma de gráfico, utilizada solo por gráficos de tipo pastel y explotados.

Propiedad	Tipo	Descripción
yaxisside	integer	Lado del gráfico donde se dibujará la escala Y del item de gráfico. Valores posibles: 0 - *(por defecto)* lado izquierdo; 1 - lado derecho.

graphitem.get

Descripción

`integer/array graphitem.get(object parameters)`

El método permite recuperar elementos de gráfico según los parámetros proporcionados.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
graphids	ID/array	Devolver solo los items de gráfico que pertenezcan a los gráficos indicados.
itemids	ID/array	Devolver solo los items de gráfico con los IDs de item indicados.
type	integer	Devolver solo los items de gráfico con el tipo indicado. Consulte la página del objeto de item de gráfico para ver una lista de los tipos de item de gráfico soportados.
selectGraphs	query	Devolver una propiedad `graphs` con un array de gráficos a los que pertenece el item.
sortfield	string/array	Ordenar el resultado por las propiedades indicadas. Valores posibles: `gitemid`.
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
editable	boolean	
limit	integer	
output	query	
preservekeys	boolean	
sortorder	string/array	

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperar métricas de un gráfico

Recupera todas las métricas utilizadas en un gráfico con información adicional sobre la métrica y el equipo.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "graphitem.get",
    "params": {
        "output": "extend",
```

```
        "graphids": "387"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "gitemid": "1242",
            "graphid": "387",
            "itemid": "22665",
            "drawtype": "1",
            "sortorder": "1",
            "color": "FF5555",
            "yaxisside": "0",
            "calc_fnc": "2",
            "type": "0"
        },
        {
            "gitemid": "1243",
            "graphid": "387",
            "itemid": "22668",
            "drawtype": "1",
            "sortorder": "2",
            "color": "55FF55",
            "yaxisside": "0",
            "calc_fnc": "2",
            "type": "0"
        },
        {
            "gitemid": "1244",
            "graphid": "387",
            "itemid": "22671",
            "drawtype": "1",
            "sortorder": "3",
            "color": "009999",
            "yaxisside": "0",
            "calc_fnc": "2",
            "type": "0"
        }
    ],
    "id": 1
}
```

Véase también

- Gráfico

Fuente

CGraphItem::get() en *ui/include/classes/api/services/CGraphItem.php*.

Macro de usuario

Esta clase está diseñada para trabajar con macros de usuario a nivel de host y globales.

Referencias de objetos:

- Macro global
- Macro de host
    - Configuración de macro

Métodos disponibles:

- usermacro.create - crear nuevas macros de host
- usermacro.createglobal - crear nuevas macros globales
- usermacro.delete - eliminar macros de host
- usermacro.deleteglobal - eliminar macros globales
- usermacro.get - recuperar macros de host y globales
- usermacro.update - actualizar macros de host
- usermacro.updateglobal - actualizar macros globales

Objeto macro de usuario

Los siguientes objetos están directamente relacionados con la API `usermacro`.

Macro global

El objeto macro global tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
globalmacroid	ID	ID de la macro global. Comportamiento de la propiedad: - *solo lectura* - *requerido* para operaciones de actualización
description	string	Descripción de la macro.
macro	string	Cadena de la macro. Comportamiento de la propiedad: - *requerido* para operaciones de creación
type	integer	Tipo de macro. Valores posibles: 0 - *(por defecto)* Macro de texto; 1 - Macro secreta; 2 - Secreto de Vault.
value	string	Valor de la macro. Comportamiento de la propiedad: - *solo escritura* si `type` está establecido en ”Macro secreta” - *requerido* para operaciones de creación

Macro de host

El objeto macro de host define una macro disponible en un host, prototipo de host o template. Tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
hostmacroid	ID	ID de la macro. Comportamiento de la propiedad: - *solo lectura* - *requerido* para operaciones de actualización
automatic	integer	Define si la macro está controlada por la regla de descubrimiento. Valores posibles: 0 - *(por defecto)* La macro es gestionada por el usuario; 1 - La macro es gestionada por la regla de descubrimiento. No se permite al usuario crear una macro automática. Para actualizar una macro automática, debe ser convertida a manual.
config	object/array	Configuración de macro responsable de cómo se muestra la macro en el Asistente de host.
description	string	Descripción de la macro.

Propiedad	Tipo	Descripción
hostid	ID	ID del host, prototipo de host o template al que pertenece la macro. Comportamiento de la propiedad: - *constante* - *requerido* para operaciones de creación
macro	string	Cadena de la macro. Comportamiento de la propiedad: - *requerido* para operaciones de creación
type	integer	Tipo de macro. Valores posibles: 0 - *(por defecto)* Macro de texto; 1 - Macro secreta; 2 - Secreto de Vault.
value	string	Valor de la macro. Comportamiento de la propiedad: - *solo escritura* si `type` está establecido en ”Macro secreta” - *requerido* para operaciones de creación

Configuración de macro

El objeto de configuración de macro define cómo se muestra una macro en el Asistente de host.

Propiedad	Tipo	Descripción
type	integer	Tipo de campo de entrada de macro. Valores posibles: 0 - La macro no se utiliza en el Asistente de host; 1 - Cuadro de texto; 2 - Lista; 3 - Casilla de verificación. Comportamiento de la propiedad: - *requerido*
label	string	Etiqueta para el campo de entrada de macro. Comportamiento de la propiedad: - *requerido* si `type` está establecido en ”Cuadro de texto”, ”Lista” o ”Casilla de verificación”
description	text	Texto de ayuda que se muestra junto al campo de entrada de macro. Soporta formato Markdown.
priority	integer	Posición del campo de entrada de macro en la lista de macros. Si `priority` no está establecido, la macro se añade al final de la lista de macros no agrupadas.
required	integer	Marca la macro como obligatoria. Valores posibles: 0 - No obligatoria; 1 - Obligatoria. Comportamiento de la propiedad: - *soportado* si `type` está establecido en ”Cuadro de texto” o ”Lista”
regex	string	Expresión regular para validar la entrada del usuario en un campo de cuadro de texto. Comportamiento de la propiedad: - *soportado* si `type` está establecido en ”Cuadro de texto”

Propiedad	Tipo	Descripción
section_name	string	Etiqueta de la sección plegable donde se agrupa la macro. Si `section_name` no está establecido, la macro no está agrupada. Las macros no agrupadas se muestran primero; las agrupadas se muestran debajo, ordenadas por su `priority` dentro de cada sección plegable.
options	text	Cadena JSON que define los elementos de la lista o los valores de la casilla de verificación. Para listas: un array de objetos con las propiedades `value` y `text`. Ejemplo: `[{"value": "http", "text": "HTTP"}, {"value": "https", "text": "HTTPS"}]` Para casillas de verificación: un objeto con las propiedades `checked` y `unchecked`. Ejemplo: `{"checked": true, "unchecked": false}` Comportamiento de la propiedad: - *requerido* si `type` está establecido en ”Lista” o ”Casilla de verificación”

usermacro.create

Descripción

`object usermacro.create(object/array hostMacros)`

Este método permite crear nuevos macros de host.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object/array)` Macros de host a crear.

El método acepta macros de host con las propiedades estándar de las macros de host.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las macros de equipo creadas bajo la propiedad `hostmacroids`. El orden de los IDs devueltos coincide con el orden de las macros de equipo enviadas.

Ejemplos

Creación de un macro de host

Cree un macro de host `{$SNMP_COMMUNITY}` con el valor ”public” en el host ”10198”.

Request:

```
{
    "jsonrpc": "2.0",
    "method": "usermacro.create",
    "params": {
        "hostid": "10198",
        "macro": "{$SNMP_COMMUNITY}",
        "value": "public"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
```

```
    "result": {
        "hostmacroids": [
            "11"
        ]
    },
    "id": 1
}
```

Fuente

CUserMacro::create() en *ui/include/classes/api/services/CUserMacro.php*.

usermacro.createglobal

Descripción

`object usermacro.createglobal(object/array globalMacros)`

Este método permite crear nuevas macros globales.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object/array)` Macros globales a crear.

El método acepta macros globales con las propiedades estándar de macro global.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las macros globales creadas bajo la propiedad `globalmacroids`. El orden de los IDs devueltos coincide con el orden de las macros globales proporcionadas.

Ejemplos

Creación de una macro global

Cree una macro global `{$SNMP_COMMUNITY}` con el valor "public".

Request:

```
{
    "jsonrpc": "2.0",
    "method": "usermacro.createglobal",
    "params":  {
        "macro": "{$SNMP_COMMUNITY}",
        "value": "public"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "globalmacroids": [
            "6"
        ]
    },
    "id": 1
}
```

Fuente

CUserMacro::createGlobal() en *ui/include/classes/api/services/CUserMacro.php*.

usermacro.delete

Descripción

`object usermacro.delete(array hostMacroIds)`

Este método le permite eliminar macros de host.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración del rol de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(array)` IDs de las macros de equipo a eliminar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las macros de equipo eliminadas bajo la propiedad `hostmacroids`.

Ejemplos

Eliminando múltiples macros de equipo

Eliminar dos macros de equipo.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "usermacro.delete",
    "params": [
        "32",
        "11"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostmacroids": [
            "32",
            "11"
        ]
    },
    "id": 1
}
```

Fuente

CUserMacro::delete() en *ui/include/classes/api/services/CUserMacro.php*.

usermacro.deleteglobal

Descripción

`object usermacro.deleteglobal(array globalMacroIds)`

Este método permite eliminar macros globales.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(array)` IDs de las macros globales a eliminar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las macros globales eliminadas bajo la propiedad `globalmacroids`.

Ejemplos

Eliminando múltiples macros globales

Eliminar dos macros globales.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "usermacro.deleteglobal",
    "params": [
        "32",
        "11"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "globalmacroids": [
            "32",
            "11"
        ]
    },
    "id": 1
}
```

Fuente

CUserMacro::deleteGlobal() en *ui/include/classes/api/services/CUserMacro.php*.

usermacro.get

Descripción

`integer/array usermacro.get(object parameters)`

El método permite recuperar macros de host y globales según los parámetros proporcionados.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método soporta los siguientes parámetros.

Parámetro	Tipo	Descripción
globalmacro	flag	Devolver macros globales en lugar de macros de host.
globalmacroids	ID/array	Devolver solo las macros globales con los IDs dados.
groupids	ID/array	Devolver solo las macros de host que pertenecen a hosts o templates de los grupos de host o grupos de template dados.
hostids	ID/array	Devolver solo las macros que pertenecen a los hosts o templates dados.
hostmacroids	ID/array	Devolver solo las macros de host con los IDs dados.
inherited	boolean	Si se establece en `true`, devolver solo las macros de usuario prototipo de host heredadas de un template.

Parámetro	Tipo	Descripción
selectHostGroups	query	Devolver los grupos de host a los que pertenece la macro de host en la propiedad `hostgroups`.
selectHosts	query	Se utiliza solo al recuperar macros de host. Devolver los hosts a los que pertenece la macro de host en la propiedad `hosts`.
selectTemplateGroups	query	Se utiliza solo al recuperar macros de host. Devolver los grupos de template a los que pertenece la macro de template en la propiedad `templategroups`.
selectTemplates	query	Se utiliza solo al recuperar macros de template. Devolver los templates a los que pertenece la macro de host en la propiedad `templates`.
sortfield	string/array	Se utiliza solo al recuperar macros de host. Ordenar el resultado por las propiedades dadas. Valores posibles: `macro`.
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperar macros de equipo para un equipo

Recuperar todas las macros de equipo definidas para el equipo "10198".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "usermacro.get",
    "params": {
        "output": "extend",
        "hostids": "10198"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "hostmacroid": "9",
            "hostid": "10198",
```

```
            "macro": "{$INTERFACE}",
            "value": "eth0",
            "description": "",
            "type": "0",
            "automatic": "0"
        },
        {
            "hostmacroid": "11",
            "hostid": "10198",
            "macro": "{$SNMP_COMMUNITY}",
            "value": "public",
            "description": "",
            "type": "0",
            "automatic": "0"
        }
    ],
    "id": 1
}
```

Recuperar macros de equipo para una plantilla

Recuperar las macros de equipo definidas para la plantilla ”10265” que contienen ”STATUS” en el nombre de la macro.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "usermacro.get",
    "params": {
        "output": "extend",
        "hostids": "10265",
        "search": {
            "macro": "STATUS"
        }
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "hostmacroid": "6709",
            "hostid": "10265",
            "macro": "{$APACHE.STATUS.HOST}",
            "value": "",
            "description": "El nombre de equipo o la dirección IP del equipo de la página de estado de Apa
            "type": "0",
            "automatic": "0",
            "config": {
                "type": "1",
                "priority": "1",
                "section_name": "",
                "label": "Equipo de la página de estado de Apache",
                "description": "Nombre de equipo o dirección IP del equipo de la página de estado de Apach
                "required": "1",
                "regex": "",
                "options": []
            }
        },
        {
            "hostmacroid": "814",
            "hostid": "10265",
```

```
            "macro": "{$APACHE.STATUS.PATH}",
            "value": "server-status?auto",
            "description": "La ruta URL de la página de estado de Apache.",
            "type": "0",
            "automatic": "0",
            "config": {
                "type": "1",
                "priority": "3",
                "section_name": "",
                "label": "Ruta de la página de estado de Apache",
                "description": "Ruta URL de la página de estado de Apache.",
                "required": "1",
                "regex": "",
                "options": []
            }
        },
        {
            "hostmacroid": "815",
            "hostid": "10265",
            "macro": "{$APACHE.STATUS.PORT}",
            "value": "80",
            "description": "El puerto de la página de estado de Apache.",
            "type": "0",
            "automatic": "0",
            "config": {
                "type": "1",
                "priority": "2",
                "section_name": "",
                "label": "Puerto de la página de estado de Apache",
                "description": "En el rango de 1 a 65535 inclusive.",
                "required": "1",
                "regex": "^-?([0-9]+|(([0-9]+)\\.([0-9]+)))$",
                "options": []
            }
        },
        {
            "hostmacroid": "816",
            "hostid": "10265",
            "macro": "{$APACHE.STATUS.SCHEME}",
            "value": "http",
            "description": "El esquema de la petición, que puede ser HTTP o HTTPS.",
            "type": "0",
            "automatic": "0",
            "config": {
                "type": "2",
                "priority": "4",
                "section_name": "",
                "label": "Esquema de la petición",
                "description": "Esquema de la petición, que puede ser HTTP o HTTPS.",
                "required": "0",
                "regex": "",
                "options": [
                    {
                        "value": "http",
                        "text": "HTTP"
                    },
                    {
                        "value": "https",
                        "text": "HTTPS"
                    }
                ]
            }
```

```
        }
    ],
    "id": 1
}
```

Recuperar macros globales

Recuperar todas las macros globales.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "usermacro.get",
    "params": {
        "output": "extend",
        "globalmacro": true
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "globalmacroid": "6",
            "macro": "{$SNMP_COMMUNITY}",
            "value": "public",
            "description": "",
            "type": "0"
        }
    ],
    "id": 1
}
```

Fuente

CUserMacro::get() en *ui/include/classes/api/services/CUserMacro.php*.

usermacro.update

Descripción

`object usermacro.update(object/array hostMacros)`

Este método permite actualizar macros de host existentes.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración del rol de usuario. Consulte User roles para obtener más información.

Parámetros

`(object/array)` Propiedades de macro de host que se van a actualizar.

La propiedad `hostmacroid` debe estar definida para cada macro de host, todas las demás propiedades son opcionales. Sólo se actualizarán las propiedades pasadas, todas las demás permanecerán sin cambios.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las macros de equipo actualizadas bajo la propiedad `hostmacroids`.

Ejemplos

Cambiar el valor de una macro de equipo

Cambie el valor de una macro de equipo a ”public”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "usermacro.update",
    "params": {
        "hostmacroid": "1",
        "value": "public"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostmacroids": [
            "1"
        ]
    },
    "id": 1
}
```

Cambiar el valor de la macro que fue creada por una regla de descubrimiento

Convertir la macro ”automática” creada por una regla de descubrimiento en ”manual” y cambiar su valor a ”nuevo-valor”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "usermacro.update",
    "params": {
        "hostmacroid": "1",
        "value": "nuevo-valor",
        "automatic": "0"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostmacroids": [
            "1"
        ]
    },
    "id": 1
}
```

Fuente

CUserMacro::update() en *ui/include/classes/api/services/CUserMacro.php*.

usermacro.updateglobal

Descripción

`object usermacro.updateglobal(object/array globalMacros)`

Este método permite actualizar macros globales existentes.

Note:
Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object/array)` Propiedades de macro global que se van a actualizar.

La propiedad `globalmacroid` debe estar definida para cada macro global, todas las demás propiedades son opcionales. Sólo se actualizarán las propiedades enviadas, todas las demás permanecerán sin cambios.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las macros globales actualizadas bajo la propiedad `globalmacroids`.

Ejemplos

Cambiar el valor de una macro global

Cambie el valor de una macro global a ”public”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "usermacro.updateglobal",
    "params": {
        "globalmacroid": "1",
        "value": "public"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "globalmacroids": [
            "1"
        ]
    },
    "id": 1
}
```

Fuente

CUserMacro::updateGlobal() en *ui/include/classes/api/services/CUserMacro.php*.

Mantenimiento

Esta clase está diseñada para trabajar con mantenimientos.

Referencias de objetos:

- Mantenimiento
- Período de tiempo
- Etiqueta de problema

Métodos disponibles:

- maintenance.create - crear nuevos mantenimientos
- maintenance.delete - eliminar mantenimientos
- maintenance.get - recuperar mantenimientos
- maintenance.update - actualizar mantenimientos

Objeto de mantenimiento

Los siguientes objetos están directamente relacionados con la API de `maintenance`.

Mantenimiento

El objeto maintenance tiene las siguientes propiedades.

Property	Type	Description
maintenanceid	ID	ID del mantenimiento. Comportamiento de la propiedad: - *solo lectura* - *obligatorio* para operaciones de actualización
name	string	Nombre del mantenimiento. Comportamiento de la propiedad: - *obligatorio* para operaciones de creación
active_since	timestamp	Hora en que el mantenimiento pasa a estar activo (inclusive). El valor proporcionado se redondeará hacia abajo al minuto. Comportamiento de la propiedad: - *obligatorio* para operaciones de creación
active_till	timestamp	Hora en que el mantenimiento deja de estar activo (exclusiva). El valor proporcionado se redondeará hacia abajo al minuto. Comportamiento de la propiedad: - *obligatorio* para operaciones de creación
description	string	Descripción del mantenimiento.
maintenance_type	integer	Tipo de mantenimiento. Valores posibles: 0 - *(predeterminado)* con recopilación de datos; 1 - sin recopilación de datos.
tags_evaltype	integer	Método de evaluación de etiquetas de problema. Valores posibles: 0 - *(predeterminado)* And/Or; 2 - Or.

Periodo de tiempo

El objeto periodo de tiempo se utiliza para definir los periodos en los que el mantenimiento debe entrar en vigor. Tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
period	integer	Duración del periodo de mantenimiento en segundos. El valor dado se redondeará a la baja a minutos. Valores posibles: 300 - 86399940. Por defecto: 3600.
timeperiod_type	integer	Tipo de periodo de tiempo. Valores posibles: 0 - *(por defecto)* una sola vez; 2 - diario; 3 - semanal; 4 - mensual.

Propiedad	Tipo	Descripción
start_date	timestamp	Fecha en la que el periodo de mantenimiento debe entrar en vigor. El valor dado se redondeará a la baja a minutos. Por defecto: fecha actual. Comportamiento de la propiedad: - *soportado* si `timeperiod_type` está establecido en ”una sola vez”
start_time	integer	Hora del día en la que comienza el mantenimiento en segundos. El valor dado se redondeará a la baja a minutos. Por defecto: 0. Comportamiento de la propiedad: - *soportado* si `timeperiod_type` está establecido en ”diario”, ”semanal” o ”mensual”
every	integer	Para periodos diarios y semanales, `every` define los intervalos de días o semanas en los que el mantenimiento debe entrar en vigor. Valor por defecto si `timeperiod_type` está establecido en ”diario” o ”semanal”: 1. Para periodos mensuales cuando se establece `day`, la propiedad `every` define el día del mes en el que el mantenimiento debe entrar en vigor. Valor por defecto si `timeperiod_type` está establecido en ”mensual” y se establece `day`: 1. Para periodos mensuales cuando se establece `dayofweek`, la propiedad `every` define la semana del mes en la que el mantenimiento debe entrar en vigor. Valores posibles si `timeperiod_type` está establecido en ”mensual” y se establece `dayofweek`: 1 - *(por defecto)* primera semana; 2 - segunda semana; 3 - tercera semana; 4 - cuarta semana; 5 - última semana. Comportamiento de la propiedad: - *soportado* si `timeperiod_type` está establecido en ”diario”, ”semanal” o ”mensual”
dayofweek	integer	Días de la semana en los que el mantenimiento debe entrar en vigor. Valores posibles de la máscara de bits: 1 - Lunes; 2 - Martes; 4 - Miércoles; 8 - Jueves; 16 - Viernes; 32 - Sábado; 64 - Domingo. Este es un campo de máscara de bits; cualquier suma de los valores posibles es aceptable (por ejemplo, 21 para lunes, miércoles y viernes). Comportamiento de la propiedad: - *requerido* si `timeperiod_type` está establecido en ”semanal” o si `timeperiod_type` está establecido en ”mensual” y `day` no está establecido
day	integer	Día del mes en el que el mantenimiento debe entrar en vigor. Comportamiento de la propiedad: - *requerido* si `timeperiod_type` está establecido en ”mensual” y `dayofweek` no está establecido

Propiedad	Tipo	Descripción
month	integer	Meses en los que el mantenimiento debe entrar en vigor. Valores posibles de la máscara de bits: 1 - Enero; 2 - Febrero; 4 - Marzo; 8 - Abril; 16 - Mayo; 32 - Junio; 64 - Julio; 128 - Agosto; 256 - Septiembre; 512 - Octubre; 1024 - Noviembre; 2048 - Diciembre. Este es un campo de máscara de bits; cualquier suma de los valores posibles es aceptable (por ejemplo, 585 para enero, abril, julio y octubre). Comportamiento de la propiedad: - *requerido* si `timeperiod_type` está establecido en ”mensual”

Nombre del evento

El objeto de nombre del evento se utiliza para definir qué problemas deben suprimirse cuando el mantenimiento entre en vigor. Los nombres de eventos solo se pueden especificar si `maintenance_type` del objeto Maintenance está establecido en ”with data collection”.
Tiene las siguientes propiedades.

Property	Type	Description
operator	integer	Operador de la condición. Valores posibles: 2 - *(predeterminado)* Contiene; 3 - No contiene.
value	string	Valor del nombre del evento. Comportamiento de la propiedad: - *obligatorio*

Etiqueta del problema

El objeto de etiqueta del problema se utiliza para definir qué problemas deben suprimirse cuando entre en vigor el mantenimiento. Las etiquetas solo se pueden especificar si `maintenance_type` del objeto de mantenimiento está establecido en ”con recopilación de datos”. Tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
tag	string	Nombre de la etiqueta del problema. Comportamiento de la propiedad: - *obligatorio*
operator	integer	Operador de la condición. Valores posibles: 0 - Igual a; 1 - No es igual a; 2 - *(predeterminado)* Contiene; 3 - No contiene.

Propiedad	Tipo	Descripción
value	string	Valor de la etiqueta del problema.

maintenance.create

Descripción

`object maintenance.create(object/array maintenances)`

Este método le permite crear nuevos mantenimientos.

Note:
Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object/array)` Mantenimientos que se crearán.

Además de las propiedades estándar de mantenimiento, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
groups	object/array	Grupos de hosts que se incluirán en el mantenimiento. Los grupos de hosts deben tener definida únicamente la propiedad `groupid`. Comportamiento del parámetro: - *obligatorio* si no se establece `hosts`
hosts	object/array	Hosts que se incluirán en el mantenimiento. Los hosts deben tener definida únicamente la propiedad `hostid`. Comportamiento del parámetro: - *obligatorio* si no se establece `groups`
triggers	object/array	Triggers que se incluirán en el mantenimiento. Los triggers deben tener definida únicamente la propiedad `triggerid`. Comportamiento del parámetro: - *obligatorio* si no se establecen `groups` ni `hosts` - *compatible* si maintenance_type=0 (con recopilación de datos)
event_names	object/array	Nombres de eventos para definir los problemas que se incluirán en el mantenimiento. Comportamiento del parámetro: - *compatible* si maintenance_type=0 (con recopilación de datos)
tags	object/array	Etiquetas de problemas para definir los problemas que se incluirán en el mantenimiento. Comportamiento del parámetro: - *compatible* si `maintenance_type` del objeto Maintenance está establecido en "con recopilación de datos"
timeperiods	object/array	Períodos de tiempo del mantenimiento. Comportamiento del parámetro: - *obligatorio*

Valores de retorno

(`object`) Devuelve un objeto que contiene los IDs de los mantenimientos creados bajo la propiedad `maintenanceids`. El orden de los IDs devueltos coincide con el orden de los mantenimientos enviados.

Ejemplos

Creación de un mantenimiento

Cree un mantenimiento con recopilación de datos para el grupo de hosts con ID "20" y con las etiquetas de problema **service:mysqld** y **error**.
Debe estar activo desde el 17.03.2026 hasta el 17.03.2027, aplicarse cada domingo a las 18:00 y durar una hora.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "maintenance.create",
    "params": {
        "name": "Sunday maintenance",
        "active_since": 1773720240,
        "active_till": 1805256240,
        "tags_evaltype": 0,
        "groups": [
            {"groupid": "20"}
        ],
        "timeperiods": [
            {
                "period": 3600,
                "timeperiod_type": 3,
                "start_time": 64800,
                "every": 1,
                "dayofweek": 64
            }
        ],
        "tags": [
            {
                "tag": "service",
                "operator": "0",
                "value": "mysqld"
            },
            {
                "tag": "error",
                "operator": "2",
                "value": ""
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "maintenanceids": [
            "3"
        ]
    },
    "id": 1
}
```

Creación de un mantenimiento ad hoc

Cree un mantenimiento ad hoc de una hora con recopilación de datos para el grupo de hosts con ID "4", el trigger con ID "12345" y el nombre del evento **Printer: ink low**.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "maintenance.create",
    "params": {
        "name": "Ad-hoc maintenance",
        "active_since": 1792057740,
        "active_till": 1792061340,
        "tags_evaltype": 0,
        "groups": [
            {"groupid": "4"}
        ],
        "triggers": [
            {"triggerid": "12345"}
        ],
        "timeperiods": [
            {
                "period": 3600,
                "timeperiod_type": 0,
                "start_date": 1792057740
            }
        ],
        "event_names": [
            {
                "operator": "2",
                "value": "Printer: ink low"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "maintenanceids": [
            "4"
        ]
    },
    "id": 1
}
```

Véase también

- Host
- Grupo de hosts
- Trigger
- Nombre del evento
- Etiqueta del problema
- Período de tiempo

Fuente

CMaintenance::create() en *ui/include/classes/api/services/CMaintenance.php*.

maintenance.delete

Descripción

`object maintenance.delete(array maintenanceIds)`

Este método le permite eliminar periodos de mantenimiento.

Note:
Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(array)` IDs de los periodos de mantenimiento a eliminar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los periodos de mantenimiento eliminados bajo la propiedad `maintenanceids`.

Ejemplos

Eliminando varios periodos de mantenimiento

Eliminar dos periodos de mantenimiento.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "maintenance.delete",
    "params": [
        "3",
        "1"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "maintenanceids": [
            "3",
            "1"
        ]
    },
    "id": 1
}
```

Fuente

CMaintenance::delete() en *ui/include/classes/api/services/CMaintenance.php*.

maintenance.get

Descripción

`integer/array maintenance.get(object parameters)`

El método permite recuperar mantenimientos según los parámetros proporcionados.

Note:
Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object)` Parámetros que definen el resultado deseado.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
groupids	ID/array	Devuelve únicamente los mantenimientos asignados a los grupos de hosts indicados.
hostids	ID/array	Devuelve únicamente los mantenimientos asignados a los hosts indicados.
triggerids	ID/array	Devuelve únicamente los mantenimientos asignados a los triggers indicados.
maintenanceids	ID/array	Devuelve únicamente los mantenimientos con los ID indicados.
selectHostGroups	query	Devuelve una propiedad `hostgroups` con los grupos de hosts asignados al mantenimiento.
selectHosts	query	Devuelve una propiedad `hosts` con los hosts asignados al mantenimiento.
selectTriggers	query	Devuelve una propiedad `triggers` con los triggers asignados al mantenimiento.
selectEventNames	query	Devuelve una propiedad `event_names` con los nombres de eventos asignados al mantenimiento.
selectTags	query	Devuelve una propiedad `tags` con las etiquetas de problema del mantenimiento.
selectTimeperiods	query	Devuelve una propiedad `timeperiods` con los períodos de tiempo del mantenimiento.
sortfield	string/array	Ordena el resultado por las propiedades indicadas. Valores posibles: `maintenanceid`, `name`, `maintenance_type`, `active_since`, `active_till`.
countOutput	boolean	Estos parámetros se describen en los comentarios de referencia.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

`(entero/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperación de mantenimientos

Recupere todos los mantenimientos configurados, los datos sobre el alcance del mantenimiento asignado (grupo de hosts, host, trigger, nombre del evento, etiquetas) y los períodos de tiempo definidos.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "maintenance.get",
    "params": {
        "output": "extend",
        "selectHostGroups": "extend",
        "selectHosts": "extend",
        "selectTriggers": "extend",
        "selectEventNames": "extend",
        "selectTags": "extend",
        "selectTimeperiods": "extend"
    },
```

```
    "id": 1
}
```

Respuesta:


```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "maintenanceid": "1",
      "name": "Mantenimiento periódico de la base de datos",
      "maintenance_type": "0",
      "description": "",
      "active_since": "1791936000",
      "active_till": "2107555200",
      "tags_evaltype": "0",
      "hostgroups": [
        {
          "groupid": "20",
          "name": "Bases de datos",
          "flags": "0",
          "uuid": "748ad4d098d447d492bb935c907f652f"
        }
      ],
      "triggers": [],
      "event_names": [],
      "hosts": [],
      "tags": [],
      "timeperiods": [
        {
          "timeperiod_type": "4",
          "every": "1",
          "month": "4095",
          "dayofweek": "0",
          "day": "1",
          "start_time": "64800",
          "period": "3600",
          "start_date": "0"
        }
      ]
    },
    {
      "maintenanceid": "2",
      "name": "Ad hoc: Impresora: tinta AGOTADA",
      "maintenance_type": "0",
      "description": "",
      "active_since": "1792057740",
      "active_till": "1792061340",
      "tags_evaltype": "0",
      "hostgroups": [
        {
          "groupid": "4",
          "name": "Servidores Zabbix",
          "flags": "0",
          "uuid": "6f6799aa69e844b4b3918f779f2abf08"
        }
      ],
      "triggers": [],
      "event_names": [
        {
          "operator": "2",
          "value": "Impresora: tinta AGOTADA"
        }
```

```
      ],
      "hosts": [],
      "tags": [],
      "timeperiods": [
        {
          "timeperiod_type": "0",
          "every": "1",
          "month": "0",
          "dayofweek": "0",
          "day": "0",
          "start_time": "0",
          "period": "3600",
          "start_date": "1792057740"
        }
      ]
    }
  ],
  "id": 1
}
```


Véase también

- Host
- Grupo de hosts
- Trigger
- Nombre del evento
- Etiqueta del problema
- Período de tiempo

Fuente

CMaintenance::get() en *ui/include/classes/api/services/CMaintenance.php*.

maintenance.update

Descripción

`object maintenance.update(object/array maintenances)`

Este método le permite actualizar mantenimientos existentes.

Note:
Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object/array)` Propiedades de mantenimiento que se actualizarán.

La propiedad `maintenanceid` debe definirse para cada mantenimiento; todas las demás propiedades son opcionales. Solo se actualizarán las propiedades pasadas; las demás permanecerán sin cambios.

Además de las propiedades de mantenimiento estándar, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
groups	object/array	Grupos de hosts que reemplazarán los grupos actuales. Los grupos de hosts deben tener definida únicamente la propiedad `groupid`. Comportamiento del parámetro: - *obligatorio* si no se establece `hosts`

Parámetro	Tipo	Descripción
hosts	object/array	Hosts que reemplazarán los hosts actuales. Los hosts deben tener definida únicamente la propiedad `hostid`. Comportamiento del parámetro: - *obligatorio* si no se establece `groups`
triggers	object/array	Triggers que reemplazarán los triggers actuales. Los triggers deben tener definida únicamente la propiedad `triggerid`. Comportamiento del parámetro: - *obligatorio* si no se establecen `groups` ni `hosts` - *compatible* si `maintenance_type`=0 (con recopilación de datos)
event_names	object/array	Nombres de eventos que reemplazarán los nombres de eventos actuales. Comportamiento del parámetro: - *compatible* si `maintenance_type`=0 (con recopilación de datos)
tags	object/array	Etiquetas de problemas que reemplazarán las etiquetas actuales. Comportamiento del parámetro: - *compatible* si `maintenance_type` del objeto Maintenance está establecido en "con recopilación de datos"
timeperiods	object/array	Periodos de tiempo de mantenimiento que reemplazarán los periodos actuales.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los mantenimientos actualizados bajo la propiedad `maintenanceids`.

Ejemplos

Asignar diferentes equipos

Reemplace los equipos actualmente asignados al mantenimiento por otros dos diferentes.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "maintenance.update",
    "params": {
        "maintenanceid": "3",
        "hosts": [
            {"hostid": "10085"},
            {"hostid": "10084"}
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "maintenanceids": [
            "3"
        ]
    },
    "id": 1
}
```

Ver también

- Período de tiempo

Fuente

CMaintenance::update() en *ui/include/classes/api/services/CMaintenance.php*.

Mapa de iconos

Esta clase está diseñada para trabajar con mapas de iconos.

Referencias de objetos:

- Mapa de iconos
- Asignación de iconos

Métodos disponibles:

- iconmap.create - crear nuevos mapas de iconos
- iconmap.delete - eliminar mapas de iconos
- iconmap.get - obtener mapas de iconos
- iconmap.update - actualizar mapas de iconos

Objeto de mapa de iconos

Los siguientes objetos están directamente relacionados con la API `iconmap`.

Mapa de iconos

El objeto mapa de iconos tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
iconmapid	ID	ID del mapa de iconos. Comportamiento de la propiedad: - *solo lectura* - *requerido* para operaciones de actualización
default_iconid	ID	ID del icono por defecto. Comportamiento de la propiedad: - *requerido* para operaciones de creación
name	string	Nombre del mapa de iconos. Comportamiento de la propiedad: - *requerido* para operaciones de creación

Asignación de iconos

El objeto de asignación de iconos define un icono específico que se utilizará para los hosts con un determinado valor de campo de inventario. Tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
iconid	ID	ID del icono utilizado por la asignación de iconos. Comportamiento de la propiedad: - *obligatorio*
expression	string	Expresión para comparar con el campo de inventario. Comportamiento de la propiedad: - *obligatorio*

Propiedad	Tipo	Descripción
inventory_link	integer	ID del campo de inventario del host. Consulte el objeto de inventario de host para ver una lista de los campos de inventario soportados. Comportamiento de la propiedad: - *obligatorio*
sortorder	integer	Posición de la asignación de iconos en el mapa de iconos. Comportamiento de la propiedad: - *solo lectura*

iconmap.create

Descripción

`object iconmap.create(object/array iconMaps)`

Este método permite crear nuevos mapas de iconos.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(objeto/array)` Mapas de iconos a crear.

Además de las propiedades estándar del mapa de iconos, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
mappings	array	Asignaciones de iconos que se crearán para el mapa de iconos. Comportamiento del parámetro: - *obligatorio*

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los mapas de iconos creados bajo la propiedad `iconmapids`. El orden de los IDs devueltos coincide con el orden de los mapas de iconos pasados.

Ejemplos

Crear un mapa de iconos

Cree un mapa de iconos para mostrar equipos de diferentes tipos.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "iconmap.create",
    "params": {
        "name": "Type icons",
        "default_iconid": "2",
        "mappings": [
            {
                "inventory_link": 1,
                "expression": "server",
                "iconid": "3"
            },
            {
                "inventory_link": 1,
```

```
                "expression": "switch",
                "iconid": "4"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "iconmapids": [
            "2"
        ]
    },
    "id": 1
}
```

Ver también

- Mapeo de iconos

Fuente

CIconMap::create() en *ui/include/classes/api/services/CIconMap.php*.

iconmap.delete

Descripción

`object iconmap.delete(array iconMapIds)`

Este método permite eliminar mapas de iconos.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(array)` IDs de los mapas de iconos a eliminar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los mapas de iconos eliminados bajo la propiedad `iconmapids`.

Ejemplos

Eliminar varios mapas de iconos

Eliminar dos mapas de iconos.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "iconmap.delete",
    "params": [
        "2",
        "5"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "iconmapids": [
            "2",
            "5"
        ]
    },
    "id": 1
}
```

Fuente

CIconMap::delete() en *ui/include/classes/api/services/CIconMap.php*.

iconmap.get

Descripción

`integer/array iconmap.get(object parameters)`

El método permite recuperar mapas de iconos según los parámetros proporcionados.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
iconmapids	ID/array	Devolver solo los mapas de iconos con los IDs dados.
sysmapids	ID/array	Devolver solo los mapas de iconos que se utilizan en los mapas dados.
selectMappings	query	Devolver una propiedad `mappings` con las asignaciones de iconos utilizadas.
sortfield	string/array	Ordenar el resultado por las propiedades dadas. Valores posibles: `iconmapid`, `name`.
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperar un mapa de iconos

Recuperar todos los datos sobre el mapa de iconos "3".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "iconmap.get",
    "params": {
        "iconmapids": "3",
        "output": "extend",
        "selectMappings": "extend"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "mappings": [
                {
                    "iconmappingid": "3",
                    "iconmapid": "3",
                    "iconid": "6",
                    "inventory_link": "1",
                    "expression": "server",
                    "sortorder": "0"
                },
                {
                    "iconmappingid": "4",
                    "iconmapid": "3",
                    "iconid": "10",
                    "inventory_link": "1",
                    "expression": "switch",
                    "sortorder": "1"
                }
            ],
            "iconmapid": "3",
            "name": "Host type icons",
            "default_iconid": "2"
        }
    ],
    "id": 1
}
```

Ver también

- Mapeo de iconos

Fuente

CIconMap::get() en *ui/include/classes/api/services/CIconMap.php*.

iconmap.update

Descripción

`object iconmap.update(object/array iconMaps)`

Este método permite actualizar icon maps existentes.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

(objeto/array) Propiedades del mapa de iconos que se van a actualizar.

La propiedad iconmapid debe definirse para cada mapa de iconos, todas las demás propiedades son opcionales. Sólo se actualizarán las propiedades pasadas, todas las demás permanecerán sin cambios.

Además de las propiedades estándar del mapa de iconos, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
mappings	array	Asignaciones de iconos para reemplazar las asignaciones de iconos existentes.

Valores de retorno

(object) Devuelve un objeto que contiene los IDs de los mapas de iconos actualizados bajo la propiedad iconmapids.

Ejemplos

Renombrar mapa de iconos

Renombrar un mapa de iconos a "OS icons".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "iconmap.update",
    "params": {
        "iconmapid": "1",
        "name": "OS icons"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "iconmapids": [
            "1"
        ]
    },
    "id": 1
}
```

Ver también

- Mapeo de iconos

Fuente

CIconMap::update() en *ui/include/classes/api/services/CIconMap.php*.

Mapa de valores

Esta clase está diseñada para trabajar con mapas de valores.

Referencias de objetos:

- Mapa de valores
  - Mapeos de valores

Métodos disponibles:

- valuemap.create - crear nuevos mapas de valores
- valuemap.delete - eliminar mapas de valores
- valuemap.get - obtener mapas de valores
- valuemap.update - actualizar mapas de valores

Objeto de mapa de valores

Los siguientes objetos están directamente relacionados con la API de `valuemap`.

Mapa de valores

El objeto mapa de valores tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
valuemapid	ID	ID del mapa de valores. Comportamiento de la propiedad: - *solo lectura* - *requerido* para operaciones de actualización
hostid	ID	ID del host o template al que pertenece el mapa de valores. Comportamiento de la propiedad: - *constante* - *requerido* para operaciones de creación
name	string	Nombre del mapa de valores. Comportamiento de la propiedad: - *requerido* para operaciones de creación
mappings	array	Mapeos de valores para el mapa de valores actual. El objeto de mapeo se describe en detalle más abajo. Comportamiento de la propiedad: - *requerido* para operaciones de creación
uuid	string	Identificador único universal, utilizado para vincular mapas de valores importados con los ya existentes. Se genera automáticamente si no se proporciona. Comportamiento de la propiedad: - *soportado* si el mapa de valores pertenece a un template

Mapeos de valores

El objeto de mapeos de valores define los mapeos de valores del mapa de valores. Tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
type	integer	Tipo de coincidencia de mapeo. Valores posibles: 0 - *(por defecto)* el mapeo se aplicará si el valor es igual; 1 - el mapeo se aplicará si el valor es mayor o igual[1]; 2 - el mapeo se aplicará si el valor es menor o igual[1]; 3 - el mapeo se aplicará si el valor está en el rango (los rangos son inclusivos; se pueden definir múltiples rangos, separados por comas)[1]; 4 - el mapeo se aplicará si el valor coincide con una expresión regular[2]; 5 - si no se encuentran coincidencias, el mapeo no se aplicará y se utilizará el valor por defecto. Si `type` se establece en ”0”, ”1”, ”2”, ”3”, ”4”, entonces `value` no puede estar vacío. Si `type` se establece en ”5”, entonces `value` debe estar vacío.
value	string	Valor original. Comportamiento de la propiedad: - *requerido* si `type` se establece en ”1”, ”2”, ”3”, ”4” - *soportado* si `type` se establece en ”5”

Propiedad	Tipo	Descripción
newvalue	string	Valor al que se mapea el valor original. Comportamiento de la propiedad: *- requerido*

[1] soportado solo para items con tipo de valor ”numérico sin signo”, ”numérico flotante”.
 [2] soportado solo para items con tipo de valor ”carácter”.

valuemap.create

Descripción

`object valuemap.create(object/array valuemaps)`

Este método permite crear nuevos mapas de valores.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object/array)` Mapas de valores a crear.

El método acepta mapas de valores con las propiedades estándar de los mapas de valores.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los mapas de valores creados en la propiedad `valuemapids`. El orden de los IDs devueltos coincide con el orden de los mapas de valores enviados.

Ejemplos

Creando un mapa de valores

Cree un mapa de valores con dos asignaciones.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "valuemap.create",
    "params": {
        "hostid": "50009",
        "name": "Estado del servicio",
        "mappings": [
            {
                "type": "1",
                "value": "1",
                "newvalue": "Arriba"
            },
            {
                "type": "5",
                "newvalue": "Abajo"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "valuemapids": [
```

```
            "1"
        ]
    },
    "id": 1
}
```

Fuente

CValueMap::create() en *ui/include/classes/api/services/CValueMap.php*.

valuemap.delete

Descripción

`object valuemap.delete(array valuemapids)`

Este método le permite eliminar mapas de valores.

Note:
Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración del rol de usuario. Consulte User roles para obtener más información.

Parámetros

`(array)` IDs de los mapas de valores a eliminar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los mapas de valores eliminados bajo la propiedad `valuemapids`.

Ejemplos

Eliminando varios mapas de valores

Eliminar dos mapas de valores.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "valuemap.delete",
    "params": [
        "1",
        "2"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "valuemapids": [
            "1",
            "2"
        ]
    },
    "id": 1
}
```

Fuente

CValueMap::delete() en *ui/include/classes/api/services/CValueMap.php*.

valuemap.get

Descripción

`integer/array valuemap.get(object parameters)`

El método permite recuperar mapas de valores según los parámetros proporcionados.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
valuemapids	ID/array	Devolver solo los mapas de valores con los IDs dados.
selectMappings	query	Devolver los mapeos de valores para el mapa de valores actual en la propiedad `mappings`. Admite `count`.
sortfield	string/array	Ordenar el resultado por las propiedades dadas. Valores posibles: `valuemapid`, `name`.
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperar mapas de valores

Recuperar todos los mapas de valores configurados.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "valuemap.get",
    "params": {
        "output": "extend"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "valuemapid": "4",
```

```
            "name": "APC Battery Replacement Status"
        },
        {
            "valuemapid": "5",
            "name": "APC Battery Status"
        },
        {
            "valuemapid": "7",
            "name": "Dell Open Manage System Status"
        }
    ],
    "id": 1
}
```

Recuperar un mapa de valores con sus asignaciones.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "valuemap.get",
    "params": {
        "output": "extend",
        "selectMappings": "extend",
        "valuemapids": ["4"]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "valuemapid": "4",
            "name": "APC Battery Replacement Status",
            "mappings": [
                {
                    "type": "0",
                    "value": "1",
                    "newvalue": "unknown"
                },
                {
                    "type": "0",
                    "value": "2",
                    "newvalue": "notInstalled"
                },
                {
                    "type": "0",
                    "value": "3",
                    "newvalue": "ok"
                },
                {
                    "type": "0",
                    "value": "4",
                    "newvalue": "failed"
                },
                {
                    "type": "0",
                    "value": "5",
                    "newvalue": "highTemperature"
                },
                {
                    "type": "0",
```

```
                    "value": "6",
                    "newvalue": "replaceImmediately"
                },
                {
                    "type": "0",
                    "value": "7",
                    "newvalue": "lowCapacity"
                }
            ]
        }
    ],
    "id": 1
}
```

Fuente

CValueMap::get() en *ui/include/classes/api/services/CValueMap.php*.

valuemap.update

Descripción

`object valuemap.update(object/array valuemaps)`

Este método permite actualizar mapas de valores existentes.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object/array)` Propiedades del mapa de valores que se actualizarán.

La propiedad `valuemapid` debe estar definida para cada mapa de valores, todas las demás propiedades son opcionales. Sólo se actualizarán las propiedades pasadas, todas las demás permanecerán sin cambios.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los mapas de valores actualizados bajo la propiedad `valuemapids`.

Ejemplos

Cambiar el nombre del mapa de valores

Cambie el nombre del mapa de valores a ”Device status”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "valuemap.update",
    "params": {
        "valuemapid": "2",
        "name": "Device status"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "valuemapids": [
            "2"
        ]
    },
```

```
    "id": 1
}
```

Cambiando asignaciones para un mapa de valores.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "valuemap.update",
    "params": {
        "valuemapid": "2",
        "mappings": [
            {
                "type": "0",
                "value": "0",
                "newvalue": "Online"
            },
            {
                "type": "0",
                "value": "1",
                "newvalue": "Offline"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "valuemapids": [
            "2"
        ]
    },
    "id": 1
}
```

Fuente

CValueMap::update() en *ui/include/classes/api/services/CValueMap.php*.

Mapas

Esta clase está diseñada para trabajar con mapas.

Referencias de objetos:

- Mapa
- Elemento de mapa
  - Elemento de mapa Host
  - Elemento de mapa Grupo de hosts
  - Elemento de mapa Mapa
  - Elemento de mapa Trigger
  - Elemento de mapa Etiqueta
  - Elemento de mapa URL
- Enlace de mapa
  - Enlace de mapa trigger
  - Indicadores de enlace de mapa
- URL de mapa
- Usuario de mapa
- Grupo de usuarios de mapa
- Formas de mapa
- Líneas de mapa

Métodos disponibles:

- map.create - crear nuevos mapas
- map.delete - eliminar mapas
- map.get - obtener mapas
- map.update - actualizar mapas

Objeto de mapa

Los siguientes objetos están directamente relacionados con la API de `map`.

Mapa

El objeto mapa tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
sysmapid	ID	ID del mapa. Comportamiento de la propiedad: - *solo lectura* - *requerido* para operaciones de actualización
height	integer	Altura del mapa en píxeles. Comportamiento de la propiedad: - *requerido* para operaciones de creación
name	string	Nombre del mapa. Comportamiento de la propiedad: - *requerido* para operaciones de creación
width	integer	Ancho del mapa en píxeles. Comportamiento de la propiedad: - *requerido* para operaciones de creación
backgroundid	ID	ID de la imagen utilizada como fondo para el mapa.
background_scale	integer	Si se habilita el escalado proporcional de la imagen de fondo. Valores posibles: 0 - deshabilitar el escalado; 1 - *(por defecto)* habilitar el escalado de la imagen.
expand_macros	integer	Si se expanden las macros en las etiquetas al configurar el mapa. Valores posibles: 0 - *(por defecto)* no expandir macros; 1 - expandir macros.
expandproblem	integer	Si el trigger de problema se mostrará para elementos con un solo problema. Valores posibles: 0 - mostrar siempre el número de problemas; 1 - *(por defecto)* mostrar el trigger de problema si solo hay un problema.
grid_align	integer	Si se habilita la alineación de la cuadrícula. Valores posibles: 0 - deshabilitar la alineación de la cuadrícula; 1 - *(por defecto)* habilitar la alineación de la cuadrícula.
grid_show	integer	Si se muestra la cuadrícula en el mapa. Valores posibles: 0 - no mostrar la cuadrícula; 1 - *(por defecto)* mostrar la cuadrícula.

Propiedad	Tipo	Descripción
grid_size	integer	Tamaño de la cuadrícula del mapa en píxeles. Valores soportados: 20, 40, 50, 75 y 100. Por defecto: 50.
highlight	integer	Si la función de resaltar iconos está habilitada. Valores posibles: 0 - resaltado deshabilitado; 1 - *(por defecto)* resaltado habilitado.
iconmapid	ID	ID del mapa de iconos utilizado en el mapa.
label_format	integer	Si se habilitan las etiquetas avanzadas. Valores posibles: 0 - *(por defecto)* deshabilitar etiquetas avanzadas; 1 - habilitar etiquetas avanzadas.
label_location	integer	Ubicación de la etiqueta del elemento del mapa. Valores posibles: 0 - *(por defecto)* abajo; 1 - izquierda; 2 - derecha; 3 - arriba.
label_string_host	string	Etiqueta personalizada para elementos de host. Comportamiento de la propiedad: - *requerido* si `label_type_host` está establecido en ”custom”
label_string_hostgroup	string	Etiqueta personalizada para elementos de grupo de hosts. Comportamiento de la propiedad: - *requerido* si `label_type_hostgroup` está establecido en ”custom”
label_string_image	string	Etiqueta personalizada para elementos de imagen. Comportamiento de la propiedad: - *requerido* si `label_type_image` está establecido en ”custom”
label_string_map	string	Etiqueta personalizada para elementos de mapa. Comportamiento de la propiedad: - *requerido* si `label_type_map` está establecido en ”custom”
label_string_trigger	string	Etiqueta personalizada para elementos de trigger. Comportamiento de la propiedad: - *requerido* si `label_type_trigger` está establecido en ”custom”
label_type	integer	Tipo de etiqueta del elemento del mapa. Valores posibles: 0 - etiqueta; 1 - dirección IP; 2 - *(por defecto)* nombre del elemento; 3 - solo estado; 4 - nada.
label_type_host	integer	Tipo de etiqueta para elementos de host. Valores posibles: 0 - etiqueta; 1 - dirección IP; 2 - *(por defecto)* nombre del elemento; 3 - solo estado; 4 - nada; 5 - personalizado.

Propiedad	Tipo	Descripción
label_type_hostgroup	integer	Tipo de etiqueta para elementos de grupo de hosts. Valores posibles: 0 - etiqueta; 2 - *(por defecto)* nombre del elemento; 3 - solo estado; 4 - nada; 5 - personalizado.
label_type_image	integer	Tipo de etiqueta para elementos de grupo de hosts. Valores posibles: 0 - etiqueta; 2 - *(por defecto)* nombre del elemento; 4 - nada; 5 - personalizado.
label_type_map	integer	Tipo de etiqueta para elementos de mapa. Valores posibles: 0 - etiqueta; 2 - *(por defecto)* nombre del elemento; 3 - solo estado; 4 - nada; 5 - personalizado.
label_type_trigger	integer	Tipo de etiqueta para elementos de trigger. Valores posibles: 0 - etiqueta; 2 - *(por defecto)* nombre del elemento; 3 - solo estado; 4 - nada; 5 - personalizado.
markelements	integer	Si se resaltan los elementos del mapa que han cambiado recientemente su estado. Valores posibles: 0 - *(por defecto)* no resaltar elementos; 1 - resaltar elementos.
severity_min	integer	Severidad mínima de los triggers que se mostrarán en el mapa. Consulte la propiedad `severity` de trigger para ver una lista de severidades de trigger soportadas.
show_element_label	integer	Cómo mostrar las etiquetas de los elementos por defecto. Valores posibles: 0 - mostrar siempre; 1 - *(por defecto)* ocultar automáticamente.
show_link_label	integer	Cómo mostrar las etiquetas de los enlaces por defecto. Valores posibles: 0 - mostrar siempre; 1 - *(por defecto)* ocultar automáticamente.
show_unack	integer	Cómo se deben mostrar los problemas. Valores posibles: 0 - *(por defecto)* mostrar el recuento de todos los problemas; 1 - mostrar solo el recuento de problemas no reconocidos; 2 - mostrar el recuento de problemas reconocidos y no reconocidos por separado.
userid	ID	ID del usuario que es el propietario del mapa.

Propiedad	Tipo	Descripción
private	integer	Tipo de compartición del mapa. Valores posibles: 0 - mapa público; 1 - *(por defecto)* mapa privado.
show_suppressed	integer	Si se muestran los problemas suprimidos. Valores posibles: 0 - *(por defecto)* ocultar problemas suprimidos; 1 - mostrar problemas suprimidos.

Elemento de mapa

El objeto de elemento de mapa define un objeto mostrado en un mapa. Tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
selementid	ID	ID del elemento de mapa. Comportamiento de la propiedad: - *solo lectura*
elements	array	Objeto de datos Elemento. Comportamiento de la propiedad: - *requerido* si `elementtype` está establecido en ”host”, ”map”, ”trigger” o ”grupo de hosts”
elementtype	integer	Tipo de elemento de mapa. Valores posibles: 0 - host; 1 - mapa; 2 - trigger; 3 - grupo de hosts; 4 - imagen. Comportamiento de la propiedad: - *requerido*
iconid_off	ID	ID de la imagen utilizada para mostrar el elemento en estado por defecto. Comportamiento de la propiedad: - *requerido*
areatype	integer	Cómo se deben mostrar los hosts del grupo de hosts. Valores posibles: 0 - *(por defecto)* el elemento del grupo de hosts ocupará todo el mapa; 1 - el elemento del grupo de hosts tendrá un tamaño fijo.
elementsubtype	integer	Cómo se debe mostrar un elemento de grupo de hosts en un mapa. Valores posibles: 0 - *(por defecto)* mostrar el grupo de hosts como un solo elemento; 1 - mostrar cada host del grupo por separado.
evaltype	integer	Condición de filtrado de etiquetas de elementos de mapa método de evaluación. Valores posibles: 0 - *(por defecto)* And/Or; 2 - Or.
height	integer	Altura del elemento de grupo de hosts de tamaño fijo en píxeles. Por defecto: 200.

Propiedad	Tipo	Descripción
iconid_disabled	ID	ID de la imagen utilizada para mostrar elementos de mapa deshabilitados. Comportamiento de la propiedad: - *soportado* si `elementtype` está establecido en ”host”, ”map”, ”trigger” o ”grupo de hosts”
iconid_maintenance	ID	ID de la imagen utilizada para mostrar elementos de mapa en mantenimiento. Comportamiento de la propiedad: - *soportado* si `elementtype` está establecido en ”host”, ”map”, ”trigger” o ”grupo de hosts”
iconid_on	ID	ID de la imagen utilizada para mostrar elementos de mapa con problemas. Comportamiento de la propiedad: - *soportado* si `elementtype` está establecido en ”host”, ”map”, ”trigger” o ”grupo de hosts”
label	string	Etiqueta del elemento.
label_location	integer	Ubicación de la etiqueta del elemento de mapa. Valores posibles: -1 - *(por defecto)* ubicación por defecto; 0 - abajo; 1 - izquierda; 2 - derecha; 3 - arriba.
permission	integer	Tipo de nivel de permiso. Valores posibles: -1 - ninguno; 2 - solo lectura; 3 - lectura-escritura.
show_label	integer	Cómo mostrar la etiqueta del elemento. Valores posibles: -1 - *(por defecto)* por defecto del mapa; 0 - mostrar siempre; 1 - ocultar automáticamente.
sysmapid	ID	ID del mapa al que pertenece el elemento. Comportamiento de la propiedad: - *solo lectura*
urls	array	URLs de elementos de mapa. El objeto URL de elemento de mapa se describe en detalle más abajo.
use_iconmap	integer	Si se debe usar la asignación de iconos para elementos de host. Valores posibles: 0 - no usar asignación de iconos; 1 - *(por defecto)* usar asignación de iconos.
viewtype	integer	Algoritmo de colocación de elementos de grupo de hosts. Valores posibles: 0 - *(por defecto)* cuadrícula.
width	integer	Ancho del elemento de grupo de hosts de tamaño fijo en píxeles. Por defecto: 200.
x	integer	Coordenadas X del elemento en píxeles. Por defecto: 0.

Propiedad	Tipo	Descripción
y	integer	Coordenadas Y del elemento en píxeles. Por defecto: 0.
zindex	integer	Valor utilizado para ordenar los elementos del mapa (z-index). Por defecto: 0.

Elemento de mapa Host

El objeto de elemento de mapa Host define un elemento de host.

Propiedad	Tipo	Descripción
hostid	ID	ID del host.

Elemento de mapa Grupo de hosts

El objeto de elemento de mapa Grupo de hosts define un elemento de grupo de hosts.

Propiedad	Tipo	Descripción
groupid	ID	ID del grupo de hosts.

Mapa de elementos Mapa

El objeto de mapa de elementos Mapa define un elemento de mapa.

Propiedad	Tipo	Descripción
sysmapid	ID	ID del mapa.

Elemento de mapa Trigger

El objeto de elemento de mapa Trigger define uno o más elementos de trigger.

Propiedad	Tipo	Descripción
triggerid	ID	ID del trigger.

Etiqueta de elemento de mapa

El objeto de etiqueta de elemento de mapa tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
tag	string	Nombre de la etiqueta del elemento de mapa. Comportamiento de la propiedad: - *requerido*
operator	integer	Operador de condición de etiqueta de elemento de mapa. Valores posibles: 0 - *(por defecto)* Contiene; 1 - Es igual a; 2 - No contiene; 3 - No es igual a; 4 - Existe; 5 - No existe.
value	string	Valor de la etiqueta del elemento de mapa.

URL del elemento del mapa

El objeto de URL del elemento del mapa define un enlace clicable que estará disponible para un elemento de mapa específico. Tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
sysmapelementurlid	ID	ID de la URL del elemento del mapa. Comportamiento de la propiedad: - *solo lectura*
name	string	Título del enlace. Comportamiento de la propiedad: - *requerido*
url	string	URL del enlace. Comportamiento de la propiedad: - *requerido*
selementid	ID	ID del elemento del mapa al que pertenece la URL.

Enlace de mapa

El objeto enlace de mapa define un enlace entre dos elementos del mapa. Tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
linkid	ID	ID del enlace de mapa. Comportamiento de la propiedad: - *solo lectura*
sysmapid	ID	ID del mapa al que pertenece el enlace.
selementid1	ID	ID del primer elemento del mapa enlazado en un extremo. Comportamiento de la propiedad: - *requerido*
selementid2	ID	ID del primer elemento del mapa enlazado en el otro extremo. Comportamiento de la propiedad: - *requerido*
drawtype	integer	Estilo de línea del enlace. Valores posibles: 0 - *(por defecto)* línea; 2 - línea en negrita; 3 - línea de puntos; 4 - línea discontinua.
color	string	Color de la línea como código hexadecimal. Por defecto: 000000.
label	string	Etiqueta del enlace.
show_label	integer	Cómo mostrar la etiqueta del enlace. Valores posibles: -1 - *(por defecto)* valor por defecto del mapa; 0 - mostrar siempre; 1 - ocultar automáticamente.
indicator_type	integer	Seleccionar el tipo de indicador del enlace. Valores posibles: 0 - *(por defecto)* enlace estático; 1 - trigger; 2 - valor del item.

Propiedad	Tipo	Descripción
linktriggers	array	Triggers de enlace de mapa a usar como indicadores de estado del enlace. El objeto trigger de enlace de mapa se describe en detalle más abajo. Comportamiento de la propiedad: - *requerido* si `indicator_type` está establecido en ”trigger” (1).
itemid	ID	ID del item. Comportamiento de la propiedad: - *requerido* si `indicator_type` está establecido en ”valor del item” (2).
highlights	array	Resaltados de enlace de mapa a usar como indicadores de estado del enlace. El objeto indicadores de enlace de mapa se describe en detalle más abajo. Comportamiento de la propiedad: - *requerido* si `indicator_type` está establecido en ”valor del item” (2).
thresholds	array	Umbrales de enlace de mapa a usar como indicadores de estado del enlace. El objeto indicadores de enlace de mapa se describe en detalle más abajo. Comportamiento de la propiedad: - *requerido* si `indicator_type` está establecido en ”valor del item” (2).
permission	integer	Tipo de nivel de permiso. Valores posibles: -1 - ninguno; 2 - solo lectura; 3 - lectura-escritura.

Trigger de enlace de mapa

El objeto trigger de enlace de mapa define un indicador de estado de enlace de mapa basado en el estado de un trigger. Tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
triggerid	ID	ID del trigger utilizado como indicador de enlace. Comportamiento de la propiedad: - *obligatorio*
color	string	Color del indicador como código de color hexadecimal. Por defecto: `DD0000`.
drawtype	integer	Estilo de dibujo del indicador. Valores posibles: 0 - *(por defecto)* línea; 2 - línea en negrita; 3 - línea de puntos; 4 - línea discontinua.

Indicadores de enlaces de mapa

El objeto de indicadores de enlaces de mapa define un indicador de estado del enlace de mapa basado en el valor del item. Tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
pattern	string	Expresión regular para comparar. Disponible solo para resaltados.
sortorder	integer	Se utiliza para definir el orden de los resaltados. Disponible solo para resaltados.
threshold	string	Valor numérico para comparar. Disponible solo para umbrales.
drawtype	integer	Estilo de dibujo del indicador. Valores posibles: 0 - *(por defecto)* línea; 2 - línea en negrita; 3 - línea de puntos; 4 - línea discontinua.
color	string	Color del indicador como código de color hexadecimal. Por defecto: `DD0000`. Comportamiento de la propiedad: - *obligatorio*

URL de mapa

El objeto URL de mapa define un enlace clicable que estará disponible para todos los elementos de un tipo específico en el mapa. Tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
sysmapurlid	ID	ID de la URL del mapa. Comportamiento de la propiedad: - *solo lectura*
name	string	Título del enlace. Comportamiento de la propiedad: - *requerido*
url	string	URL del enlace. Comportamiento de la propiedad: - *requerido*
elementtype	integer	Tipo de elemento de mapa para el que estará disponible la URL. Consulte la propiedad `type` del elemento de mapa para obtener una lista de los tipos soportados. Por defecto: 0.
sysmapid	ID	ID del mapa al que pertenece la URL.

Usuario del mapa

Lista de permisos de mapa basados en usuarios. Tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
sysmapuserid	ID	ID del usuario del mapa. Comportamiento de la propiedad: - *solo lectura*
userid	ID	ID del usuario. Comportamiento de la propiedad: - *requerido*

Propiedad	Tipo	Descripción
permission	integer	Tipo de nivel de permiso. Valores posibles: 2 - solo lectura; 3 - lectura-escritura. Comportamiento de la propiedad: - *requerido*

Grupo de usuarios del mapa

Lista de permisos de mapas basados en grupos de usuarios. Tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
sysmapusrgrpid	ID	ID del grupo de usuarios del mapa. Comportamiento de la propiedad: - *solo lectura*
usrgrpid	ID	ID del grupo de usuarios. Comportamiento de la propiedad: - *requerido*
permission	integer	Tipo de nivel de permiso. Valores posibles: 2 - solo lectura; 3 - lectura-escritura. Comportamiento de la propiedad: - *requerido*

Formas del mapa

El objeto de forma del mapa define una forma geométrica (con o sin texto) que se muestra en un mapa. Tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
sysmap_shapeid	ID	ID del elemento de forma del mapa. Comportamiento de la propiedad: - *solo lectura*
type	integer	Tipo de elemento de forma del mapa. Valores posibles: 0 - rectángulo; 1 - elipse. La propiedad es obligatoria cuando se crean nuevas formas. Comportamiento de la propiedad: - *obligatorio*
x	integer	Coordenadas X de la forma en píxeles. Por defecto: 0.
y	integer	Coordenadas Y de la forma en píxeles. Por defecto: 0.

Propiedad	Tipo	Descripción
width	integer	Ancho de la forma en píxeles. Por defecto: 200.
height	integer	Alto de la forma en píxeles. Por defecto: 200.
text	string	Texto de la forma.
font	integer	Fuente del texto dentro de la forma. Valores posibles: 0 - Georgia, serif 1 - ”Palatino Linotype”, ”Book Antiqua”, Palatino, serif 2 - ”Times New Roman”, Times, serif 3 - Arial, Helvetica, sans-serif 4 - ”Arial Black”, Gadget, sans-serif 5 - ”Comic Sans MS”, cursive, sans-serif 6 - Impact, Charcoal, sans-serif 7 - ”Lucida Sans Unicode”, ”Lucida Grande”, sans-serif 8 - Tahoma, Geneva, sans-serif 9 - ”Trebuchet MS”, Helvetica, sans-serif 10 - Verdana, Geneva, sans-serif 11 - ”Courier New”, Courier, monospace 12 - ”Lucida Console”, Monaco, monospace Por defecto: 9.
font_size	integer	Tamaño de la fuente en píxeles. Por defecto: 11.
font_color	string	Color de la fuente. Por defecto: 000000.
text_halign	integer	Alineación horizontal del texto. Valores posibles: 0 - centro; 1 - izquierda; 2 - derecha. Por defecto: 0.
text_valign	integer	Alineación vertical del texto. Valores posibles: 0 - medio; 1 - arriba; 2 - abajo. Por defecto: 0.
border_type	integer	Tipo de borde. Valores posibles: 0 - ninguno; 1 - ——————; 2 - ┅; 3 - – – –. Por defecto: 0.
border_width	integer	Ancho del borde en píxeles. Por defecto: 0.
border_color	string	Color del borde. Por defecto: 000000.

Propiedad	Tipo	Descripción
background_color	string	Color de fondo (color de relleno). Por defecto: `(vacío)`.
zindex	integer	Valor utilizado para ordenar todas las formas y líneas (z-index). Por defecto: 0.

Líneas de mapa

El objeto línea de mapa define una línea mostrada en un mapa. Tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
sysmap_shapeid	ID	ID del elemento de forma del mapa. Comportamiento de la propiedad: - *solo lectura*
x1	integer	Coordenadas X del punto 1 de la línea en píxeles. Por defecto: 0.
y1	integer	Coordenadas Y del punto 1 de la línea en píxeles. Por defecto: 0.
x2	integer	Coordenadas X del punto 2 de la línea en píxeles. Por defecto: 200.
y2	integer	Coordenadas Y del punto 2 de la línea en píxeles. Por defecto: 200.
line_type	integer	Tipo de líneas. Valores posibles: 0 - ninguna; 1 - ————; 2 - ┅; 3 - − − −. Por defecto: 0.
line_width	integer	Ancho de las líneas en píxeles. Por defecto: 0.
line_color	string	Color de la línea. Por defecto: 000000.
zindex	integer	Valor utilizado para ordenar todas las formas y líneas (z-index). Por defecto: 0.

map.create

Descripción

`object map.create(object/array maps)`

Este método permite crear nuevos maps.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

(object/array) Mapas a crear.

Además de las propiedades estándar del mapa, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
links	array	Enlaces de mapa a crear en el mapa.
selements	array	Elementos de mapa a crear en el mapa.
urls	array	URLs de mapa a crear en el mapa.
users	array	Comparticiones de usuario de mapa a crear en el mapa.
userGroups	array	Comparticiones de grupo de usuarios de mapa a crear en el mapa.
shapes	array	Formas de mapa a crear en el mapa.
lines	array	Líneas de mapa a crear en el mapa.

Note:
Para crear enlaces de mapa, deberá establecer un `selementid` de elemento de mapa en un valor arbitrario y luego usar este valor para referenciar este elemento en las propiedades `selementid1` o `selementid2` de los enlaces. Cuando se cree el elemento, este valor será reemplazado por el ID correcto generado por Zabbix. Ver ejemplo.

Valores de retorno

(object) Devuelve un objeto que contiene los IDs de los mapas creados bajo la propiedad `sysmapids`. El orden de los IDs devueltos coincide con el orden de los mapas enviados.

Ejemplos

Crear un mapa vacío

Crea un mapa sin elementos.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "map.create",
    "params": {
        "name": "Map",
        "width": 600,
        "height": 600
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "sysmapids": [
            "8"
        ]
    },
    "id": 1
}
```

Crear un mapa de equipos

Cree un mapa con dos elementos de equipo y un enlace entre ellos. Observe el uso de valores temporales "selementid1" y "selementid2" en el objeto de enlace del mapa para hacer referencia a los elementos del mapa.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "map.create",
    "params": {
        "name": "Host map",
        "width": 600,
```

```
        "height": 600,
        "selements": [
            {
                "selementid": "1",
                "elements": [
                    {"hostid": "1033"}
                ],
                "elementtype": 0,
                "iconid_off": "2"
            },

            {
                "selementid": "2",
                "elements": [
                    {"hostid": "1037"}
                ],
                "elementtype": 0,
                "iconid_off": "2"
            }
        ],
        "links": [
            {
                "selementid1": "1",
                "selementid2": "2"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "sysmapids": [
            "9"
        ]
    },
    "id": 1
}
```

Crear un mapa de disparadores

Cree un mapa con un elemento de disparador, que contenga dos disparadores.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "map.create",
    "params": {
        "name": "Trigger map",
        "width": 600,
        "height": 600,
        "selements": [
            {
                "elements": [
                    {"triggerid": "12345"},
                    {"triggerid": "67890"}
                ],
                "elementtype": 2,
                "iconid_off": "2"
            }
        ]
```

```
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "sysmapids": [
            "10"
        ]
    },
    "id": 1
}
```

Compartir mapa

Crear un mapa con dos tipos de compartición (usuario y grupo de usuarios).

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "map.create",
    "params": {
        "name": "Map sharing",
        "width": 600,
        "height": 600,
        "users": [
            {
                "userid": "4",
                "permission": "3"
            }
        ],
        "userGroups": [
            {
                "usrgrpid": "7",
                "permission": "2"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "sysmapids": [
            "9"
        ]
    },
    "id": 1
}
```

Formas de mapa

Cree un mapa con el título del nombre del mapa.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "map.create",
    "params": {
        "name": "Host map",
```

```
        "width": 600,
        "height": 600,
        "shapes": [
            {
                "type": 0,
                "x": 0,
                "y": 0,
                "width": 600,
                "height": 11,
                "text": "{MAP.NAME}"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "sysmapids": [
            "10"
        ]
    },
    "id": 1
}
```

Líneas de mapa

Crear una línea de mapa.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "map.create",
    "params": {
        "name": "Map API lines",
        "width": 500,
        "height": 500,
        "lines": [
            {
                "x1": 30,
                "y1": 10,
                "x2": 100,
                "y2": 50,
                "line_type": 1,
                "line_width": 10,
                "line_color": "009900"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "sysmapids": [
            "11"
        ]
    },
    "id": 1
```

```
}
```

Ver también

- Elemento de mapa
- Enlace de mapa
- URL de mapa
- Usuario de mapa
- Grupo de usuarios de mapa
- Forma de mapa
- Línea de mapa

Fuente

CMap::create() en *ui/include/classes/api/services/CMap.php*.

map.delete

Descripción

`object map.delete(array mapIds)`

Este método le permite eliminar mapas.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(array)` IDs de los mapas a eliminar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los mapas eliminados bajo la propiedad `sysmapids`.

Ejemplos

Eliminar varios mapas

Eliminar dos mapas.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "map.delete",
    "params": [
        "12",
        "34"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "sysmapids": [
            "12",
            "34"
        ]
    },
    "id": 1
}
```

Fuente

CMap::delete() en *ui/include/classes/api/services/CMap.php*.

map.get

Descripción

`integer/array map.get(object parameters)`

El método permite recuperar mapas según los parámetros proporcionados.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
sysmapids	ID/array	Devuelve solo los mapas con los IDs dados.
userids	ID/array	Devuelve solo los mapas que pertenecen a los IDs de usuario dados.
expandUrls	flag	Agrega las URLs globales del mapa a los elementos de mapa correspondientes y expande las macros en todas las URLs de los elementos del mapa.
selectIconMap	query	Devuelve una propiedad `iconmap` con el mapa de iconos utilizado en el mapa.
selectLinks	query	Devuelve una propiedad `links` con los enlaces de mapa entre elementos.
selectSelements	query	Devuelve una propiedad `selements` con los elementos del mapa.
selectUrls	query	Devuelve una propiedad `urls` con las URLs del mapa.
selectUsers	query	Devuelve una propiedad `users` con los usuarios con los que se comparte el mapa.
selectUserGroups	query	Devuelve una propiedad `userGroups` con los grupos de usuarios con los que se comparte el mapa.
selectShapes	query	Devuelve una propiedad `shapes` con las formas del mapa.
selectLines	query	Devuelve una propiedad `lines` con las líneas del mapa.
sortfield	string/array	Ordena el resultado por las propiedades dadas. Valores posibles: `name`, `width`, `height`.
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperar un mapa

Recuperar todos los datos sobre el mapa "3".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "map.get",
    "params": {
        "output": "extend",
        "selectSelements": "extend",
        "selectLinks": "extend",
        "selectUsers": "extend",
        "selectUserGroups": "extend",
        "selectShapes": "extend",
        "selectLines": "extend",
        "sysmapids": "3"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "selements": [
                {
                    "selementid": "10",
                    "sysmapid": "3",
                    "elementtype": "4",
                    "evaltype": "0",
                    "iconid_off": "1",
                    "iconid_on": "0",
                    "label": "Zabbix server",
                    "label_location": "3",
                    "x": "11",
                    "y": "141",
                    "iconid_disabled": "0",
                    "iconid_maintenance": "0",
                    "elementsubtype": "0",
                    "areatype": "0",
                    "width": "200",
                    "height": "200",
                    "viewtype": "0",
                    "use_iconmap": "1",
                    "show_label": "-1",
                    "zindex":"0",
                    "elements": [],
                    "urls": [],
                    "tags": [
                        {
                            "tag": "service",
                            "value": "mysqld",
                            "operator": "0"
                        }
                    ]
                },
                {
                    "selementid": "11",
                    "sysmapid": "3",
                    "elementtype": "4",
                    "evaltype": "0",
                    "iconid_off": "1",
                    "iconid_on": "0",
                    "label": "Web server",
                    "label_location": "3",
```

```
                "x": "211",
                "y": "191",
                "iconid_disabled": "0",
                "iconid_maintenance": "0",
                "elementsubtype": "0",
                "areatype": "0",
                "width": "200",
                "height": "200",
                "viewtype": "0",
                "use_iconmap": "1",
                "show_label": "0",
                "zindex":"0",
                "elements": [],
                "urls": [],
                "tags": []
            },
            {
                "selementid": "12",
                "sysmapid": "3",
                "elementtype": "0",
                "evaltype": "0",
                "iconid_off": "185",
                "iconid_on": "0",
                "label": "{HOST.NAME}\r\n{HOST.CONN}",
                "label_location": "0",
                "x": "111",
                "y": "61",
                "iconid_disabled": "0",
                "iconid_maintenance": "0",
                "elementsubtype": "0",
                "areatype": "0",
                "width": "200",
                "height": "200",
                "viewtype": "0",
                "use_iconmap": "0",
                "show_label": "1",
                "zindex":"0",
                "elements": [
                    {
                        "hostid": "10084"
                    }
                ],
                "urls": [],
                "tags": []
            }
        ],
        "links": [
            {
                "linkid": "23",
                "sysmapid": "3",
                "selementid1": "10",
                "selementid2": "11",
                "drawtype": "0",
                "color": "00CC00",
                "label": "",
                "show_label": "1",
                "indicator_type": "0",
                "itemid": "0",
                "linktriggers": [],
                "thresholds": [],
                "highlights": []
            }
```

```
        ],
        "users": [
            {
                "sysmapuserid": "1",
                "userid": "2",
                "permission": "2"
            }
        ],
        "userGroups": [
            {
                "sysmapusrgrpid": "1",
                "usrgrpid": "7",
                "permission": "2"
            }
        ],
        "shapes":[
            {
                "sysmap_shapeid":"1",
                "type":"0",
                "x":"0",
                "y":"0",
                "width":"680",
                "height":"15",
                "text":"{MAP.NAME}",
                "font":"9",
                "font_size":"11",
                "font_color":"000000",
                "text_halign":"0",
                "text_valign":"0",
                "border_type":"0",
                "border_width":"0",
                "border_color":"000000",
                "background_color":"",
                "zindex":"0"
            }
        ],
        "lines":[
            {
                "sysmap_shapeid":"2",
                "x1": 30,
                "y1": 10,
                "x2": 100,
                "y2": 50,
                "line_type": 1,
                "line_width": 10,
                "line_color": "009900",
                "zindex":"1"
            }
        ],
        "sysmapid": "3",
        "name": "Local network",
        "width": "400",
        "height": "400",
        "backgroundid": "0",
        "background_scale": "1",
        "label_type": "2",
        "label_location": "3",
        "show_element_label": "0",
        "show_link_label": "1",
        "highlight": "1",
        "expandproblem": "1",
        "markelements": "0",
```

```
            "show_unack": "0",
            "grid_size": "50",
            "grid_show": "1",
            "grid_align": "1",
            "label_format": "0",
            "label_type_host": "2",
            "label_type_hostgroup": "2",
            "label_type_trigger": "2",
            "label_type_map": "2",
            "label_type_image": "2",
            "label_string_host": "",
            "label_string_hostgroup": "",
            "label_string_trigger": "",
            "label_string_map": "",
            "label_string_image": "",
            "iconmapid": "0",
            "expand_macros": "0",
            "severity_min": "0",
            "userid": "1",
            "private": "1",
            "show_suppressed": "1"
        }
    ],
    "id": 1
}
```


Ver también

- Mapa de iconos
- Elemento de mapa
- Enlace de mapa
- URL de mapa
- Usuario de mapa
- Grupo de usuarios de mapa
- Formas de mapa
- Líneas de mapa

Fuente

CMap::get() en *ui/include/classes/api/services/CMap.php*.

map.update

Descripción

`object map.update(object/array maps)`

Este método permite actualizar mapas existentes.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object/array)` Propiedades del mapa que se actualizarán.

La propiedad `mapid` debe definirse para cada mapa, todas las demás propiedades son opcionales. Solo se actualizarán las propiedades pasadas, todas las demás permanecerán sin cambios.

Además de las propiedades estándar del mapa, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
links	array	Enlaces del mapa para reemplazar los enlaces existentes.
selements	array	Elementos del mapa para reemplazar los elementos existentes.

Parámetro	Tipo	Descripción
urls	array	URLs del mapa para reemplazar las URLs existentes.
users	array	Usuarios del mapa para reemplazar los elementos existentes.
userGroups	array	Grupos de usuarios del mapa para reemplazar los elementos existentes.
shapes	array	Formas del mapa para reemplazar las formas existentes.
lines	array	Líneas del mapa para reemplazar las líneas existentes.

Note:
Para crear enlaces de mapa entre nuevos elementos del mapa, deberá establecer el `selementid` de un elemento en un valor arbitrario y luego usar este valor para hacer referencia a este elemento en las propiedades `selementid1` o `selementid2` de los enlaces. Cuando se cree el elemento, este valor se reemplazará por el ID correcto generado por Zabbix. Consulte el ejemplo para map.create.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los mapas actualizados bajo la propiedad `sysmapids`.

Ejemplos

Cambiar el tamaño de un mapa

Cambie el tamaño del mapa a 1200x1200 píxeles.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "map.update",
    "params": {
        "sysmapid": "8",
        "width": 1200,
        "height": 1200
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "sysmapids": [
            "8"
        ]
    },
    "id": 1
}
```

Cambiar propietario del mapa

Disponible solo para administradores y superadministradores.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "map.update",
    "params": {
        "sysmapid": "9",
        "userid": "1"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "sysmapids": [
            "9"
        ]
    },
    "id": 1
}
```

Ver también

- Elemento de mapa
- Enlace de mapa
- URL de mapa
- Usuario de mapa
- Grupo de usuarios de mapa
- Formas de mapa
- Líneas de mapa

Fuente

CMap::update() en *ui/include/classes/api/services/CMap.php*.

MFA

Esta clase está diseñada para trabajar con métodos MFA (Autenticación Multifactor).

Referencias de objetos:

- MFA

Métodos disponibles:

- mfa.create - crear nuevos métodos MFA
- mfa.delete - eliminar métodos MFA
- mfa.get - obtener métodos MFA
- mfa.update - actualizar métodos MFA

Objeto MFA

Los siguientes objetos están directamente relacionados con la API `mfa`.

MFA

El objeto MFA (Autenticación Multifactor) tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
mfaid	ID	ID del método MFA. Comportamiento de la propiedad: - *solo lectura* - *requerido* para operaciones de actualización
type	integer	Tipo de método MFA. Valores posibles: 1 - TOTP (Contraseñas de un solo uso basadas en tiempo); 2 - Duo Universal Prompt.
name	string	Nombre único del método MFA. Comportamiento de la propiedad: - *requerido* para operaciones de creación

Propiedad	Tipo	Descripción
hash_function	integer	Tipo de función hash para generar códigos TOTP. Valores posibles: 1 - SHA-1; 2 - SHA-256; 3 - SHA-512. Comportamiento de la propiedad: - *requerido* si `type` está establecido en ”TOTP”
code_length	integer	Longitud del código de verificación. Valores posibles: 6 - 6 dígitos; 8 - 8 dígitos. Comportamiento de la propiedad: - *requerido* si `type` está establecido en ”TOTP”
api_hostname	string	Nombre de host de la API proporcionado por el servicio de autenticación Duo. Comportamiento de la propiedad: - *requerido* si `type` está establecido en ”Duo Universal Prompt”
clientid	string	ID de cliente proporcionado por el servicio de autenticación Duo. Comportamiento de la propiedad: - *requerido* si `type` está establecido en ”Duo Universal Prompt”
client_secret	string	Secreto de cliente proporcionado por el servicio de autenticación Duo. Comportamiento de la propiedad: - *solo escritura* - *requerido* si `type` está establecido en ”Duo Universal Prompt”

mfa.create

Descripción

`object mfa.create(object/array MFA methods)`

Este método permite crear nuevos métodos MFA.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(objeto/array)` Métodos MFA a crear.

El método acepta métodos MFA con las propiedades estándar del método MFA.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los métodos MFA creados bajo la propiedad `mfaids`. El orden de los IDs devueltos coincide con el orden de las métricas pasadas.

Ejemplos

Creación de métodos MFA

Cree un método MFA ”Zabbix TOTP” utilizando contraseñas de un solo uso basadas en tiempo (TOTP), con la función hash para generar los códigos TOTP establecida en SHA-1 y la longitud del código de verificación establecida en 6 dígitos.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "mfa.create",
    "params": {
        "type": 1,
        "name": "Zabbix TOTP",
        "hash_function": 1,
        "code_length": 6
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "mfaids": [
            "1"
        ]
    },
    "id": 1
}
```

Fuente

CMfa::create() en *ui/include/classes/api/services/CMfa.php*.

mfa.delete

Descripción

`object mfa.delete(array mfaids)`

Este método permite eliminar métodos MFA.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración del rol de usuario. Consulte User roles para obtener más información.

Parámetros

`(array)` IDs de los métodos MFA a eliminar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los métodos MFA eliminados bajo la propiedad `mfaids`.

Ejemplos

Eliminando métodos MFA

Elimina un método MFA.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "mfa.delete",
    "params": [
        "2"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
```

```
    "result": {
        "mfaids": [
            "2"
        ]
    },
    "id": 1
}
```

Fuente

CMfa::delete() en *ui/include/classes/api/services/CMfa.php*.

mfa.get

Descripción

`integer/array mfa.get(object parameters)`

El método permite recuperar métodos MFA según los parámetros proporcionados.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración del rol de usuario. Consulte User roles para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
mfaids	ID/array	Devolver solo los métodos MFA con los IDs indicados.
selectUsrgrps	query	Devuelve una propiedad `usrgrps` con los grupos de usuarios asociados a los métodos MFA. Admite `count`.
filter	object	Devuelve solo aquellos resultados que coincidan exactamente con el filtro dado. Acepta un objeto, donde las claves son nombres de propiedades y los valores son un único valor o un array de valores con los que comparar. Admite las propiedades: `mfaid` - ID del método MFA; `type` - Tipo del método MFA.
sortfield	string/array	Ordena el resultado por las propiedades indicadas. Valores posibles: `name`.
search	object	Devuelve resultados que coincidan con el patrón dado (no distingue mayúsculas/minúsculas). Acepta un objeto, donde las claves son nombres de propiedades y los valores son cadenas a buscar. Si no se dan opciones adicionales, esto realizará una búsqueda `LIKE "%…%"`. Propiedades soportadas: `name`.
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
excludeSearch	boolean	
limit	integer	
output	query	
preservekeys	boolean	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	

Parámetro	Tipo	Descripción
startSearch	boolean	

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Buscar métodos MFA por nombre

Recuperar todos los métodos MFA con "Zabbix" en su nombre.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "mfa.get",
    "params": {
        "output": "extend",
        "search": {
            "name": "Zabbix"
        }
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "mfaid": "1",
            "type": "1",
            "name": "Zabbix TOTP 1",
            "hash_function": "1",
            "code_length": "6",
            "api_hostname": "",
            "clientid": ""
        },
        {
            "mfaid": "2",
            "type": "1",
            "name": "Zabbix TOTP 2",
            "hash_function": "3",
            "code_length": "8",
            "api_hostname": "",
            "clientid": ""
        }
    ],
    "id": 1
}
```

Fuente

CMfa::get() en *ui/include/classes/api/services/CMfa.php*.

mfa.update

Descripción

`object mfa.update(object/array MFA methods)`

Este método permite actualizar métodos MFA existentes.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(objeto/array)` Propiedades del método MFA que se van a actualizar.

La propiedad `mfaid` debe definirse para cada elemento, todas las demás propiedades son opcionales. Solo se actualizarán las propiedades pasadas, todas las demás permanecerán sin cambios.

El método acepta métodos MFA con las propiedades estándar del método MFA.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los métodos MFA actualizados bajo la propiedad `mfaids`.

Ejemplos

Actualización de las propiedades del método

Actualiza la función hash para generar códigos TOTP y la longitud del código de verificación para el método MFA "Zabbix TOTP" que utiliza contraseñas de un solo uso basadas en tiempo (TOTP).

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "mfa.update",
    "params": {
        "mfaid": "1",
        "hash_function": 3,
        "code_length": 8
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "mfaids": [
            "1"
        ]
    },
    "id": 1
}
```

Fuente

CMfa::update() en *ui/include/classes/api/services/CMfa.php*.

Métrica de APM

Esta clase está diseñada para trabajar con métricas recopiladas para la monitorización del rendimiento de aplicaciones (APM).

Referencias de objetos:

- Métrica de APM
- Ejemplar

Métodos disponibles:

- apmmetric.get: recuperar puntos de datos de métricas

Objeto de métrica APM

Los siguientes objetos están directamente relacionados con la API `apmmetric`.

> **Note:**
> Cada propiedad, excepto `type`, corresponde a una columna de las tablas de métricas (`otel_metrics_gauge`, `otel_metrics_sum`, `otel_metrics_histogram` y `otel_metrics_exponential_histogram`) de la fuente de datos de telemetría. La mayoría de las propiedades corresponden a la columna con el mismo nombre (por ejemplo, la propiedad `metric_name` corresponde a la columna `MetricName`). Cuando los nombres difieren, las columnas de la fuente de datos se especifican en las tablas siguientes. Para consultar la lista de columnas, consulte Columnas de datos de telemetría compatibles.

Métrica de APM

El objeto de métrica de APM representa un punto de datos de una métrica (el valor, o los valores agregados, de una medición en un momento específico).

El objeto de métrica de APM tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
type	integer	Tipo del punto de datos. Valores posibles: 0 - Gauge; 1 - Sum; 2 - Histograma; 3 - Histograma exponencial. El tipo está determinado por la tabla que contiene el punto de datos: `otel_metrics_gauge`, `otel_metrics_sum`, `otel_metrics_histogram` o `otel_metrics_exponential_histogram`. Comportamiento de la propiedad: - *solo lectura*
resource_attributes	object	Atributos del recurso del punto de datos en formato clave-valor (por ejemplo, `host.name: eabf8c2a5ddc`). Comportamiento de la propiedad: - *solo lectura*
resource_schema_url	string	URL, si existe, del esquema de OpenTelemetry de los atributos del recurso. Comportamiento de la propiedad: - *solo lectura*
scope_name	string	Nombre del ámbito de instrumentación que produjo el punto de datos. Comportamiento de la propiedad: - *solo lectura*
scope_version	string	Versión, si existe, del ámbito de instrumentación. Comportamiento de la propiedad: - *solo lectura*
scope_attributes	object	Atributos del ámbito de instrumentación en formato clave-valor. Comportamiento de la propiedad: - *solo lectura*
scope_schema_url	string	URL, si existe, del esquema de OpenTelemetry del ámbito de instrumentación y sus atributos. Comportamiento de la propiedad: - *solo lectura*

Propiedad	Tipo	Descripción
service_name	string	Nombre del servicio que produjo el punto de datos (por ejemplo, `zabbix-frontend`), tomado del atributo de recurso `service.name` de OpenTelemetry. Comportamiento de la propiedad: - *solo lectura*
metric_name	string	Nombre de la métrica a la que pertenece el punto de datos (por ejemplo, `http.server.request.duration`). Comportamiento de la propiedad: - *solo lectura*
metric_description	string	Descripción, si existe, de la métrica. Comportamiento de la propiedad: - *solo lectura*
metric_unit	string	Unidad, si existe, de los valores de la métrica (por ejemplo, `s` o `By`; consulte UCUM). Comportamiento de la propiedad: - *solo lectura*
attributes	object	Atributos del punto de datos en formato clave-valor (por ejemplo, `http.request.method: POST`). Comportamiento de la propiedad: - *solo lectura*
start_time_unix	integer	Hora Unix (en nanosegundos) en la que comenzó el período de medición del punto de datos. Comportamiento de la propiedad: - *solo lectura*
time_unix	integer	Hora Unix (en nanosegundos) del punto de datos. Para los puntos de datos de tipo *Sum*, *Histograma* e *Histograma exponencial*, corresponde al final del período de medición. Comportamiento de la propiedad: - *solo lectura*
value	float	Valor del punto de datos. Se aplica a los puntos de datos de tipo *Gauge* y *Sum*. Predeterminado: 0. Comportamiento de la propiedad: - *solo lectura*
flags	integer	Máscara de bits de los indicadores del punto de datos. Si el bit menos significativo está establecido (valor 1), no se registró ningún valor para el punto de datos. Comportamiento de la propiedad: - *solo lectura*
exemplars	array	Matriz de objetos exemplar, es decir, mediciones individuales conservadas como ejemplos, junto con el trace y el span durante los que se registraron. Comportamiento de la propiedad: - *solo lectura*

Propiedad	Tipo	Descripción
aggregation_temporality	integer	Forma en que los valores del punto de datos se agregan a lo largo del tiempo. Se aplica a los puntos de datos de tipo *Sum*, *Histograma* e *Histograma exponencial*. Valores posibles: 0 - *(predeterminado)* No especificado; 1 - Delta (agregado durante el período transcurrido desde el punto de datos anterior); 2 - Acumulativo (agregado durante el período transcurrido desde `start_time_unix`). Comportamiento de la propiedad: - *solo lectura*
is_monotonic	boolean	Indica si la suma solo puede aumentar o permanecer igual con el tiempo. Se aplica a los puntos de datos de tipo *Sum*. Predeterminado: `false`. Comportamiento de la propiedad: - *solo lectura*
count	integer	Número de mediciones del punto de datos. Se aplica a los puntos de datos de tipo *Histograma* e *Histograma exponencial*. Predeterminado: 0. Comportamiento de la propiedad: - *solo lectura*
sum	float	Suma de todas las mediciones del punto de datos. Se aplica a los puntos de datos de tipo *Histograma* e *Histograma exponencial*. Predeterminado: 0. Comportamiento de la propiedad: - *solo lectura*
bucket_counts	array	Matriz de enteros con el número de mediciones en cada intervalo del histograma. Los intervalos se definen mediante `explicit_bounds`. Se aplica a los puntos de datos de tipo *Histograma*. Predeterminado: `[]`. Comportamiento de la propiedad: - *solo lectura*
explicit_bounds	array	Matriz de números de coma flotante con los límites entre los intervalos del histograma. Se aplica a los puntos de datos de tipo *Histograma*. Predeterminado: `[]`. Comportamiento de la propiedad: - *solo lectura*

Propiedad	Tipo	Descripción
min	float	Menor de las mediciones del punto de datos. Se aplica a los puntos de datos de tipo *Histograma* e *Histograma exponencial*. Predeterminado: 0. Comportamiento de la propiedad: - *solo lectura*
max	float	Mayor de las mediciones del punto de datos. Se aplica a los puntos de datos de tipo *Histograma* e *Histograma exponencial*. Predeterminado: 0. Comportamiento de la propiedad: - *solo lectura*
scale	integer	Escala del histograma exponencial, que determina los límites y la resolución de los intervalos. Una escala mayor proporciona intervalos más estrechos y una resolución más precisa; una escala menor proporciona intervalos más amplios y una resolución más aproximada. Se aplica a los puntos de datos de tipo *Histograma exponencial*. Predeterminado: 0. Comportamiento de la propiedad: - *solo lectura*
zero_count	integer	Número de mediciones en el intervalo cero del histograma exponencial. Se aplica a los puntos de datos de tipo *Histograma exponencial*. Predeterminado: 0. Comportamiento de la propiedad: - *solo lectura*
positive_bucket_counts	array	Matriz de enteros con el número de mediciones positivas en cada intervalo del histograma exponencial. Los límites de los intervalos se determinan mediante `scale`; el índice del primer intervalo se establece en `positive_offset`. Se aplica a los puntos de datos de tipo *Histograma exponencial*. Predeterminado: `[]`. Comportamiento de la propiedad: - *solo lectura*
positive_offset	integer	Índice del primer intervalo en `positive_bucket_counts`. Se aplica a los puntos de datos de tipo *Histograma exponencial*. Predeterminado: 0. Comportamiento de la propiedad: - *solo lectura*
negative_bucket_counts	array	Matriz de enteros con el número de mediciones negativas en cada intervalo del histograma exponencial. Los límites de los intervalos se determinan mediante `scale`; el índice del primer intervalo se establece en `negative_offset`. Se aplica a los puntos de datos de tipo *Histograma exponencial*. Predeterminado: `[]`. Comportamiento de la propiedad: - *solo lectura*

Propiedad	Tipo	Descripción
negative_offset	integer	Índice del primer intervalo en `negative_bucket_counts`. Se aplica a los puntos de datos de tipo *Histograma exponencial*. Predeterminado: 0. Comportamiento de la propiedad: - *solo lectura*

Ejemplar

El objeto ejemplar tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
traceid	string	ID, si existe, del trace durante el cual se registró la medición. Columna de origen de datos: `Exemplars.TraceId`.
spanid	string	ID, si existe, del span durante el cual se registró la medición. Columna de origen de datos: `Exemplars.SpanId`.
value	float	Valor de la medición. Columna de origen de datos: `Exemplars.Value`.
time_unix	integer	Hora Unix (en nanosegundos) en la que se registró la medición. Columna de origen de datos: `Exemplars.TimeUnix`.
filtered_attributes	object	Atributos clave-valor de la medición que no están incluidos en los `attributes` del punto de datos. Columna de origen de datos: `Exemplars.FilteredAttributes`.

apmmetric.get

Descripción

`integer/array apmmetric.get(object parameters)`

El método permite recuperar puntos de datos de métricas según los parámetros indicados.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object)` Parámetros que definen el resultado deseado.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
time_from	timestamp	Devuelve únicamente los puntos de datos cuyo tiempo sea igual o posterior al tiempo indicado. Comportamiento del parámetro: - *obligatorio*
time_till	timestamp	Devuelve únicamente los puntos de datos cuyo tiempo sea anterior al tiempo indicado. Comportamiento del parámetro: - *obligatorio*
types	integer/array	Devuelve únicamente los puntos de datos de los tipos indicados. Consulte la página del objeto de métrica de APM para obtener una lista de los tipos de métricas compatibles.

Parámetro	Tipo	Descripción
with_flags_on	integer	Devuelve únicamente los puntos de datos que tengan establecidos todos los indicadores indicados. Para especificar varios indicadores, utilice la suma de sus valores.
with_flags_off	integer	Devuelve únicamente los puntos de datos que no tengan establecido ninguno de los indicadores indicados. Para especificar varios indicadores, utilice la suma de sus valores.
resource_attributes	object/array	Devuelve únicamente los puntos de datos con los atributos de recurso indicados. Formato: `[{"key": "<key>", "value": "<value>", "operator": "<operator>"}, ...]`. Una matriz vacía devuelve todos los puntos de datos. Valores posibles de operator: 0 - *(predeterminado)* Contiene; 1 - Es igual a; 2 - No contiene; 3 - No es igual a; 4 - Existe; 5 - No existe.
resource_attributes_evaltype	integer	Método de evaluación de las condiciones de `resource_attributes`. Valores posibles: 0 - *(predeterminado)* Y/O; 2 - O.
scope_attributes	object/array	Devuelve únicamente los puntos de datos con los atributos de ámbito indicados. Formato: `[{"key": "<key>", "value": "<value>", "operator": "<operator>"}, ...]`. Una matriz vacía devuelve todos los puntos de datos. Valores posibles de operator: 0 - *(predeterminado)* Contiene; 1 - Es igual a; 2 - No contiene; 3 - No es igual a; 4 - Existe; 5 - No existe.
scope_attributes_evaltype	integer	Método de evaluación de las condiciones de `scope_attributes`. Valores posibles: 0 - *(predeterminado)* Y/O; 2 - O.
attributes	object/array	Devuelve únicamente los puntos de datos con los atributos indicados. Formato: `[{"key": "<key>", "value": "<value>", "operator": "<operator>"}, ...]`. Una matriz vacía devuelve todos los puntos de datos. Valores posibles de operator: 0 - *(predeterminado)* Contiene; 1 - Es igual a; 2 - No contiene; 3 - No es igual a; 4 - Existe; 5 - No existe.
attributes_evaltype	integer	Método de evaluación de las condiciones de `attributes`. Valores posibles: 0 - *(predeterminado)* Y/O; 2 - O.

Parámetro	Tipo	Descripción
filter	object	Devuelve únicamente los resultados que coincidan exactamente con el filtro indicado. Acepta un objeto cuyas claves son nombres de propiedades y cuyos valores son un único valor o una matriz de valores con los que se debe comparar. Propiedades compatibles: `resource_schema_url`, `scope_schema_url`, `scope_name`, `scope_version`, `service_name`, `metric_name`, `metric_unit`, `flags`.
search	object	Devuelve los resultados que coincidan con el patrón indicado (sin distinguir entre mayúsculas y minúsculas). Acepta un objeto cuyas claves son nombres de propiedades y cuyos valores son cadenas que se deben buscar. Si no se proporcionan opciones adicionales, se realizará una búsqueda `LIKE "%…%"`. Propiedades compatibles: `resource_schema_url`, `scope_schema_url`, `scope_name`, `scope_version`, `service_name`, `metric_name`, `metric_description`, `metric_unit`.
sortfield	string/array	Ordena el resultado según las propiedades indicadas. Valores posibles: `type`, `resource_schema_url`, `scope_schema_url`, `scope_name`, `scope_version`, `service_name`, `metric_name`, `metric_unit`, `start_time_unix`, `time_unix`.
offset	integer	Número de registros que se omitirán antes de devolver el resultado.
countOutput	boolean	Estos parámetros se describen en los comentarios de referencia.
excludeSearch	boolean	
limit	integer	
output	query	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores devueltos

`(integer/array)` Devuelve uno de los siguientes valores:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperación de puntos de datos de histogramas

Recupere el punto de datos más reciente de la métrica de histograma `http.server.request.duration` del servicio `zabbix-frontend`.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "apmmetric.get",
    "params": {
        "output": ["metric_name", "metric_unit", "time_unix", "count", "sum", "min", "max", "explicit_boun
        "time_from": 1790848800,
        "time_till": 1790852400,
        "types": 2,
        "filter": {
            "service_name": "zabbix-frontend",
            "metric_name": "http.server.request.duration"
```

```
        },
        "sortfield": "time_unix",
        "sortorder": "DESC",
        "limit": 1
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "metric_name": "http.server.request.duration",
            "metric_unit": "s",
            "time_unix": "1790852390000000000",
            "count": "1842",
            "sum": "412.773",
            "min": "0.0061",
            "max": "1.098",
            "explicit_bounds": ["0.005", "0.05", "0.1", "0.25", "0.5", "1"],
            "bucket_counts": ["0", "211", "904", "512", "167", "41", "7"],
            "exemplars": [
                {
                    "traceid": "dee625f33cb0994e86c5cd4640e3b09c",
                    "spanid": "3f8b2c6d1e9a7054",
                    "value": "1.098",
                    "time_unix": "1790850513439147623",
                    "filtered_attributes": {}
                }
            ]
        }
    ],
    "id": 1
}
```

Véase también

- Traza de APM
- Span de APM
- Registro de APM

Origen

CApmMetric::get() en *ui/include/classes/api/services/CApmMetric.php*.

Módulo

Esta clase está diseñada para trabajar con módulos de la interfaz web.

Referencias de objetos:

- Module

Métodos disponibles:

- module.create - instalar nuevos módulos
- module.delete - desinstalar módulos
- module.get - recuperar módulos
- module.update - actualizar módulos

Objeto módulo

Los siguientes objetos están directamente relacionados con la API `module`.

Módulo

El objeto module tiene las siguientes propiedades.

Property	Type	Description
moduleid	ID	ID del módulo tal como se almacena en la base de datos. Comportamiento de la propiedad: - *solo lectura* - *obligatorio* para operaciones de actualización
id	string	ID único del módulo, definido por un desarrollador en el archivo `manifest.json` del módulo. Valores posibles para módulos integrados: consulte la descripción de la propiedad "type" en Dashboard widget. Comportamiento de la propiedad: - *obligatorio* para operaciones de creación
relative_path	string	Ruta al directorio del módulo relativa al directorio del frontend de Zabbix. Valores posibles: `widgets/*` - para módulos de widgets integrados; `modules/*` - para módulos de terceros. Comportamiento de la propiedad: - *obligatorio* para operaciones de creación
status	integer	Indica si el módulo está habilitado o deshabilitado. Valores posibles: 0 - *(predeterminado)* Deshabilitado; 1 - Habilitado.
config	object	Configuración del módulo.

module.create

Descripción

`object module.create(object/array modules)`

Este método permite instalar nuevos módulos del frontend.

Note:
Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.
 Si `$ZBX_FEATURE_FLAGS['modules_config_enabled']` está establecido en `false` en el archivo de configuración del frontend, el usuario no podrá llamar a este método.

Attention:
Los archivos de los módulos deben descomprimirse manualmente en los subdirectorios correctos, de acuerdo con la propiedad `relative_path` de los módulos.

Parámetros

`(objeto/array)` Módulos a crear.

El método acepta módulos con las propiedades estándar del módulo.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los módulos instalados bajo la propiedad `moduleids`. El orden de los IDs devueltos coincide con el orden de los módulos pasados.

Ejemplos

Instalación de un módulo

Instale un módulo con el estado ”Habilitado”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "module.create",
    "params": {
        "id": "example_module",
        "relative_path": "modules/example_module",
        "status": 1
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "moduleids": [
            "25"
        ]
    },
    "id": 1
}
```

Véase también

- Módulo
- Módulos de frontend

Fuente

CModule::create() en *ui/include/classes/api/services/CModule.php*.

module.delete

Descripción

`object module.delete(array moduleids)`

Este método permite desinstalar módulos.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.
 Si `$ZBX_FEATURE_FLAGS['modules_config_enabled']` está establecido en `false` en el archivo de configuración del frontend, el usuario no podrá llamar a este método.

> **Attention:**
> Los archivos de los módulos deben eliminarse manualmente.

Parámetros

`(array)` IDs de los módulos a desinstalar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los módulos desinstalados bajo la propiedad `moduleids`.

Ejemplos

Desinstalando múltiples módulos

Desinstalar los módulos ”27” y ”28”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "module.delete",
    "params": [
        "27",
        "28"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "moduleids": [
            "27",
            "28"
        ]
    },
    "id": 1
}
```

Fuente

CModule::delete() en *ui/include/classes/api/services/CModule.php*.

module.get

Descripción

`integer/array module.get(object parameters)`

El método permite recuperar módulos según los parámetros especificados.

Note:
Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulta Roles de usuario para obtener más información.
 Si `$ZBX_FEATURE_FLAGS['modules_config_enabled']` está establecido en `false` en el archivo de configuración del frontend, el usuario no podrá llamar a este método.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
moduleids	ID/array	Devolver solo los módulos con los IDs dados.
sortfield	string/array	Ordenar el resultado por las propiedades dadas. Valores posibles: `moduleid`, `relative_path`.
countOutput	boolean	Estos parámetros se describen en el Comentario de referencia.
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperar un módulo por ID

Recuperar todos los datos sobre los módulos "1", "2" y "25".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "module.get",
    "params": {
        "output": "extend",
        "moduleids": [
            "1",
            "2",
            "25"
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "moduleid": "1",
            "id": "actionlog",
            "relative_path": "widgets/actionlog",
            "status": "1",
            "config": []
        },
        {
            "moduleid": "2",
            "id": "clock",
            "relative_path": "widgets/clock",
            "status": "1",
            "config": []
        },
        {
            "moduleid": "25",
            "id": "example",
            "relative_path": "modules/example_module",
            "status": "1",
            "config": []
        }
    ],
    "id": 1
}
```

Véase también

- Módulo
- Widget de panel de control
- Módulos de interfaz web

Origen

CModule::get() en *ui/include/classes/api/services/CModule.php*.

module.update

Descripción

`object module.update(object/array modules)`

Este método permite actualizar módulos existentes.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulta Roles de usuario para obtener más información.
 Si `$ZBX_FEATURE_FLAGS['modules_config_enabled']` está establecido en `false` en el archivo de configuración del frontend, no se permitirá al usuario llamar a este método.

Parámetros

`(objeto/array)` Propiedades del módulo que se van a actualizar.

La propiedad `moduleid` debe definirse para cada módulo, todas las demás propiedades son opcionales. Solo se actualizarán las propiedades especificadas.

El método acepta módulos con las propiedades estándar del módulo.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los módulos actualizados bajo la propiedad `moduleids`.

Ejemplos

Deshabilitar un módulo

Deshabilitar el módulo "25".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "module.update",
    "params": {
        "moduleid": "25",
        "status": 0
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "moduleids": [
            "25"
        ]
    },
    "id": 1
}
```

Véase también

- Módulo
- Módulos de frontend

Fuente

CModule::update() en *ui/include/classes/api/services/CModule.php*.

Nodo de alta disponibilidad

Esta clase está diseñada para trabajar con nodos de servidor que forman parte de un clúster de alta disponibilidad o una instancia de servidor independiente.

Referencias de objetos:

- Nodo de alta disponibilidad

Métodos disponibles:

- hanode.get - recuperar nodos

Objeto de nodo de alta disponibilidad

El siguiente objeto está relacionado con la operación de un clúster de alta disponibilidad de servidores Zabbix.

Nodo de alta disponibilidad

> **Note:**
> Los nodos son creados por el server de Zabbix y no pueden modificarse mediante la API.

El objeto Nodo de alta disponibilidad tiene las siguientes propiedades.

Property	Type	Description
ha_nodeid	ID	ID del nodo.
name	string	Nombre asignado al nodo, usando la entrada de configuración HANodeName de `zabbix_server.conf`. Vacío para un server que se ejecuta en modo independiente.
address	string	Dirección IP o nombre DNS desde el que se conecta el nodo.
port	integer	Puerto en el que se está ejecutando el nodo.
lastaccess	integer	Hora del latido, es decir, la hora de la última actualización recibida del nodo. Marca de tiempo UTC.
status	integer	Estado del nodo. Valores posibles: 0 - en espera; 1 - detenido manualmente; 2 - no disponible; 3 - activo.

hanode.get

Descripción

`integer/array hanode.get(object parameters)`

El método permite recuperar una lista de nodos del clúster de alta disponibilidad según los parámetros proporcionados.

> **Note:**
> Este método solo está disponible para tipos de usuario *Super admin*. Consulte Roles de usuario para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
ha_nodeids	ID/array	Devolver solo los nodos con los IDs de nodo dados.
filter	object	Devolver solo aquellos resultados que coincidan exactamente con el filtro dado. Acepta un objeto, donde las claves son nombres de propiedades y los valores son un único valor o un array de valores con los que comparar. Admite las propiedades: `name`, `address`, `status`.

Parámetro	Tipo	Descripción
sortfield	string/array	Ordenar el resultado por las propiedades dadas. Valores posibles: `name`, `lastaccess`, `status`.
countOutput	flag	Estos parámetros se describen en el comentario de referencia.
limit	integer	
output	query	
preservekeys	boolean	
sortorder	string/array	

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Obtener una lista de nodos ordenados por estado

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "hanode.get",
    "params": {
        "preservekeys": true,
        "sortfield": "status",
        "sortorder": "DESC"
    },
    "id": 1
}
```

Respuesta:

```
{
  "jsonrpc": "2.0",
  "result": {
    "ckuo7i1nw000h0sajj3l3hh8u": {
      "ha_nodeid": "ckuo7i1nw000h0sajj3l3hh8u",
      "name": "node-active",
      "address": "192.168.1.13",
      "port": "10051",
      "lastaccess": "1635335704",
      "status": "3"
    },
    "ckuo7i1nw000e0sajwfttc1mp": {
      "ha_nodeid": "ckuo7i1nw000e0sajwfttc1mp",
      "name": "node6",
      "address": "192.168.1.10",
      "port": "10053",
      "lastaccess": "1635332902",
      "status": "2"
    },
    "ckuo7i1nv000c0sajz85xcrtt": {
      "ha_nodeid": "ckuo7i1nv000c0sajz85xcrtt",
      "name": "node4",
      "address": "192.168.1.8",
      "port": "10052",
      "lastaccess": "1635334214",
      "status": "1"
    },
    "ckuo7i1nv000a0saj1fcdkeu4": {
      "ha_nodeid": "ckuo7i1nv000a0saj1fcdkeu4",
```

```
      "name": "node2",
      "address": "192.168.1.6",
      "port": "10051",
      "lastaccess": "1635335705",
      "status": "0"
    }
  },
  "id": 1
}
```

Obtener una lista de nodos específicos por sus IDs

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "hanode.get",
    "params": {
        "ha_nodeids": ["ckuo7i1nw000e0sajwfttc1mp", "ckuo7i1nv000c0sajz85xcrtt"]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "ha_nodeid": "ckuo7i1nv000c0sajz85xcrtt",
            "name": "node4",
            "address": "192.168.1.8",
            "port": "10052",
            "lastaccess": "1635334214",
            "status": "1"
        },
        {
            "ha_nodeid": "ckuo7i1nw000e0sajwfttc1mp",
            "name": "node6",
            "address": "192.168.1.10",
            "port": "10053",
            "lastaccess": "1635332902",
            "status": "2"
        }
    ],
    "id": 1
}
```

Obtener una lista de nodos detenidos

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "hanode.get",
    "params": {
        "output": ["ha_nodeid", "address", "port"],
        "filter": {
            "status": 1
        }
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "ha_nodeid": "ckuo7i1nw000g0sajjsjre7e3",
            "address": "192.168.1.12",
            "port": "10051"
        },
        {
            "ha_nodeid": "ckuo7i1nv000c0sajz85xcrtt",
            "address": "192.168.1.8",
            "port": "10052"
        },
        {
            "ha_nodeid": "ckuo7i1nv000d0sajd95y1b6x",
            "address": "192.168.1.9",
            "port": "10053"
        }
    ],
    "id": 1
}
```


Obtener un recuento de nodos en espera

Solicitud:


```
{
    "jsonrpc": "2.0",
    "method": "hanode.get",
    "params": {
        "countOutput": true,
        "filter": {
            "status": 0
        }
    },
    "id": 1
}
```


Respuesta:


```
{
    "jsonrpc": "2.0",
    "result": "3",
    "id": 1
}
```


Comprobar el estado de los nodos en direcciones IP específicas

Solicitud:


```
{
    "jsonrpc": "2.0",
    "method": "hanode.get",
    "params": {
        "output": ["name", "status"],
        "filter": {
            "address": ["192.168.1.7", "192.168.1.13"]
        }
    },
    "id": 1
}
```


Respuesta:


```
{
    "jsonrpc": "2.0",
    "result": [
```

```
        {
            "name": "node3",
            "status": "0"
        },
        {
            "name": "node-active",
            "status": "3"
        }
    ],
    "id": 1
}
```

Fuente

CHaNode::get() en *ui/include/classes/api/services/CHaNode.php*.

Panel de control de template

Esta clase está diseñada para trabajar con paneles de control de template.

Referencias de objetos:

- Panel de control de template
- Página de panel de control de template
  - Widget de panel de control de template
    - Campo de widget de panel de control de template

Métodos disponibles:

- templatedashboard.create - crear nuevos paneles de control de template
- templatedashboard.delete - eliminar paneles de control de template
- templatedashboard.get - recuperar paneles de control de template
- templatedashboard.update - actualizar paneles de control de template

Objeto de panel de plantilla

Los siguientes objetos están directamente relacionados con la API de `templatedashboard`.

Panel de control de template

El objeto del panel de control de template tiene las siguientes propiedades.

Property	Type	Description
dashboardid	ID	ID del panel de control de template. Property behavior: - *read-only* - *required* for update operations
name	string	Nombre del panel de control de template. Property behavior: - *required* for create operations
templateid	ID	ID del template al que pertenece el panel de control. Property behavior: - *constant* - *required* for create operations
display_period	integer	Intervalo predeterminado de la presentación de diapositivas (en segundos). Possible values: 10, 30, 60, 120, 600, 1800, 3600. Default: 30.

Property	Type	Description
auto_start	integer	Inicio automático de la presentación de diapositivas. Possible values: 0 - no iniciar automáticamente la presentación de diapositivas; 1 - *(default)* iniciar automáticamente la presentación de diapositivas.
uuid	string	Identificador único universal, utilizado para vincular los paneles de control de template importados con los ya existentes. Se genera automáticamente si no se proporciona.

Página de dashboard de template

El objeto de página de dashboard de template tiene las siguientes propiedades.

Property	Type	Description
dashboard_pageid	ID	ID de la página de dashboard. Property behavior: - *read-only*
name	string	Nombre de la página de dashboard. Default: empty string.
display_period	integer	Intervalo de la presentación de diapositivas del dashboard (en segundos). Possible values: 0, 10, 30, 60, 120, 600, 1800, 3600. Default: 0 (will use the default slideshow interval).
widgets	array	Array de los objetos template dashboard widget.

Widget de dashboard de template

El objeto widget de dashboard de template tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
widgetid	ID	ID del widget de dashboard. Comportamiento de la propiedad: - *solo lectura*

Propiedad	Tipo	Descripción
type	string	Tipo de widget de dashboard. Valores posibles: actionlog - Registro de acciones; clock - Reloj; discovery - Estado del descubrimiento; favgraphs - Gráficas favoritas; favmaps - Mapas favoritos; gauge - Medidor; graph - Gráfica (clásica); graphprototype - Prototipo de gráfica; honeycomb - Panal; hostavail - Disponibilidad de host; hostcard - Tarjeta de host; hostnavigator - Navegador de host; itemcard - Tarjeta de item; itemnavigator - Navegador de item; item - Valor de item; map - Mapa; navtree - Árbol de navegación de mapas; piechart - Gráfica de pastel; plaintext - Texto plano; problemhosts - Hosts con problemas; problems - Problemas; problemsbysv - Problemas por severidad; scatterplot - Diagrama de dispersión; slareport - Informe de SLA; svggraph - Gráfica; systeminfo - Información del sistema; tophosts - Hosts principales; topitems - Items principales; toptriggers - Triggers principales; trigover - Vista general de triggers; url - URL; web - Monitorización web. Comportamiento de la propiedad: - *requerido*
name	string	Nombre personalizado del widget.
x	integer	Posición horizontal desde el lado izquierdo del dashboard. Los valores posibles van de 0 a 71.
y	integer	Posición vertical desde la parte superior del dashboard. Los valores posibles van de 0 a 63.
width	integer	El ancho del widget. Los valores posibles van de 1 a 72.
height	integer	La altura del widget. Los valores posibles van de 1 a 64.
view_mode	integer	El modo de visualización del widget. Valores posibles: 0 - *(por defecto)* vista por defecto del widget; 1 - con cabecera oculta;
fields	array	Array de objetos campo de widget de dashboard de template.

Campo de widget de dashboard de template

El objeto de campo de widget de dashboard de template tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
type	integer	Tipo de campo del widget. Valores posibles: 0 - Entero; 1 - Cadena; 4 - Item; 5 - Prototipo de item; 6 - Gráfico; 7 - Prototipo de gráfico; 8 - Mapa; 9 - Servicio; 10 - SLA; 11 - Usuario; 12 - Acción; 13 - Tipo de medio. Comportamiento de la propiedad: - *requerido*
name	string	Nombre del campo del widget. Valores posibles: ver Campos de widget de dashboard. Tenga en cuenta que algunos parámetros relacionados con el host (por ejemplo, *Grupos de hosts*, *Excluir grupos de hosts* y *Hosts* en el widget *Problemas*, *Grupos de hosts* en el widget *Disponibilidad de host*, etc.) no están disponibles al configurar el widget en un dashboard de template. Esto se debe a que los dashboards de template muestran datos solo del host al que está vinculado el template. Comportamiento de la propiedad: - *requerido*
value	mixed	Valor del campo del widget dependiendo del tipo. Valores posibles: ver Campos de widget de dashboard. Tenga en cuenta que algunos parámetros relacionados con el host (por ejemplo, *Grupos de hosts*, *Excluir grupos de hosts* y *Hosts* en el widget *Problemas*, *Grupos de hosts* en el widget *Disponibilidad de host*, etc.) no están disponibles al configurar el widget en un dashboard de template. Esto se debe a que los dashboards de template muestran datos solo del host al que está vinculado el template. Comportamiento de la propiedad: - *requerido*

templatedashboard.create

Descripción

`object templatedashboard.create(object/array templateDashboards)`

Este método permite crear nuevos paneles de control de template.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object/array)` Dashboards de template a crear.

Además de las propiedades estándar de dashboard de template, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
pages	array	Páginas de dashboard de template que se crearán para el dashboard. Las páginas del dashboard se ordenarán en el mismo orden que se especifique. Comportamiento del parámetro: - *requerido*

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los paneles de plantilla creados bajo la propiedad `dashboardids`. El orden de los IDs devueltos coincide con el orden de los paneles de plantilla pasados.

Ejemplos

Creación de un panel de plantilla

Cree un panel de plantilla llamado “Gráficas” con un widget de Gráfica en una sola página del panel.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "templatedashboard.create",
    "params": {
        "templateid": "10318",
        "name": "Graphs",
        "pages": [
            {
                "widgets": [
                    {
                        "type": "graph",
                        "x": 0,
                        "y": 0,
                        "width": 12,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 6,
                                "name": "graphid",
                                "value": "1123"
                            }
                        ]
                    }
                ]

            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "32"
        ]
    },
    "id": 1
}
```

Ver también

- Página de dashboard de template
- Widget de dashboard de template
- Campo de widget de dashboard de template

Fuente

CTemplateDashboard::create() en *ui/include/classes/api/services/CTemplateDashboard.php*.

templatedashboard.delete

Descripción

`object templatedashboard.delete(array templateDashboardIds)`

Este método le permite eliminar dashboards de template.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(array)` IDs de los paneles de plantilla a eliminar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los paneles de plantilla eliminados bajo la propiedad `dashboardids`.

Ejemplos

Eliminando varios paneles de plantilla

Eliminar dos paneles de plantilla.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "templatedashboard.delete",
    "params": [
        "45",
        "46"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "45",
            "46"
        ]
    },
    "id": 1
}
```

Fuente

CTemplateDashboard::delete() en *ui/include/classes/api/services/CTemplateDashboard.php*.

templatedashboard.get

Descripción

`integer/array templatedashboard.get(object parameters)`

El método permite recuperar dashboards de template según los parámetros proporcionados.

Note:
Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
dashboardids	ID/array	Devolver solo los paneles de template con los IDs dados.
templateids	ID/array	Devolver solo los paneles de template que pertenecen a los templates dados.
selectPages	query	Devolver una propiedad `pages` con las páginas del panel de template, correctamente ordenadas.
sortfield	string/array	Ordenar el resultado por las propiedades dadas. Valores posibles: `dashboardid`, `name`.
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperar paneles de plantilla

Recupera todos los paneles de plantilla con widgets para una plantilla especificada.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "templatedashboard.get",
    "params": {
        "output": "extend",
        "selectPages": "extend",
        "templateids": "10001"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "dashboardid": "23",
            "name": "Docker overview",
```

```
            "templateid": "10001",
            "display_period": "30",
            "auto_start": "1",
            "uuid": "6dfcbe0bc5ad400ea9c1c2dd7649282f",
            "pages": [
                {
                    "dashboard_pageid": "1",
                    "name": "",
                    "display_period": "0",
                    "widgets": [
                        {
                            "widgetid": "220",
                            "type": "graph",
                            "name": "",
                            "x": "0",
                            "y": "0",
                            "width": "36",
                            "height": "5",
                            "view_mode": "0",
                            "fields": [
                                {
                                    "type": "6",
                                    "name": "graphid",
                                    "value": "1125"
                                }
                            ]
                        },
                        {
                            "widgetid": "221",
                            "type": "graph",
                            "name": "",
                            "x": "12",
                            "y": "0",
                            "width": "36",
                            "height": "5",
                            "view_mode": "0",
                            "fields": [
                                {
                                    "type": "6",
                                    "name": "graphid",
                                    "value": "1129"
                                }
                            ]
                        },
                        {
                            "widgetid": "222",
                            "type": "graph",
                            "name": "",
                            "x": "0",
                            "y": "5",
                            "width": "36",
                            "height": "5",
                            "view_mode": "0",
                            "fields": [
                                {
                                    "type": "6",
                                    "name": "graphid",
                                    "value": "1128"
                                }
                            ]
                        },
                        {
```

```
                            "widgetid": "223",
                            "type": "graph",
                            "name": "",
                            "x": "12",
                            "y": "5",
                            "width": "36",
                            "height": "5",
                            "view_mode": "0",
                            "fields": [
                                {
                                    "type": "6",
                                    "name": "graphid",
                                    "value": "1126"
                                }
                            ]
                        },
                        {
                            "widgetid": "224",
                            "type": "graph",
                            "name": "",
                            "x": "0",
                            "y": "10",
                            "width": "36",
                            "height": "5",
                            "view_mode": "0",
                            "fields": [
                                {
                                    "type": "6",
                                    "name": "graphid",
                                    "value": "1127"
                                }
                            ]
                        }
                    ]
                }
            ]
        }
    ],
    "id": 1
}
```


Ver también

- Página de dashboard de template
- Widget de dashboard de template
- Campo de widget de dashboard de template

Fuente

CTemplateDashboard::get() en *ui/include/classes/api/services/CTemplateDashboard.php*.

templatedashboard.update

Descripción

`object templatedashboard.update(object/array templateDashboards)`

Este método permite actualizar los dashboards de template existentes.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración del rol de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

(object/array) Propiedades del dashboard de template que se actualizarán.

La propiedad dashboardid debe especificarse para cada dashboard, todas las demás propiedades son opcionales. Sólo se actualizarán las propiedades especificadas.

Además de las propiedades estándar del dashboard de template, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
pages	array	Páginas del dashboard de template para reemplazar las páginas existentes del dashboard. Las páginas del dashboard se actualizan por la propiedad dashboard_pageid. Se crearán nuevas páginas del dashboard para los objetos sin la propiedad dashboard_pageid y las páginas existentes del dashboard se eliminarán si no se reutilizan. Las páginas del dashboard se ordenarán en el mismo orden en que se especifiquen. Sólo se actualizarán las propiedades especificadas de las páginas del dashboard. Se requiere al menos un objeto de página de dashboard para la propiedad pages.

Valores de retorno

(object) Devuelve un objeto que contiene los IDs de los tableros de plantilla actualizados bajo la propiedad dashboardids.

Ejemplos

Renombrar un panel de plantilla

Renombrar un panel de plantilla a ”Gráficas de rendimiento”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "templatedashboard.update",
    "params": {
        "dashboardid": "23",
        "name": "Gráficas de rendimiento"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "23"
        ]
    },
    "id": 1
}
```

Actualización de las páginas del panel de la plantilla

Cambie el nombre de la primera página del panel, reemplace los widgets en la segunda página del panel y agregue una nueva página como la tercera. Elimine todas las demás páginas del panel.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "templatedashboard.update",
    "params": {
        "dashboardid": "2",
        "pages": [
            {
```

```
                "dashboard_pageid": 1,
                "name": "Página renombrada"
            },
            {
                "dashboard_pageid": 2,
                "widgets": [
                    {
                        "type": "clock",
                        "x": 0,
                        "y": 0,
                        "width": 12,
                        "height": 3
                    }
                ]
            },
            {
                "display_period": 60
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "2"
        ]
    },
    "id": 1
}
```

Ver también

- Widget de dashboard de template
- Campo de widget de dashboard de template

Fuente

CTemplateDashboard::update() en *ui/include/classes/api/services/CTemplateDashboard.php*.

Problem

Esta clase está diseñada para trabajar con problemas.

Referencias de objetos:

- Problem
  - Media type URL
- Problem tag

Métodos disponibles:

- problem.get - recuperar problemas

Objeto problema

Los siguientes objetos están directamente relacionados con la API de `problem`.

Problema

> **Note:**
> Los problemas son creados por el servidor Zabbix y no pueden ser modificados a través de la API.

El objeto problema tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
eventid	ID	ID del evento de problema.
source	integer	Tipo de evento de problema. Valores posibles: 0 - evento creado por un trigger; 3 - evento interno; 4 - evento creado al actualizar el estado de un servicio.
object	integer	Tipo de objeto relacionado con el evento de problema. Valores posibles si `source` está establecido en ”evento creado por un trigger”: 0 - trigger. Valores posibles si `source` está establecido en ”evento interno”: 0 - trigger; 4 - item; 5 - regla LLD. Valores posibles si `source` está establecido en ”evento creado al actualizar el estado de un servicio”: 6 - servicio.
objectid	ID	ID del objeto relacionado.
clock	timestamp	Hora en la que se creó el evento de problema.
ns	integer	Nanosegundos en los que se creó el evento de problema.
r_eventid	ID	ID del evento de recuperación.
r_clock	timestamp	Hora en la que se creó el evento de recuperación.
r_ns	integer	Nanosegundos en los que se creó el evento de recuperación.
cause_eventid	ID	ID del evento causa.
correlationid	ID	ID de la regla de correlación si este evento fue recuperado por una regla de correlación global.
userid	ID	ID del usuario que cerró el problema (si el problema fue cerrado manualmente).
name	string	Nombre del problema resuelto.
acknowledged	integer	Estado de reconocimiento del problema. Valores posibles: 0 - no reconocido; 1 - reconocido.
severity	integer	Gravedad actual del problema. Valores posibles: 0 - no clasificado; 1 - información; 2 - advertencia; 3 - media; 4 - alta; 5 - desastre.
suppressed	integer	Indica si el problema está suprimido. Valores posibles: 0 - el problema está en estado normal; 1 - el problema está suprimido.
opdata	string	Datos operativos con macros expandidas.
urls	array	URLs de tipo de medio activas.

URL del tipo de medio

El objeto de URL del tipo de medio tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
name	string	Nombre de la URL definida en el tipo de medio.
url	string	Valor de la URL definida en el tipo de medio.

Los resultados contendrán entradas solo para los tipos de medios activos con la entrada de menú de evento habilitada. La macro utilizada en las propiedades se expandirá, pero si una de las propiedades contiene una macro sin expandir, ambas propiedades se excluirán de los resultados. Para las macros soportadas, consulte *Macros soportadas*.

Etiqueta de problema

El objeto de etiqueta de problema tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
tag	string	Nombre de la etiqueta de problema.
value	string	Valor de la etiqueta de problema.

problem.get

Descripción

`integer/array problem.get(object parameters)`

El método permite recuperar problemas según los parámetros dados.

Este método sirve para recuperar problemas no resueltos. También es posible, si se especifica, recuperar adicionalmente problemas resueltos recientemente. El período que determina cuán antiguo es "recientemente" se define en *Administration* > *General*. Los problemas que se resolvieron antes de ese período no se conservan en la tabla de problemas. Para recuperar problemas que se resolvieron más atrás en el pasado, use el método `event.get`.

Attention:
Este método puede devolver problemas de una entidad eliminada si estos problemas aún no han sido eliminados por el housekeeper.

Note:
Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parameter	Type	Description
eventids	ID/array	Devolver solo problemas con los IDs dados.
groupids	ID/array	Devolver solo problemas creados por objetos que pertenecen a los grupos de hosts dados.
hostids	ID/array	Devolver solo problemas creados por objetos que pertenecen a los hosts dados.
objectids	ID/array	Devolver solo problemas creados por los objetos dados.
source	integer	Devolver solo problemas con el tipo dado. Consulte la página del objeto de evento problem para obtener una lista de los tipos de evento admitidos. Predeterminado: 0 - problem creado por un trigger.

Parameter	Type	Description
object	integer	Devolver solo problemas creados por objetos del tipo dado. Consulte la página del objeto de evento problem para obtener una lista de los tipos de objeto admitidos. Predeterminado: 0 - trigger.
acknowledged	boolean	Si se establece en `true`, devolver solo problemas reconocidos.
action	integer	Devolver solo problemas para los que se hayan realizado las acciones de actualización de evento dadas. Para varias acciones, use una suma de cualquier valor de mapa de bits aceptable como máscara de bits (por ejemplo, 34 para reconocer y suprimir el evento).
action_userids	ID/array	Devolver solo problemas con los IDs dados de los usuarios que realizaron las acciones de actualización del evento problem.
suppressed	boolean	Si se establece en `true`, devolver solo problemas suprimidos.
symptom	boolean	Si se establece en `true`, devolver solo eventos problem de síntoma.
severities	integer/array	Devolver solo problemas con las severidades de evento dadas. Se aplica solo si object es trigger.
evaltype	integer	Método de evaluación de etiquetas. Valores posibles: 0 - *(predeterminado)* And/Or; 2 - Or.
tags	array	Devolver solo problemas con las etiquetas dadas. Formato: `[{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]`. Un array vacío devuelve todos los problemas. Valores posibles de operator: 0 - *(predeterminado)* Contains; 1 - Equals; 2 - Does not contain; 3 - Does not equal; 4 - Exists; 5 - Does not exist.
recent	boolean	Si se establece en `true`, devolver problemas recientes, incluidos los resueltos dentro de `ok_period` y cualquier problema no resuelto. Si se establece en `false`, devolver solo problemas no resueltos.
eventid_from	string	Devolver solo problemas con IDs mayores o iguales que el ID dado.
eventid_till	string	Devolver solo problemas con IDs menores o iguales que el ID dado.
time_from	timestamp	Devolver solo problemas que se hayan creado después o en el momento dado.
time_till	timestamp	Devolver solo problemas que se hayan creado antes o en el momento dado.

Parameter	Type	Description
selectAcknowledges	query	Devolver una propiedad `acknowledges` con las actualizaciones del problem. Las actualizaciones del problem se ordenan en orden cronológico inverso. El objeto de actualización del problem tiene las siguientes propiedades: `acknowledgeid` - `(ID)` ID de la actualización; `userid` - `(ID)` ID del usuario que actualizó el evento; `eventid` - `(ID)` ID del evento actualizado; `clock` - `(timestamp)` hora en que se actualizó el evento; `message` - `(string)` texto del mensaje; `action` - `(integer)`tipo de acción de actualización (consulte `event.acknowledge`); `old_severity` - `(integer)` severidad del evento antes de esta acción de actualización; `new_severity` - `(integer)` severidad del evento después de esta acción de actualización; `suppress_until` - `(timestamp)` hora hasta la que se suprimirá el evento; `taskid` - `(ID)` ID de la tarea si el evento actual está experimentando un cambio de rango; `maintenanceid` - `(ID)` ID del mantenimiento. Admite `count`.
selectTags	query	Devolver una propiedad `tags` con las etiquetas del problem. Formato de salida: `[{"tag": "<tag>", "value": "<value>"}, ...]`.
selectSuppressionData	query	Devolver una propiedad `suppression_data` con la lista de mantenimientos activos y supresiones manuales: `maintenanceid` - `(ID)` ID del mantenimiento; `userid` - `(ID)` ID del usuario que suprimió el problem; `suppress_until` - `(integer)` tiempo hasta el que el problem está suprimido.
filter	object	Devolver solo aquellos resultados que coincidan exactamente con el filtro dado. Acepta un objeto, donde las claves son nombres de propiedades y los valores son un único valor o un array de valores con los que comparar. No admite propiedades de tipo de datos data type `text`.
sortfield	string/array	Ordenar el resultado por las propiedades dadas. Valores posibles: `eventid`.
countOutput	boolean	Estos parámetros se describen en la comentario de referencia.
editable	boolean	
excludeSearch	boolean	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

`(integer/array)` Devuelve cualquiera de los siguientes:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperar eventos de problema de trigger

Recupera eventos recientes del trigger "15112."

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "problem.get",
    "params": {
        "output": "extend",
        "selectAcknowledges": "extend",
        "selectTags": "extend",
        "selectSuppressionData": "extend",
        "objectids": "15112",
        "recent": true,
        "sortfield": ["eventid"],
        "sortorder": "DESC"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "eventid": "1245463",
            "source": "0",
            "object": "0",
            "objectid": "15112",
            "clock": "1472457242",
            "ns": "209442442",
            "r_eventid": "1245468",
            "r_clock": "1472457285",
            "r_ns": "125644870",
            "correlationid": "0",
            "userid": "1",
            "name": "Zabbix agent on localhost is unreachable for 5 minutes",
            "acknowledged": "1",
            "severity": "3",
            "cause_eventid": "0",
            "suppressed": "1",
            "acknowledges": [
                {
                    "acknowledgeid": "14443",
                    "userid": "1",
                    "eventid": "1245463",
                    "clock": "1472457281",
                    "message": "problem solved",
                    "action": "6",
                    "old_severity": "0",
                    "new_severity": "0",
                    "suppress_until": "1472511600",
                    "taskid": "0",
                    "maintenanceid": "0"
                }
            ],
            "opdata": "",
            "suppression_data": [
                {
                    "maintenanceid": "15",
                    "suppress_until": "1472511600",
                    "userid": "0"
                }
            ],
```

```
            "tags": [
                {
                    "tag": "test-tag",
                    "value": "test-value"
                }
            ],
            "urls": []
        }
    ],
    "id": 1
}
```

Recuperar problemas reconocidos por un usuario especificado

Recuperar problemas reconocidos por el usuario con ID=10

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "problem.get",
    "params": {
        "output": "extend",
        "action": 2,
        "action_userids": [10],
        "selectAcknowledges": ["userid", "action"],
        "sortfield": ["eventid"],
        "sortorder": "DESC"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "eventid": "1248566",
            "source": "0",
            "object": "0",
            "objectid": "15142",
            "clock": "1472457242",
            "ns": "209442442",
            "r_eventid": "1245468",
            "r_clock": "1472457285",
            "r_ns": "125644870",
            "correlationid": "0",
            "userid": "10",
            "name": "Zabbix agent on localhost is unreachable for 5 minutes",
            "acknowledged": "1",
            "severity": "3",
            "cause_eventid": "0",
            "suppressed": "0",
            "acknowledges": [
                {
                    "acknowledgeid": "9",
                    "userid": "10",
                    "action": "2"
                }
            ],
            "opdata": "",
            "urls": []
        }
    ],
```

```
    "id": 1
}
```

Ver también

- Alerta
- Item
- Host
- Regla LLD
- Trigger

Fuente

CEvent::get() en *ui/include/classes/api/services/CProblem.php*.

Prototipo de gráfico

Esta clase está diseñada para trabajar con prototipos de gráficos.

Referencias de objetos:

- Prototipo de gráfico

Métodos disponibles:

- graphprototype.create - crear nuevos prototipos de gráficos
- graphprototype.delete - eliminar prototipos de gráficos
- graphprototype.get - recuperar prototipos de gráficos
- graphprototype.update - actualizar prototipos de gráficos

Objeto prototipo de gráfico

Los siguientes objetos están directamente relacionados con la API `graphprototype`.

Prototipo de gráfico

El objeto prototipo de gráfico tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
graphid	ID	ID del prototipo de gráfico. Comportamiento de la propiedad: - *solo lectura* - *requerido* para operaciones de actualización
height	integer	Altura del prototipo de gráfico en píxeles. Comportamiento de la propiedad: - *requerido* para operaciones de creación
name	string	Nombre del prototipo de gráfico. Comportamiento de la propiedad: - *requerido* para operaciones de creación
width	integer	Ancho del prototipo de gráfico en píxeles. Comportamiento de la propiedad: - *requerido* para operaciones de creación
flags	integer	Origen del prototipo de gráfico. Valores posibles: 2 - un prototipo de gráfico; 6 - un prototipo de gráfico descubierto Comportamiento de la propiedad: - *solo lectura*

Propiedad	Tipo	Descripción
graphtype	integer	Tipo de diseño del prototipo de gráfico. Valores posibles: 0 - *(por defecto)* normal; 1 - apilado; 2 - circular; 3 - circular desglosado.
percent_left	float	Percentil izquierdo. Por defecto: 0.
percent_right	float	Percentil derecho. Por defecto: 0.
show_3d	integer	Si se muestran los gráficos circulares y desglosados descubiertos en 3D. Valores posibles: 0 - *(por defecto)* mostrar en 2D; 1 - mostrar en 3D.
show_legend	integer	Si se muestra la leyenda en el gráfico descubierto. Valores posibles: 0 - ocultar; 1 - *(por defecto)* mostrar.
show_work_period	integer	Si se muestra el tiempo de trabajo en el gráfico descubierto. Valores posibles: 0 - ocultar; 1 - *(por defecto)* mostrar.
templateid	ID	ID del prototipo de gráfico de la template padre. Comportamiento de la propiedad: - *solo lectura*
yaxismax	float	El valor máximo fijo para el eje Y.
yaxismin	float	El valor mínimo fijo para el eje Y.
ymax_itemid	ID	ID del item que se utiliza como valor máximo para el eje Y. Si un usuario no tiene acceso al item especificado, el gráfico se representa como si `ymax_type` estuviera configurado como "calculado".
ymax_type	integer	Método de cálculo del valor máximo para el eje Y. Valores posibles: 0 - *(por defecto)* calculado; 1 - fijo; 2 - item.
ymin_itemid	ID	ID del item que se utiliza como valor mínimo para el eje Y. Si un usuario no tiene acceso al item especificado, el gráfico se representa como si `ymin_type` estuviera configurado como "calculado".
ymin_type	integer	Método de cálculo del valor mínimo para el eje Y. Valores posibles: 0 - *(por defecto)* calculado; 1 - fijo; 2 - item.

Propiedad	Tipo	Descripción
discover	integer	Estado de descubrimiento del prototipo de gráfico. Valores posibles: 0 - *(por defecto)* se descubrirán nuevos gráficos; 1 - no se descubrirán nuevos gráficos y los gráficos existentes se marcarán como perdidos.
uuid	string	Identificador único universal, utilizado para vincular prototipos de gráficos importados a los ya existentes. Se genera automáticamente si no se proporciona. Comportamiento de la propiedad: - *soportado* si el prototipo de gráfico pertenece a una template

graphprototype.create

Descripción

`object graphprototype.create(object/array graphPrototypes)`

Este método permite crear nuevos prototipos de gráficos.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(objeto/array)` Prototipos de gráficos a crear.

Además de las propiedades estándar de los prototipos de gráficos, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
gitems	array	Items de gráfico que se crearán para los prototipos de gráficos. Los items de gráfico pueden hacer referencia tanto a items como a prototipos de items, pero debe haber al menos un prototipo de item. Comportamiento del parámetro: - *obligatorio*

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los prototipos de gráficos creados bajo la propiedad `graphids`. El orden de los IDs devueltos coincide con el orden de los prototipos de gráficos enviados.

Ejemplos

Creando un prototipo de gráfico

Cree un prototipo de gráfico con dos métricas.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "graphprototype.create",
    "params": {
        "name": "Uso de espacio en disco {#FSNAME}",
        "width": 900,
        "height": 200,
        "gitems": [
            {
                "itemid": "22828",
                "color": "00AA00"
```

```
            },
            {
                "itemid": "22829",
                "color": "3333FF"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "graphids": [
            "652"
        ]
    },
    "id": 1
}
```

Ver también

- Item de gráfico

Fuente

CGraphPrototype::create() en *ui/include/classes/api/services/CGraphPrototype.php*.

graphprototype.delete

Descripción

`object graphprototype.delete(array graphPrototypeIds)`

Este método permite eliminar prototipos de gráficos.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(array)` IDs de las prototipos de gráficas a eliminar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las prototipos de gráficas eliminadas bajo la propiedad `graphids`.

Ejemplos

Eliminando múltiples prototipos de gráficos

Eliminar dos prototipos de gráficos.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "graphprototype.delete",
    "params": [
        "652",
        "653"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "graphids": [
            "652",
            "653"
        ]
    },
    "id": 1
}
```

Fuente

CGraphPrototype::delete() en *ui/include/classes/api/services/CGraphPrototype.php*.

graphprototype.get

Descripción

`integer/array graphprototype.get(object parameters)`

El método permite recuperar prototipos de gráficos según los parámetros proporcionados.

Note:
Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
discoveryids	ID/array	Devolver solo los prototipos de gráficos que pertenecen a las reglas de descubrimiento dadas.
graphids	ID/array	Devolver solo los prototipos de gráficos con los IDs dados.
groupids	ID/array	Devolver solo los prototipos de gráficos que pertenecen a hosts o templates en los grupos de hosts o grupos de templates dados.
hostids	ID/array	Devolver solo los prototipos de gráficos que pertenecen a los hosts dados.
inherited	boolean	Si se establece en `true`, devolver solo los prototipos de gráficos heredados de un template.
itemids	ID/array	Devolver solo los prototipos de gráficos que contienen los prototipos de item dados.
templated	boolean	Si se establece en `true`, devolver solo los prototipos de gráficos que pertenecen a templates.
templateids	ID/array	Devolver solo los prototipos de gráficos que pertenecen a los templates dados.
selectDiscoveryData	query	Devolver una propiedad `discoveryData` con los datos del objeto de descubrimiento del prototipo de gráfico. El objeto de descubrimiento del prototipo de gráfico vincula un prototipo de gráfico descubierto con un prototipo de gráfico del que se descubrió. Tiene las siguientes propiedades: `parent_graphid` - `(ID)` ID del prototipo de gráfico del que se ha creado el gráfico; `status` - `(int)` estado de descubrimiento del gráfico: 0 - *(por defecto)* el prototipo de gráfico está descubierto, 1 - el prototipo de gráfico ya no está descubierto; `ts_delete` - `(timestamp)` hora en la que se eliminará un prototipo de gráfico que ya no se descubre.
selectDiscoveryRule	query	Devolver una propiedad `discoveryRule` con la regla LLD a la que pertenece el prototipo de gráfico.

Parámetro	Tipo	Descripción
selectDiscoveryRulePrototype	query	Devolver una propiedad `discoveryRulePrototype` con el prototipo de regla LLD principal al que pertenece el prototipo de gráfico.
selectGraphItems	query	Devolver una propiedad `gitems` con los items de gráfico utilizados en el prototipo de gráfico.
selectHostGroups	query	Devolver una propiedad `hostgroups` con los grupos de hosts a los que pertenece el prototipo de gráfico.
selectHosts	query	Devolver una propiedad `hosts` con los hosts a los que pertenece el prototipo de gráfico.
selectItems	query	Devolver una propiedad `items` con los items y prototipos de item utilizados en el prototipo de gráfico.
selectTemplateGroups	query	Devolver una propiedad `templategroups` con los grupos de templates a los que pertenece el prototipo de gráfico.
selectTemplates	query	Devolver una propiedad `templates` con los templates a los que pertenece el prototipo de gráfico.
filter	object	Devolver solo aquellos resultados que coincidan exactamente con el filtro dado. Acepta un objeto, donde las claves son nombres de propiedades y los valores son un único valor o un array de valores con los que comparar. No admite propiedades del tipo de dato `text` data type. Admite propiedades adicionales: `host` - nombre técnico del host al que pertenece el prototipo de gráfico; `hostid` - ID del host al que pertenece el prototipo de gráfico.
sortfield	string/array	Ordenar el resultado por las propiedades dadas. Valores posibles: `graphid`, `name`, `graphtype`, `discovered`.
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
editable	boolean	
excludeSearch	boolean	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

`(entero/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperar prototipos de gráficos de una regla LLD

Recuperar todos los prototipos de gráficos de una regla LLD.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "graphprototype.get",
    "params": {
        "output": "extend",
        "discoveryids": "27426"
    },
```

```
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "graphid": "1017",
            "name": "Disk space usage {#FSNAME}",
            "width": "600",
            "height": "340",
            "yaxismin": "0.0000",
            "yaxismax": "0.0000",
            "templateid": "442",
            "show_work_period": "0",
            "show_triggers": "0",
            "graphtype": "2",
            "show_legend": "1",
            "show_3d": "1",
            "percent_left": "0.0000",
            "percent_right": "0.0000",
            "ymin_type": "0",
            "ymax_type": "0",
            "ymin_itemid": "0",
            "ymax_itemid": "0",
            "flags": "2"
            "discover": "0"
        }
    ],
    "id": 1
}
```

Ver también

- Regla de descubrimiento
- Item de gráfico
- Item
- Host
- Grupo de hosts
- Template
- Grupo de templates

Fuente

CGraphPrototype::get() en *ui/include/classes/api/services/CGraphPrototype.php*.

graphprototype.update

Descripción

`object graphprototype.update(object/array graphPrototypes)`

Este método permite actualizar prototipos de gráficos existentes.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(objeto/array)` Propiedades de la gráfica prototipo que se actualizarán.

La propiedad `graphid` debe definirse para cada gráfica prototipo, todas las demás propiedades son opcionales. Sólo se actualizarán las propiedades pasadas, todas las demás permanecerán sin cambios.

Además de las propiedades estándar de la gráfica prototipo, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
gitems	array	Items de la gráfica para reemplazar los items de la gráfica existentes. Si un item de la gráfica tiene definida la propiedad `gitemid`, se actualizará, de lo contrario se creará un nuevo item de la gráfica.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las prototipos de gráficas actualizadas bajo la propiedad `graphids`.

Ejemplos

Cambiar el tamaño de una plantilla de gráfico

Cambie el tamaño de una plantilla de gráfico a 1100 por 400 píxeles.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "graphprototype.update",
    "params": {
        "graphid": "439",
        "width": 1100,
        "height": 400
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "graphids": [
            "439"
        ]
    },
    "id": 1
}
```

Fuente

CGraphPrototype::update() en *ui/include/classes/api/services/CGraphPrototype.php*.

Prototipo de host

Esta clase está diseñada para trabajar con prototipos de host.

Referencias de objetos:

- Prototipo de host
- Enlace de grupo
- Prototipo de grupo
- Etiqueta de prototipo de host
- Interfaz personalizada
    - Detalles de la interfaz personalizada

Métodos disponibles:

- hostprototype.create - crear nuevos prototipos de host
- hostprototype.delete - eliminar prototipos de host
- hostprototype.get - recuperar prototipos de host
- hostprototype.update - actualizar prototipos de host

Objeto prototipo de equipo

Los siguientes objetos están directamente relacionados con la API `hostprototype`.

Prototipo de host

El objeto prototipo de host tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
hostid	ID	ID del prototipo de host. Comportamiento de la propiedad: - *solo lectura* - *requerido* para operaciones de actualización
host	string	Nombre técnico del prototipo de host. Comportamiento de la propiedad: - *requerido* para operaciones de creación - *solo lectura* para objetos heredados
name	string	Nombre visible del prototipo de host. Por defecto: valor de la propiedad `host`. Comportamiento de la propiedad: - *solo lectura* para objetos heredados
status	integer	Estado del prototipo de host. Valores posibles: 0 - *(por defecto)* host monitorizado; 1 - host no monitorizado.
flags	integer	Origen del prototipo de host. Valores posibles: 2 - un prototipo de host; 6 - un prototipo de host descubierto Comportamiento de la propiedad: - *solo lectura*
inventory_mode	integer	Modo de población del inventario del host. Valores posibles: -1 - *(por defecto)* deshabilitado; 0 - manual; 1 - automático.
templateid	ID	ID del prototipo de host de la template padre. Comportamiento de la propiedad: - *solo lectura*
discover	integer	Estado de descubrimiento del prototipo de host. Valores posibles: 0 - *(por defecto)* se descubrirán nuevos hosts; 1 - no se descubrirán nuevos hosts y los hosts existentes se marcarán como perdidos.
custom_interfaces	integer	Origen de las interfaces personalizadas para los hosts creados por el prototipo de host. Valores posibles: 0 - *(por defecto)* heredar interfaces del host padre; 1 - usar interfaces personalizadas de los prototipos de host. Comportamiento de la propiedad: - *solo lectura* para objetos heredados

Propiedad	Tipo	Descripción
uuid	string	Identificador único universal, utilizado para vincular prototipos de host importados a los ya existentes. Se genera automáticamente si no se proporciona. Comportamiento de la propiedad: - *soportado* si el prototipo de host pertenece a una template

Enlace de grupo

El objeto de enlace de grupo vincula un prototipo de host con un grupo de hosts. Tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
groupid	ID	ID del grupo de hosts. Comportamiento de la propiedad: - *obligatorio*

Prototipo de grupo

El objeto prototipo de grupo define un grupo que se creará para un host descubierto. Tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
group_prototypeid	ID	ID del prototipo de grupo. Comportamiento de la propiedad: - *solo lectura*
name	string	Nombre del prototipo de grupo. Comportamiento de la propiedad: - *requerido* para operaciones de creación

Etiqueta de prototipo de host

El objeto de etiqueta de prototipo de host tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
tag	string	Nombre de la etiqueta del prototipo de host. Comportamiento de la propiedad: - *requerido*
value	string	Valor de la etiqueta del prototipo de host. Comportamiento de la propiedad: - *solo lectura*.
object	integer	Tipo de objeto del que se heredó la etiqueta. Valores posibles: 0 - template. Comportamiento de la propiedad: - *solo lectura*.
objectid	ID	ID del objeto del que se heredó la etiqueta. Comportamiento de la propiedad: - *solo lectura*.

Interfaz personalizada

Las interfaces personalizadas son compatibles si `custom_interfaces` del objeto prototipo de host está configurado en "usar interfaces personalizadas de prototipos de host". El objeto de interfaz personalizada tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
type	integer	Tipo de interfaz. Valores posibles: 1 - Agent; 2 - SNMP; 3 - IPMI; 4 - JMX. Comportamiento de la propiedad: - *requerido*
useip	integer	Si la conexión debe realizarse a través de IP. Valores posibles: 0 - conectar usando el nombre DNS del host; 1 - conectar usando la dirección IP del host. Comportamiento de la propiedad: - *requerido*
ip	string	Dirección IP utilizada por la interfaz. Puede contener macros. Comportamiento de la propiedad: - *requerido* si `useip` está configurado en "conectar usando la dirección IP del host"
dns	string	Nombre DNS utilizado por la interfaz. Puede contener macros. Comportamiento de la propiedad: - *requerido* si `useip` está configurado en "conectar usando el nombre DNS del host"
port	string	Número de puerto utilizado por la interfaz. Puede contener macros de usuario y LLD. Comportamiento de la propiedad: - *requerido*
main	integer	Si la interfaz se utiliza como predeterminada en el host. Solo una interfaz de algún tipo puede establecerse como predeterminada en un host. Valores posibles: 0 - no predeterminada; 1 - predeterminada. Comportamiento de la propiedad: - *requerido*
details	object	Objeto adicional para detalles de la interfaz personalizada. Comportamiento de la propiedad: - *requerido* si `type` está configurado en "SNMP"

Detalles de la interfaz personalizada

El objeto de detalles tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
version	integer	Versión de la interfaz SNMP. Valores posibles: 1 - SNMPv1; 2 - SNMPv2c; 3 - SNMPv3. Comportamiento de la propiedad: - *requerido*
bulk	integer	Si se deben usar solicitudes SNMP bulk. Valores posibles: 0 - no usar solicitudes bulk; 1 - *(por defecto)* - usar solicitudes bulk.
community	string	Comunidad SNMP. Comportamiento de la propiedad: - *requerido* si `version` está configurado como ”SNMPv1” o ”SNMPv2c”
max_repetitions	integer	Valor máximo de repetición para solicitudes SNMP bulk nativas (GetBulkRequest-PDUs). Se usa solo para items `discovery[]` y `walk[]` en SNMPv2 y v3. Por defecto: 10.
securityname	string	Nombre de seguridad SNMPv3. Comportamiento de la propiedad: - *soportado* si `version` está configurado como ”SNMPv3”
securitylevel	integer	Nivel de seguridad SNMPv3. Valores posibles: 0 - *(por defecto)* - noAuthNoPriv; 1 - authNoPriv; 2 - authPriv. Comportamiento de la propiedad: - *soportado* si `version` está configurado como ”SNMPv3”
authpassphrase	string	Frase de autenticación SNMPv3. Comportamiento de la propiedad: - *soportado* si `version` está configurado como ”SNMPv3” y `securitylevel` está configurado como ”authNoPriv” o ”authPriv”
privpassphrase	string	Frase de privacidad SNMPv3. Comportamiento de la propiedad: - *soportado* si `version` está configurado como ”SNMPv3” y `securitylevel` está configurado como ”authPriv”
authprotocol	integer	Protocolo de autenticación SNMPv3. Valores posibles: 0 - *(por defecto)* - MD5; 1 - SHA1; 2 - SHA224; 3 - SHA256; 4 - SHA384; 5 - SHA512. Comportamiento de la propiedad: - *soportado* si `version` está configurado como ”SNMPv3” y `securitylevel` está configurado como ”authNoPriv” o ”authPriv”

Propiedad	Tipo	Descripción
privprotocol	integer	Protocolo de privacidad SNMPv3. Usado solo por interfaces SNMPv3. Valores posibles: 0 - *(por defecto)* - DES; 1 - AES128; 2 - AES192; 3 - AES256; 4 - AES192C; 5 - AES256C. Comportamiento de la propiedad: - *soportado* si `version` está configurado como "SNMPv3" y `securitylevel` está configurado como "authPriv"
contextname	string	Nombre de contexto SNMPv3. Comportamiento de la propiedad: - *soportado* si `version` está configurado como "SNMPv3"

hostprototype.create

Descripción

`object hostprototype.create(object/array hostPrototypes)`

Este método le permite crear nuevos prototipos de host.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object/array)` Prototipos de host a crear.

Además de las propiedades estándar del prototipo de host, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
groupLinks	array	Enlaces de grupo que se crearán para el prototipo de host. Comportamiento del parámetro: - *requerido*
ruleid	ID	ID de la regla LLD a la que pertenece el prototipo de host. Comportamiento del parámetro: - *requerido*
groupPrototypes	array	Prototipos de grupo que se crearán para el prototipo de host.
macros	object/array	Macros de usuario que se crearán para el prototipo de host.
tags	object/array	Etiquetas de prototipo de host.
interfaces	object/array	Interfaces personalizadas del prototipo de host.
templates	object/array	Templates que se vincularán al prototipo de host. Los templates deben tener definida únicamente la propiedad `templateid`.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las plantillas de equipo creadas bajo la propiedad `hostids`. El orden de los IDs devueltos coincide con el orden de las plantillas de equipo pasadas.

Ejemplos

Creando una plantilla de equipo

Cree una plantilla de equipo "{#VM.NAME}" en la regla LLD "23542" con una plantilla de grupo "{#HV.NAME}", par de etiquetas "datacenter": "{#DATACENTER.NAME}" e interfaz SNMPv2 personalizada 127.0.0.1:161 con comunidad {$SNMP_COMMUNITY}. Enlácela al grupo de equipos "2".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "hostprototype.create",
    "params": {
        "host": "{#VM.NAME}",
        "ruleid": "23542",
        "custom_interfaces": "1",
        "groupLinks": [
            {
                "groupid": "2"
            }
        ],
        "groupPrototypes": [
            {
                "name": "{#HV.NAME}"
            }
        ],
        "tags": [
            {
                "tag": "datacenter",
                "value": "{#DATACENTER.NAME}"
            }
        ],
        "interfaces": [
            {
                "main": "1",
                "type": "2",
                "useip": "1",
                "ip": "127.0.0.1",
                "dns": "",
                "port": "161",
                "details": {
                    "version": "2",
                    "bulk": "1",
                    "community": "{$SNMP_COMMUNITY}"
                }
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10103"
        ]
    },
    "id": 1
}
```

Ver también

- Enlace de grupo
- Prototipo de grupo
- Etiqueta de prototipo de host
- Interfaz personalizada

- Macro de usuario

Fuente

CHostPrototype::create() en *ui/include/classes/api/services/CHostPrototype.php*.

hostprototype.delete

Descripción

`object hostprototype.delete(array hostPrototypeIds)`

Este método le permite eliminar prototipos de host.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración del rol de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(array)` IDs de los prototipos de equipo a eliminar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los prototipos de equipo eliminados bajo la propiedad `hostids`.

Ejemplos

Eliminando múltiples prototipos de equipo

Eliminar dos prototipos de equipo.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "hostprototype.delete",
    "params": [
        "10103",
        "10105"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10103",
            "10105"
        ]
    },
    "id": 1
}
```

Fuente

CHostPrototype::delete() en *ui/include/classes/api/services/CHostPrototype.php*.

hostprototype.get

Descripción

`integer/array hostprototype.get(object parameters)`

El método permite recuperar prototipos de host según los parámetros proporcionados.

Note:
Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
hostids	ID/array	Devolver solo los prototipos de host con los IDs dados.
discoveryids	ID/array	Devolver solo los prototipos de host que pertenecen a las reglas LLD dadas.
inherited	boolean	Si se establece en `true`, devuelve solo los items heredados de un template.
selectDiscoveryData	query	Devuelve una propiedad `discoveryData` con los datos del objeto de descubrimiento del prototipo de host. El objeto de descubrimiento del prototipo de host vincula un prototipo de host descubierto con un prototipo de host del que se descubrió. Tiene las siguientes propiedades: `host` - `(string)` ID del host; `parent_hostid` - `(string)` ID del prototipo de host del que se ha creado el prototipo de host; `status` - `(int)` estado de descubrimiento del prototipo de host: 0 - *(por defecto)* el prototipo de host está descubierto, 1 - el prototipo de host ya no está descubierto; `ts_delete` - `(timestamp)` hora en la que se eliminará un prototipo de host que ya no se descubre; `ts_disable` - `(timestamp)` hora en la que se deshabilitará un prototipo de host que ya no se descubre; `disable_source` - `(int)` indicador de si el prototipo de host fue deshabilitado por una regla LLD o manualmente: 0 - *(por defecto)* deshabilitado automáticamente, 1 - deshabilitado por una regla LLD.
selectDiscoveryRule	query	Devuelve una propiedad `discoveryRule` con la regla LLD a la que pertenece el prototipo de host.
selectDiscoveryRulePrototype	query	Devuelve una propiedad `discoveryRulePrototype` con el prototipo de regla LLD principal al que pertenece el prototipo de host.
selectInterfaces	query	Devuelve una propiedad `interfaces` con las interfaces personalizadas del prototipo de host.
selectGroupLinks	query	Devuelve una propiedad `groupLinks` con los enlaces de grupo del prototipo de host.
selectGroupPrototypes	query	Devuelve una propiedad `groupPrototypes` con los prototipos de grupo del prototipo de host.
selectInheritedTags	query	Devuelve una propiedad `inheritedTags` con las etiquetas que están en los templates vinculados.
selectMacros	query	Devuelve una propiedad `macros` con las macros del prototipo de host.
selectParentHost	query	Devuelve una propiedad `parentHost` con el host al que pertenece el prototipo de host.
selectTags	query	Devuelve una propiedad `tags` con las etiquetas del prototipo de host.
selectTemplates	query	Devuelve una propiedad `templates` con los templates vinculados al prototipo de host. Admite `count`.
sortfield	string/array	Ordena el resultado por las propiedades dadas. Valores posibles: `hostid`, `host`, `name`, `status`, `discovered`.
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
editable	boolean	
excludeSearch	boolean	
filter	object	

Parámetro	Tipo	Descripción
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperar prototipos de host de una regla LLD

Recuperar todos los prototipos de host, sus enlaces de grupo, prototipos de grupo y etiquetas de una regla LLD.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "hostprototype.get",
    "params": {
        "output": "extend",
        "selectInterfaces": "extend",
        "selectGroupLinks": "extend",
        "selectGroupPrototypes": "extend",
        "selectTags": "extend",
        "discoveryids": "23554"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "hostid": "10092",
            "host": "{#HV.UUID}",
            "name": "{#HV.UUID}",
            "status": "0",
            "templateid": "0",
            "discover": "0",
            "custom_interfaces": "1",
            "uuid": "051a1469d4d045cbbf818fcc843a352e",
            "flags": "2",
            "inventory_mode": "-1",
            "tags": [
                {
                    "tag": "datacenter",
                    "value": "{#DATACENTER.NAME}"
                },
                {
                    "tag": "instance-type",
                    "value": "{#INSTANCE_TYPE}"
                }
            ],
            "groupLinks": [
```

```
                {
                    "group_prototypeid": "4",
                    "hostid": "10092",
                    "groupid": "7",
                    "templateid": "0"
                }
            ],
            "groupPrototypes": [
                {
                    "group_prototypeid": "7",
                    "hostid": "10092",
                    "name": "{#CLUSTER.NAME}",
                    "templateid": "0"
                }
            ],
            "interfaces": [
                {
                    "main": "1",
                    "type": "2",
                    "useip": "1",
                    "ip": "127.0.0.1",
                    "dns": "",
                    "port": "161",
                    "available": "0",
                    "error": "",
                    "errors_from": "0",
                    "disable_until": "0",
                    "details": {
                        "version": "2",
                        "bulk": "1",
                        "community": "{$SNMP_COMMUNITY}",
                        "max_repetitions": "10"
                    }
                }
            ]
        }
    ],
    "id": 1
}
```


Ver también

- Enlace de grupo
- Prototipo de grupo
- Macro de usuario

Fuente

CHostPrototype::get() en *ui/include/classes/api/services/CHostPrototype.php*.

hostprototype.update

Descripción

`object hostprototype.update(object/array hostPrototypes)`

Este método le permite actualizar prototipos de host existentes.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object/array)` Propiedades de la plantilla de host a actualizar.

La propiedad `hostid` debe estar definida para cada plantilla de host, todas las demás propiedades son opcionales. Sólo se actualizarán las propiedades pasadas, todas las demás permanecerán sin cambios.

Además de las propiedades estándar de la plantilla de host, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
groupLinks	array	Enlaces de grupo para reemplazar los enlaces de grupo actuales en la plantilla de host. Comportamiento del parámetro: - *sólo lectura* para objetos heredados
groupPrototypes	array	Plantillas de grupo para reemplazar las plantillas de grupo existentes en la plantilla de host. Todas las plantillas de grupo que no estén listadas en la solicitud serán eliminadas. Comportamiento del parámetro: - *sólo lectura* para objetos heredados
macros	object/array	Macros de usuario para reemplazar las macros de usuario actuales. Todas las macros que no estén listadas en la solicitud serán eliminadas.
tags	object/array	Etiquetas de la plantilla de host para reemplazar las etiquetas actuales. Todas las etiquetas que no estén listadas en la solicitud serán eliminadas. Comportamiento del parámetro: - *sólo lectura* para objetos heredados
interfaces	object/array	Interfaces personalizadas de la plantilla de host para reemplazar las interfaces actuales. El objeto de interfaz personalizada debe contener todos sus parámetros. Todas las interfaces que no estén listadas en la solicitud serán eliminadas. Comportamiento del parámetro: - *soportado* si `custom_interfaces` del objeto plantilla de host está configurado como "usar interfaces personalizadas de plantillas de host" - *sólo lectura* para objetos heredados
templates	object/array	Templates para reemplazar las plantillas actualmente enlazadas. Las plantillas deben tener sólo la propiedad `templateid` definida.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los prototipos de equipo actualizados bajo la propiedad `hostids`.

Ejemplos

Deshabilitar un prototipo de equipo

Deshabilite un prototipo de equipo, es decir, establezca su estado en "1".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "hostprototype.update",
    "params": {
        "hostid": "10092",
        "status": 1
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10092"
        ]
    },
    "id": 1
}
```

Actualización de etiquetas de prototipo de equipo

Reemplazar las etiquetas del prototipo de equipo por nuevas.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "hostprototype.update",
    "params": {
        "hostid": "10092",
        "tags": [
            {
                "tag": "datacenter",
                "value": "{#DATACENTER.NAME}"
            },
            {
                "tag": "instance-type",
                "value": "{#INSTANCE_TYPE}"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10092"
        ]
    },
    "id": 1
}
```

Actualización de interfaces personalizadas de prototipos de equipo

Reemplazar las interfaces heredadas por interfaces personalizadas de prototipos de equipo.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "hostprototype.update",
    "params": {
        "hostid": "10092",
        "custom_interfaces": "1",
        "interfaces": [
            {
                "main": "1",
                "type": "2",
                "useip": "1",
                "ip": "127.0.0.1",
                "dns": "",
                "port": "161",
```

```
                "details": {
                    "version": "2",
                    "bulk": "1",
                    "community": "{$SNMP_COMMUNITY}"
                }
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10092"
        ]
    },
    "id": 1
}
```

Ver también

- Enlace de grupo
- Prototipo de grupo
- Etiqueta de prototipo de host
- Interfaz personalizada
- Macro de usuario

Fuente

CHostPrototype::update() en *ui/include/classes/api/services/CHostPrototype.php*.

Prototipo de item

Esta clase está diseñada para trabajar con prototipos de item.

Referencias de objetos:

- Prototipo de item
  - Cabecera HTTP
  - Campo de consulta HTTP
- Etiqueta de prototipo de item
- Preprocesamiento de prototipo de item

Métodos disponibles:

- itemprototype.create - crear nuevos prototipos de item
- itemprototype.delete - eliminar prototipos de item
- itemprototype.get - recuperar prototipos de item
- itemprototype.update - actualizar prototipos de item

Objeto prototipo de métrica

Los siguientes objetos están directamente relacionados con la API `itemprototype`.

Prototipo de item

El objeto de prototipo de item tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
itemid	ID	ID del prototipo de item. Comportamiento de la propiedad: - *solo lectura* - *obligatoria* para las operaciones de actualización
delay	string	Intervalo de actualización del prototipo de item. Acepta un número de segundos (por ejemplo, `1`, `60`, `300`), un valor con un sufijo de tiempo (por ejemplo, `30s`, `1m`, `2h`, `1d`) o, de forma opcional, uno o más intervalos personalizados, todos separados por punto y coma. Los intervalos personalizados pueden ser una combinación de intervalos flexibles y de planificación. Acepta macros de usuario y macros LLD. Si se utiliza, el valor debe ser una única macro. No se admiten varias macros ni macros combinadas con texto. Los intervalos flexibles pueden escribirse como dos macros separadas por una barra diagonal (por ejemplo, `{$FLEX_INTERVAL}/{$FLEX_PERIOD}`). Ejemplo: `1h;wd1-5h9-18;{$Macro1}/1-7,00:00-24:00;0/6-7,12:00-24:00;{$Macr` Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”Zabbix agent” (0), ”Simple check” (3), ”Zabbix internal” (5), ”External check” (10), ”Database monitor” (11), ”IPMI agent” (12), ”SSH agent” (13), ”TELNET agent” (14), ”Calculated” (15), ”JMX agent” (16), ”HTTP agent” (19), ”SNMP agent” (20), ”Script” (21), ”Browser” (22), ”Telemetry query” (24), o si `type` está establecido en ”Zabbix agent (active)” (7) y `key_` no contiene ”mqtt.get”
hostid	ID	ID del host al que pertenece el prototipo de item. Comportamiento de la propiedad: - *constante* - *obligatoria* para las operaciones de creación
interfaceid	ID	ID de la interfaz del host del prototipo de item. Comportamiento de la propiedad: - *obligatoria* si el prototipo de item pertenece a un host y `type` está establecido en ”Zabbix agent”, ”IPMI agent”, ”JMX agent”, ”SNMP trap” o ”SNMP agent” - *compatible* si el prototipo de item pertenece a un host y `type` está establecido en ”Simple check”, ”External check”, ”SSH agent”, ”TELNET agent” o ”HTTP agent”
key_	string	Clave del prototipo de item. Comportamiento de la propiedad: - *obligatoria* para las operaciones de creación - *solo lectura* para objetos heredados
name	string	Nombre del prototipo de item. Admite macros de usuario. Comportamiento de la propiedad: - *obligatoria* para las operaciones de creación - *solo lectura* para objetos heredados

Propiedad	Tipo	Descripción
type	integer	Tipo del prototipo de item. Valores posibles: 0 - Zabbix agent; 2 - Zabbix trapper; 3 - Simple check; 5 - Zabbix internal; 7 - Zabbix agent (active); 10 - External check; 11 - Database monitor; 12 - IPMI agent; 13 - SSH agent; 14 - TELNET agent; 15 - Calculated; 16 - JMX agent; 17 - SNMP trap; 18 - Dependent item; 19 - HTTP agent; 20 - SNMP agent; 21 - Script; 22 - Browser; 24 - Telemetry query. Comportamiento de la propiedad: - *obligatoria* para las operaciones de creación - *solo lectura* para objetos heredados
url	string	Cadena de URL. Admite macros LLD, macros de usuario, {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {HOST.PORT}, {ITEM.ID}, {ITEM.KEY}. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados
value_type	integer	Tipo de información del prototipo de item. Valores posibles: 0 - numeric float; 1 - character; 2 - log; 3 - numeric unsigned; 4 - text; 5 - binary; 6 - JSON. Comportamiento de la propiedad: - *obligatoria* para las operaciones de creación - *solo lectura* para objetos heredados
allow_traps	integer	Permite que el valor se complete de forma similar al item trapper. 0 - *(predeterminado)* No aceptar datos entrantes; 1 - Aceptar datos entrantes. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent”

Propiedad	Tipo	Descripción
authtype	integer	Método de autenticación. Valores posibles si `type` está establecido en ”SSH agent”: 0 - *(predeterminado)* contraseña; 1 - clave pública. Valores posibles si `type` está establecido en ”HTTP agent”: 0 - *(predeterminado)* ninguno; 1 - básico; 2 - NTLM; 3 - Kerberos; 4 - Digest. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”SSH agent” o ”HTTP agent” - *solo lectura* para objetos heredados (si `type` está establecido en ”HTTP agent”)
description	string	Descripción del prototipo de item.
flags	integer	Origen del prototipo de item. Valores posibles: 2 - un prototipo de item; 6 - un prototipo de item descubierto Comportamiento de la propiedad: - *solo lectura*
follow_redirects	integer	Seguir las redirecciones de respuesta durante el sondeo de datos. Valores posibles: 0 - No seguir redirecciones; 1 - *(predeterminado)* Seguir redirecciones. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados
headers	array	Matriz de encabezados que se enviarán al realizar una solicitud HTTP. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados
history	string	Unidad de tiempo durante la que se deben conservar los datos del historial. También acepta macros de usuario y macros LLD. Predeterminado: 31d.
http_proxy	string	Cadena de conexión del proxy HTTP(S). Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados
ipmi_sensor	string	Sensor IPMI. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”IPMI agent” y `key_` no está establecido en ”ipmi.get” - *compatible* si `type` está establecido en ”IPMI agent” - *solo lectura* para objetos heredados

Propiedad	Tipo	Descripción
jmx_endpoint	string	Cadena de conexión personalizada del JMX agent. Predeterminado: service:jmx:rmi:///jndi/rmi://{HOST.CONN}:{HOST.PORT}/jmxrmi Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”JMX agent”
logtimefmt	string	Formato de la hora en las entradas del registro. Comportamiento de la propiedad: - *compatible* si `value_type` está establecido en ”log” - *solo lectura* para objetos heredados
master_itemid	ID	ID del item maestro. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”Dependent item” - *solo lectura* para objetos heredados
output_format	integer	Indica si la respuesta debe convertirse a JSON. Valores posibles: 0 - *(predeterminado)* Almacenar sin procesar; 1 - Convertir a JSON. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados
params	string	Parámetros adicionales según el tipo de prototipo de item: - script ejecutado para prototipos de item SSH agent y TELNET agent; - consulta SQL para prototipos de item Database monitor; - fórmula para prototipos de item Calculated; - script para prototipos de item Script y Browser. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”Database monitor”, ”SSH agent”, ”TELNET agent”, ”Calculated”, ”Script” o ”Browser” - *solo lectura* para objetos heredados (si `type` está establecido en ”Script” o ”Browser”)
parameters	object/array	Parámetros adicionales si `type` está establecido en ”Script” o ”Browser”. Matriz de objetos con las propiedades `name` y `value`, donde `name` debe ser único. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”Script” o ”Browser” - *solo lectura* para objetos heredados
password	string	Contraseña para la autenticación. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”JMX agent” y `username` está establecido - *compatible* si `type` está establecido en ”Simple check”, ”SSH agent”, ”TELNET agent”, ”Database monitor” o ”HTTP agent” - *solo lectura* para objetos heredados (si `type` está establecido en ”HTTP agent”)

Propiedad	Tipo	Descripción
post_type	integer	Tipo del cuerpo de datos de publicación almacenado en la propiedad `posts`. Valores posibles: 0 - *(predeterminado)* Datos sin procesar. 2 - Datos JSON. 3 - Datos XML. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados
posts	string	Datos del cuerpo de la solicitud HTTP(S). Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”HTTP agent” y `post_type` está establecido en ”JSON data” o ”XML data” - *compatible* si `type` está establecido en ”HTTP agent” y `post_type` está establecido en ”Raw data” - *solo lectura* para objetos heredados
privatekey	string	Nombre del archivo de clave privada. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”SSH agent” y `authtype` está establecido en ”public key”
publickey	string	Nombre del archivo de clave pública. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”SSH agent” y `authtype` está establecido en ”public key”
query_fields	array	Matriz de campos de consulta que se enviarán al realizar una solicitud HTTP. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados
request_method	integer	Tipo de método de solicitud. Valores posibles: 0 - *(predeterminado)* GET; 1 - POST; 2 - PUT; 3 - HEAD. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados
retrieve_mode	integer	Parte de la respuesta que se debe almacenar. Valores posibles si `request_method` está establecido en ”GET”, ”POST” o ”PUT”: 0 - *(predeterminado)* Cuerpo; 1 - Encabezados; 2 - Se almacenarán tanto el cuerpo como los encabezados. Valores posibles si `request_method` está establecido en ”HEAD”: 1 - Encabezados. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados

Propiedad	Tipo	Descripción
snmp_oid	string	OID de SNMP. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”SNMP agent” - *solo lectura* para objetos heredados
ssl_cert_file	string	Ruta del archivo de clave pública SSL. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados
ssl_key_file	string	Ruta del archivo de clave privada SSL. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados
ssl_key_password	string	Contraseña del archivo de clave SSL. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados
status	integer	Estado del prototipo de item. Valores posibles: 0 - *(predeterminado)* prototipo de item habilitado; 1 - prototipo de item deshabilitado; 3 - prototipo de item no compatible.
status_codes	string	Rangos de códigos de estado HTTP requeridos, separados por comas. También admite macros de usuario o macros LLD como parte de una lista separada por comas. Ejemplo: 200,200-{$M},{$M},200-400 Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados
templateid	ID	ID del prototipo de item del template principal. Comportamiento de la propiedad: - *solo lectura*
timeout	string	Tiempo de espera de la solicitud de sondeo de datos del item. Acepta un número de segundos (por ejemplo, 1, 60, 300) o un valor con un sufijo de tiempo (por ejemplo, 30s, 1m). También acepta macros de usuario y macros LLD. Rango de valores posibles: 1-600s. Predeterminado: ”” - utilizar la configuración del proxy/global. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”Zabbix agent” (0), ”Simple check” (3) y `key_` no comienza por ”`vmware.`” ni ”`icmpping`”, ”Zabbix agent (active)” (7), ”External check” (10), ”Database monitor” (11), ”SSH agent” (13), ”TELNET agent” (14), ”HTTP agent” (19), ”SNMP agent” (20) y `snmp_oid` comienza por ”`walk[`” o ”`get[`”, ”Script” (21), ”Browser” (22) - *solo lectura* para objetos heredados

Propiedad	Tipo	Descripción
trapper_hosts	string	Hosts permitidos. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”Zabbix trapper”, o si `type` está establecido en ”HTTP agent” y `allow_traps` está establecido en ”Allow incoming data”. Si se omite la propiedad, se establece automáticamente en la macro de usuario `{$TRAPPER.ALLOWED_HOSTS}` (valor predeterminado). De forma predeterminada, `{$TRAPPER.ALLOWED_HOSTS}` es `127.0.0.1,::1`. Si la propiedad se establece en una cadena vacía, se denegarán las conexiones entrantes de todos los hosts.
trends	string	Unidad de tiempo durante la que se deben conservar los datos de tendencias. También acepta macros de usuario y macros LLD. Predeterminado: 365d. Comportamiento de la propiedad: - *compatible* si `value_type` está establecido en ”numeric float” o ”numeric unsigned”
units	string	Unidades del valor. Comportamiento de la propiedad: - *compatible* si `value_type` está establecido en ”numeric float” o ”numeric unsigned” - *solo lectura* para objetos heredados
username	string	Nombre de usuario para la autenticación. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”SSH agent” o ”TELNET agent”, o si `type` está establecido en ”JMX agent” y `password` está establecido - *compatible* si `type` está establecido en ”Simple check”, ”Database monitor” o ”HTTP agent” - *solo lectura* para objetos heredados (si `type` está establecido en ”HTTP agent”)
uuid	string	Identificador único universal, utilizado para vincular prototipos de item importados con los que ya existen. Se genera automáticamente si no se proporciona. Comportamiento de la propiedad: - *compatible* si el prototipo de item pertenece a un template
valuemapid	ID	ID del mapa de valores asociado. Comportamiento de la propiedad: - *compatible* si `value_type` está establecido en ”numeric float”, ”character” o ”numeric unsigned” - *solo lectura* para objetos heredados
verify_host	integer	Indica si se debe validar que el nombre del host de la conexión coincida con el del certificado del host. Valores posibles: 0 - *(predeterminado)* No validar; 1 - Validar. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados

Propiedad	Tipo	Descripción
verify_peer	integer	Indica si se debe validar que el certificado del host sea auténtico. Valores posibles: 0 - *(predeterminado)* No validar; 1 - Validar. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados
discover	integer	Estado de descubrimiento del prototipo de item. Valores posibles: 0 - *(predeterminado)* se descubrirán los items nuevos; 1 - no se descubrirán items nuevos y los items existentes se marcarán como perdidos.
query	object	Configuración de la consulta de telemetría. Define qué consulta el item en el origen de datos de telemetría, cómo filtra los datos y cómo agrega el resultado. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”Telemetry query” - *solo lectura* para objetos heredados o descubiertos
time_shift	string	Hasta qué momento anterior a la hora actual termina el intervalo de tiempo que consulta el item. Acepta un número de segundos (por ejemplo, `1`, `60`, `300`) o un valor con un sufijo de tiempo (por ejemplo, `30s`, `1m`, `2h`, `1d`). También acepta macros de usuario y macros LLD. Rango de valores posibles: 0-86400s. Predeterminado: 15s. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”Telemetry query”
lookback_limit	string	Hasta qué momento anterior a la hora actual consulta el item los datos cuando no existe una consulta anterior desde la que continuar (por ejemplo, en la primera ejecución del item), o cuando la consulta anterior es más antigua que `delay` (por ejemplo, porque el Zabbix server estaba inactivo). Acepta un número de segundos (por ejemplo, `1`, `60`, `300`) o un valor con un sufijo de tiempo (por ejemplo, `30s`, `1m`, `2h`, `1d`). También acepta macros de usuario y macros LLD. Rango de valores posibles: 1-259200s. Predeterminado: 10m. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”Telemetry query”

Propiedad	Tipo	Descripción
granularity	string	Duración de cada segmento en el que se divide el intervalo de tiempo completo de la consulta. Acepta un número de segundos (por ejemplo, 1, 60, 300) o un valor con un sufijo de tiempo (por ejemplo, 30s, 1m, 2h, 1d). También acepta macros de usuario y macros LLD. Rango de valores posibles: 1-86400s. Predeterminado: 15s. Comportamiento de la propiedad: - *compatible* si type está establecido en "Telemetry query"

Encabezado HTTP

El objeto header tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
name	string	Nombre del encabezado HTTP. Comportamiento de la propiedad: - *obligatorio*
value	string	Valor del encabezado. Comportamiento de la propiedad: - *obligatorio*

Campo de consulta HTTP

El objeto de campo de consulta define un nombre y un valor que se utilizan para especificar un parámetro de URL. Tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
name	string	Nombre del parámetro. Comportamiento de la propiedad: - *obligatorio*
value	string	Valor del parámetro. Comportamiento de la propiedad: - *obligatorio*

Consulta de telemetría

El objeto de consulta de telemetría tiene las siguientes propiedades.

Property	Type	Description
signal_type	integer	Tipo de señal que consulta el item. Valores posibles: 0 - Trazas; 1 - Métricas; 2 - Registros. Comportamiento de la propiedad: - *required*
metric_point_type	integer	Tipo de punto de métrica que consulta el item. Valores posibles: 0 - Suma; 1 - Medidor; 2 - Histograma; 3 - Histograma exponencial. Comportamiento de la propiedad: - *required* si `signal_type` está establecido en ”Métricas”
columns	array	Matriz de columnas de datos de telemetría que consulta el item. Puede estar vacía. Comportamiento de la propiedad: - *required*
aggregated_columns	array	Matriz de columnas de datos de telemetría agregadas que consulta el item y de cómo calcula un valor a partir de los registros de las columnas. Comportamiento de la propiedad: - *required*
filter	object	Filtro de consulta de telemetría para los registros de columnas de la consulta. Comportamiento de la propiedad: - *required*

Columnas de datos de telemetría

El objeto de columnas de datos de telemetría tiene las siguientes propiedades.

Property	Type	Description
column	string	Nombre de la columna de datos de telemetría que consulta el item. Consulte las columnas de datos de telemetría compatibles. Para varias columnas de valor único, cada valor de `column` debe ser único. Para varias columnas que contienen pares clave-valor, cada par de `column` y `attribute_key` debe ser único.
attribute_key	string	Nombre de la clave que consulta el item dentro de la columna que contiene pares clave-valor. Para varias columnas que contienen pares clave-valor, cada par de `column` y `attribute_key` debe ser único. Comportamiento de la propiedad: - *obligatorio* si `column` está establecido en una columna que contiene pares clave-valor

Columnas agregadas de datos de telemetría

El objeto de columnas agregadas de datos de telemetría tiene las siguientes propiedades.

Property	Type	Description
column	string	Nombre de la columna de datos de telemetría que consulta el item. Consulte las columnas de datos de telemetría compatibles. Para varias columnas, cada valor de `column` debe ser único. Esta propiedad no es compatible si `function` está establecido en "Count". Comportamiento de la propiedad: - *obligatoria* si `function` está establecido en "Min", "Max", "Average", "Sum" o "Percentile"
function	integer	Función que el item aplica a los registros de la columna. Valores posibles: 1 - Min; 2 - Max; 3 - Average; 4 - Count; 5 - Sum; 8 - Percentile. Comportamiento de la propiedad: - *obligatoria*
parameters	array	Percentil que el item calcula a partir de los registros de la columna. Valor posible: un número del 0 al 100, proporcionado como una cadena. Comportamiento de la propiedad: - *obligatoria* si `function` está establecido en "Percentile"
alias	string	Nombre único del valor calculado, utilizado para identificarlo en el valor del item. Cada valor de `alias` debe ser único. Comportamiento de la propiedad: - *obligatoria*

Filtro de datos de telemetría

El objeto de filtro de datos de telemetría tiene las siguientes propiedades.

Property	Type	Description
conditions	array	Condiciones del filtro de datos de telemetría que debe cumplir un registro de una columna de datos de telemetría. Comportamiento de la propiedad: - *required*
evaltype	integer	Método de evaluación de la condición del filtro. Valores posibles: 0 - And/Or; 1 - And; 2 - Or; 3 - Expresión personalizada. Comportamiento de la propiedad: - *required*

Property	Type	Description
eval_formula	string	Expresión generada que se utilizará para evaluar las condiciones del filtro. La expresión contiene identificadores que hacen referencia a condiciones de filtro específicas mediante su `formulaid`. El valor de `eval_formula` es igual al valor de `formula` para los filtros con una expresión personalizada. Comportamiento de la propiedad: - *read-only*
formula	string	Expresión definida por el usuario que se utilizará para evaluar las condiciones de los filtros con una expresión personalizada. La expresión debe contener identificadores que hagan referencia a condiciones de filtro específicas mediante su `formulaid`. Los identificadores utilizados en la expresión deben coincidir exactamente con los definidos en las condiciones del filtro: no puede quedar ninguna condición sin utilizar ni omitirse ninguna. Comportamiento de la propiedad: - *required* si `evaltype` está establecido en ”Expresión personalizada”

Condiciones de filtrado de datos de telemetría

El objeto de condiciones de filtrado de datos de telemetría tiene las siguientes propiedades.

Property	Type	Description
column	string	Nombre de la columna de datos de telemetría a la que se aplica la condición de filtrado. Consulte las columnas de datos de telemetría compatibles. Comportamiento de la propiedad: - *obligatoria*
attribute_key	string	Nombre de la clave a la que se aplica la condición, dentro de la columna que contiene pares clave-valor. Comportamiento de la propiedad: - *obligatoria* si `column` está configurada como una columna que contiene pares clave-valor
operator	integer	Operador de la condición de filtrado. Valores posibles para columnas de valor único: 0 - Es igual; 1 - No es igual; 2 - Contiene; 3 - No contiene. Valores posibles para columnas que contienen pares clave-valor: 0 - Es igual; 1 - No es igual; 12 - Existe. Comportamiento de la propiedad: - *obligatoria*
value	string	Valor con el que se debe comparar. Comportamiento de la propiedad: - *obligatoria* si `operator` está configurado como ”Es igual”, ”No es igual”, ”Contiene” o ”No contiene”

Property	Type	Description
formulaid	string	ID único arbitrario que se utiliza para hacer referencia a la condición desde una expresión personalizada. Solo puede contener letras mayúsculas. El ID debe ser definido por el usuario al modificar las condiciones de filtrado, pero se generará nuevamente al solicitarlas posteriormente. Comportamiento de la propiedad: - *obligatoria* si `evaltype` del objeto Filtro de consulta de telemetría está configurado como "Expresión personalizada"

Etiqueta de prototipo de item

El objeto de etiqueta de prototipo de item tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
tag	string	Nombre de la etiqueta del prototipo de item. Comportamiento de la propiedad: - *obligatorio*
value	string	Valor de la etiqueta del prototipo de item. Comportamiento de la propiedad: - *solo lectura*.
object	integer	Tipo de objeto del que se heredó la etiqueta. Valores posibles: 0 - template; 1 - host. Comportamiento de la propiedad: - *solo lectura*.
objectid	ID	ID del objeto del que se heredó la etiqueta. Comportamiento de la propiedad: - *solo lectura*.

Preprocesamiento del prototipo de item

El objeto de preprocesamiento del prototipo de item tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
type	integer	El tipo de opción de preprocesamiento. Valores posibles: 1 - Multiplicador personalizado; 2 - Recorte derecho; 3 - Recorte izquierdo; 4 - Recorte; 5 - Expresión regular; 6 - Booleano a decimal; 7 - Octal a decimal; 8 - Hexadecimal a decimal; 9 - Cambio simple; 10 - Cambio por segundo; 11 - XPath XML; 12 - JSONPath; 13 - En el rango; 14 - Coincide con la expresión regular; 15 - No coincide con la expresión regular; 16 - Comprobar errores en JSON; 17 - Comprobar errores en XML; 18 - Comprobar errores mediante una expresión regular; 19 - Descartar sin cambios; 20 - Descartar sin cambios con latido; 21 - JavaScript; 22 - Patrón de Prometheus; 23 - Prometheus a JSON; 24 - CSV a JSON; 25 - Reemplazar; 26 - Comprobar no compatible; 27 - XML a JSON; 28 - Valor de recorrido SNMP; 29 - Recorrido SNMP a JSON; 30 - Obtener valor SNMP. Comportamiento de la propiedad: - *required*
params	string	Parámetros adicionales utilizados por la opción de preprocesamiento. Los parámetros múltiples se separan mediante el carácter de nueva línea (\n). Si `type` está establecido en ”Comprobar no compatible”, los parámetros siguen la sintaxis `<scope>[\n<pattern>]`, donde *pattern* es una expresión regular y *scope* es uno de los siguientes: -1 - coincide con cualquier error; 0 - comprobar si el mensaje de error coincide con *pattern*; 1 - comprobar si el mensaje de error no coincide con *pattern*. Comportamiento de la propiedad: - *required* si `type` está establecido en ”Multiplicador personalizado” (1), ”Recorte derecho” (2), ”Recorte izquierdo” (3), ”Recorte” (4), ”Expresión regular” (5), ”XPath XML” (11), ”JSONPath” (12), ”En el rango” (13), ”Coincide con la expresión regular” (14), ”No coincide con la expresión regular” (15), ”Comprobar errores en JSON” (16), ”Comprobar errores en XML” (17), ”Comprobar errores mediante una expresión regular” (18), ”Descartar sin cambios con latido” (20), ”JavaScript” (21), ”Patrón de Prometheus” (22), ”Prometheus a JSON” (23), ”CSV a JSON” (24), ”Reemplazar” (25), ”Comprobar no compatible” (26), ”Valor de recorrido SNMP” (28), ”Recorrido SNMP a JSON” (29) o ”Obtener valor SNMP” (30)

Propiedad	Tipo	Descripción
error_handler	integer	Tipo de acción utilizado en caso de fallo del paso de preprocesamiento. Valores posibles: 0 - El mensaje de error lo establece Zabbix server; 1 - Descartar el valor; 2 - Establecer un valor personalizado; 3 - Establecer un mensaje de error personalizado. Valores posibles si `type` está establecido en ”Comprobar no compatible”: 1 - Descartar el valor; 2 - Establecer un valor personalizado; 3 - Establecer un mensaje de error personalizado. Comportamiento de la propiedad: - *required* si `type` está establecido en ”Multiplicador personalizado” (1), ”Expresión regular” (5), ”Booleano a decimal” (6), ”Octal a decimal” (7), ”Hexadecimal a decimal” (8), ”Cambio simple” (9), ”Cambio por segundo” (10), ”XPath XML” (11), ”JSONPath” (12), ”En el rango” (13), ”Coincide con la expresión regular” (14), ”No coincide con la expresión regular” (15), ”Comprobar errores en JSON” (16), ”Comprobar errores en XML” (17), ”Comprobar errores mediante una expresión regular” (18), ”Patrón de Prometheus” (22), ”Prometheus a JSON” (23), ”CSV a JSON” (24), ”Comprobar no compatible” (26), ”XML a JSON” (27), ”Valor de recorrido SNMP” (28), ”Recorrido SNMP a JSON” (29) u ”Obtener valor SNMP” (30)
error_handler_params	string	Parámetros del gestor de errores. Comportamiento de la propiedad: - *required* si `error_handler` está establecido en ”Establecer un valor personalizado” o ”Establecer un mensaje de error personalizado”

Los siguientes parámetros y gestores de errores son compatibles con cada tipo de preprocesamiento.

Tipo de preprocesamiento	Nombre	Parámetro 1	Parámetro 2	Parámetro 3	Gestores de errores compatibles
1	Multiplicador personalizado	número^1, 6^			0, 1, 2, 3
2	Recorte derecho	lista de caracteres[2]			
3	Recorte izquierdo	lista de caracteres[2]			
4	Recorte	lista de caracteres[2]			
5	Expresión regular	patrón[3]	salida[2]		0, 1, 2, 3
6	Booleano a decimal				0, 1, 2, 3
7	Octal a decimal				0, 1, 2, 3

Tipo de preprocesamiento	Nombre	Parámetro 1	Parámetro 2	Parámetro 3	Gestores de errores compatibles
8	Hexadecimal a decimal				0, 1, 2, 3
9	Cambio simple				0, 1, 2, 3
10	Cambio por segundo				0, 1, 2, 3
11	XPath XML	ruta[4]			0, 1, 2, 3
12	JSONPath	ruta[4]			0, 1, 2, 3
13	En el rango	mín.^1, 6^	máx.^1, 6^		0, 1, 2, 3
14	Coincide con la expresión regular	patrón[3]			0, 1, 2, 3
15	No coincide con la expresión regular	patrón[3]			0, 1, 2, 3
16	Comprobar errores en JSON	ruta[4]			0, 1, 2, 3
17	Comprobar errores en XML	ruta[4]			0, 1, 2, 3
18	Comprobar errores mediante una expresión regular	patrón[3]	salida[2]		0, 1, 2, 3
19	Descartar sin cambios				

Tipo de preprocesamiento	Nombre	Parámetro 1	Parámetro 2	Parámetro 3	Gestores de errores compatibles
20	Descartar sin cambios con latido	segundos^5, 6^			
21	JavaScript	script[2]			
22	Patrón de Prometheus	patrón^6, 7^	`value`, `label`, `function`	salida^8, 9^	0, 1, 2, 3
23	Prometheus a JSON	patrón^6, 7^			0, 1, 2, 3
24	CSV a JSON	carácter[2]	carácter[2]	0,1	0, 1, 2, 3
25	Reemplazar	cadena de búsqueda[2]	reemplazo[2]		
26	Comprobar no compatible	ámbito[1]	patrón^3, 6^		1, 2, 3
27	XML a JSON				0, 1, 2, 3
28	Valor de recorrido SNMP	OID[2]	Formato: 0 - Sin cambios 1 - UTF-8 desde Hex-STRING 2 - MAC desde Hex-STRING 3 - Entero desde BITS		0, 1, 2, 3
29	Recorrido SNMP a JSON[10]	Nombre del campo[2]	Prefijo OID[2]	Formato: 0 - Sin cambios 1 - UTF-8 desde Hex-STRING 2 - MAC desde Hex-STRING 3 - Entero desde BITS	0, 1, 2, 3
30	Obtener valor SNMP	Formato: 1 - UTF-8 desde Hex-STRING 2 - MAC desde Hex-STRING 3 - Entero desde BITS			0, 1, 2, 3

[1] número entero o de coma flotante
 [2] cadena
 [3] expresión regular
 [4] JSONPath o XPath XML
 [5] entero positivo (con compatibilidad con sufijos de tiempo, por ejemplo, 30s, 1m, 2h, 1d)
 [6] macro de usuario, macro de LLD
 [7] patrón de Prometheus que sigue la sintaxis: `<metric name>{<label name>="<label value>", ...} == <value>`. Cada componente del patrón de Prometheus (métrica, nombre de etiqueta, valor de etiqueta y valor de métrica) puede ser una macro de usuario o una macro de LLD.
 [8] salida de Prometheus que sigue la sintaxis: `<label name>` (puede ser una macro de usuario o una macro de LLD) si `label` está seleccionado como segundo parámetro.
 [9] Una de las funciones de agregación: `sum`, `min`, `max`, `avg`, `count` si `function` está seleccionado como segundo parámetro.
 [10] Admite varios registros de ”Nombre del campo,Prefijo OID,Formato” delimitados por un carácter de nueva línea.

itemprototype.create

Descripción

`object itemprototype.create(object/array itemPrototypes)`

Este método le permite crear nuevos prototipos de item.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(objeto/array)` Prototipo de item a crear.

Además de las propiedades estándar del prototipo de item, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
ruleid	ID	ID de la regla LLD a la que pertenece el item. Comportamiento del parámetro: - *obligatorio*
preprocessing	array	Opciones de preprocesamiento del prototipo de item.
tags	array	Etiquetas del prototipo de item.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las métricas prototipo creadas bajo la propiedad `itemids`. El orden de los IDs devueltos coincide con el orden de las métricas prototipo enviadas.

Ejemplos

Creando una métrica prototipo

Cree una métrica prototipo para monitorizar el espacio libre en disco en un sistema de archivos descubierto. Las métricas descubiertas deben ser métricas numéricas del agente Zabbix actualizadas cada 30 segundos.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.create",
    "params": {
        "name": "Espacio libre en disco en {#FSNAME}",
        "key_": "vfs.fs.size[{#FSNAME},free]",
        "hostid": "10197",
        "ruleid": "27665",
        "type": 0,
        "value_type": 3,
        "interfaceid": "112",
        "delay": "30s"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "27666"
        ]
    },
    "id": 1
}
```

Creación de una métrica prototipo con preprocesamiento

Cree una métrica utilizando el cambio por segundo y un multiplicador personalizado como segundo paso.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.create",
    "params": {
        "name": "Tráfico de red entrante en {#IFNAME}",
        "key_": "net.if.in[{#IFNAME}]",
        "hostid": "10001",
        "ruleid": "27665",
        "type": 0,
        "value_type": 3,
        "delay": "60s",
        "units": "bps",
        "interfaceid": "1155",
        "preprocessing": [
            {
                "type": 10,
                "params": "",
                "error_handler": 0,
                "error_handler_params": ""
            },
            {
                "type": 1,
                "params": "8",
                "error_handler": 2,
                "error_handler_params": "10"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "44211"
        ]
    },
    "id": 1
}
```

Creación de prototipo de métrica dependiente

Cree un prototipo de métrica dependiente para el prototipo de métrica principal con ID 44211. Solo se permiten dependencias en el mismo equipo (plantilla/regla de descubrimiento), por lo tanto, la métrica principal y la dependiente deben tener el mismo hostid y ruleid.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.create",
    "params": {
      "hostid": "10001",
      "ruleid": "27665",
      "name": "Prototipo de métrica dependiente de prueba",
      "key_": "dependent.prototype",
      "type": 18,
      "master_itemid": "44211",
```

```
        "value_type": 3
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "44212"
        ]
    },
    "id": 1
}
```

Crear prototipo de métrica de agente HTTP

Crear prototipo de métrica con URL usando macro de usuario, campos de consulta y cabeceras personalizadas.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.create",
    "params": {
        "type": "19",
        "hostid": "10254",
        "ruleid": "28256",
        "interfaceid": "2",
        "name": "ejemplo de prototipo de métrica api",
        "key_": "api_http_item",
        "value_type": 3,
        "url": "{$URL_PROTOTYPE}",
        "query_fields": [
            {
                "name": "min",
                "value": "10"
            },
            {
                "name": "max",
                "value" "100"
            }
        ],
        "headers": [
            {
                "name": "X-Source",
                "value": "api"
            }
        ],
        "delay": "35"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "28305"
        ]
    },
    "id": 1
```

```
}
```

Crear prototipo de métrica de script

Cree una recopilación de datos simple utilizando un prototipo de métrica de script.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.create",
    "params": {
        "name": "Ejemplo de script",
        "key_": "custom.script.itemprototype",
        "hostid": "12345",
        "type": 21,
        "value_type": 4,
        "params": "var request = new HttpRequest();\nreturn request.post(\"https://postman-echo.com/post\"
        "parameters": [
            {
                "name": "host",
                "value": "{HOST.CONN}"
            }
        ],
        "timeout": "6s",
        "delay": "30s"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "23865"
        ]
    },
    "id": 1
}
```

Fuente

CItemPrototype::create() en *ui/include/classes/api/services/CItemPrototype.php*.

itemprototype.delete

Descripción

`object itemprototype.delete(array itemPrototypeIds)`

Este método le permite eliminar prototipos de item.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(array)` IDs de las métricas prototipo a eliminar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las métricas prototipo eliminadas bajo la propiedad `prototypeids`.

Ejemplos

Eliminando múltiples prototipos de item

Eliminar dos prototipos de item.
 Los prototipos de item dependientes se eliminan automáticamente si se elimina el item principal o el prototipo de item.

Request:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.delete",
    "params": [
        "27352",
        "27356"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "prototypeids": [
            "27352",
            "27356"
        ]
    },
    "id": 1
}
```

Fuente

CItemPrototype::delete() en *ui/include/classes/api/services/CItemPrototype.php*.

itemprototype.get

Descripción

`integer/array itemprototype.get(object parameters)`

El método permite recuperar prototipos de item según los parámetros proporcionados.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
discoveryids	ID/array	Devolver solo los prototipos de item que pertenecen a las reglas LLD indicadas.
graphids	ID/array	Devolver solo los prototipos de item que se utilizan en los prototipos de gráfico indicados.
hostids	ID/array	Devolver solo los prototipos de item que pertenecen a los hosts indicados.
inherited	boolean	Si se establece en `true`, devolver solo los prototipos de item heredados de un template.
itemids	ID/array	Devolver solo los prototipos de item con los IDs indicados.
monitored	boolean	Si se establece en `true`, devolver solo los prototipos de item habilitados que pertenecen a hosts monitorizados.
templated	boolean	Si se establece en `true`, devolver solo los prototipos de item que pertenecen a templates.
templateids	ID/array	Devolver solo los prototipos de item que pertenecen a los templates indicados.

Parámetro	Tipo	Descripción
triggerids	ID/array	Devolver solo los prototipos de item que se utilizan en los prototipos de trigger indicados.
selectDiscoveryData	query	Devolver una propiedad `discoveryData` con los datos del objeto de descubrimiento del prototipo de item. El objeto de descubrimiento del prototipo de item vincula un prototipo de item descubierto con un prototipo de item a partir del cual se descubrió. Tiene las siguientes propiedades: `parent_itemid` - `(string)` ID del prototipo de item a partir del cual se ha creado el prototipo de item; `key_` - `(string)` clave del prototipo de item; `status` - `(int)` estado de descubrimiento del prototipo de item: 0 - *(por defecto)* el prototipo de item está descubierto, 1 - el prototipo de item ya no se descubre; `ts_delete` - `(timestamp)` hora en la que se eliminará un prototipo de item que ya no se descubre; `ts_disable` - `(timestamp)` hora en la que se deshabilitará un prototipo de item que ya no se descubre; `disable_source` - `(int)` indicador de si el prototipo de item fue deshabilitado por una regla LLD o manualmente: 0 - *(por defecto)* deshabilitado automáticamente, 1 - deshabilitado por una regla LLD.
selectDiscoveryRule	query	Devolver una propiedad `discoveryRule` con la regla de descubrimiento de bajo nivel a la que pertenece el prototipo de item.
selectDiscoveryRulePrototype	query	Devolver una propiedad `discoveryRulePrototype` con el prototipo de regla LLD principal al que pertenece el prototipo de item.
selectGraphs	query	Devolver una propiedad `graphs` con los prototipos de gráfico en los que se utiliza el prototipo de item. Admite `count`.
selectHosts	query	Devolver una propiedad `hosts` con un array de hosts a los que pertenece el prototipo de item.
selectInheritedTags	query	Devolver una propiedad `inheritedTags` con las etiquetas que están en el template/host/templates enlazados.
selectTags	query	Devolver las etiquetas del prototipo de item en la propiedad `tags`.
selectTriggers	query	Devolver una propiedad `triggers` con los prototipos de trigger en los que se utiliza el prototipo de item. Admite `count`.
selectPreprocessing	query	Devolver una propiedad `preprocessing` con las opciones de preprocesamiento del prototipo de item.
selectValueMap	query	Devolver una propiedad `valuemap` con el mapa de valores del prototipo de item.
filter	object	Devolver solo aquellos resultados que coincidan exactamente con el filtro dado. Acepta un objeto, donde las claves son nombres de propiedades y los valores son un único valor o un array de valores con los que comparar. No admite propiedades del tipo de datos `text` data type. Admite propiedades adicionales: `host` - nombre técnico del host al que pertenece el prototipo de item.
limitSelects	integer	Limita el número de registros devueltos por subconsultas. Se aplica a las siguientes subconsultas: `selectGraphs` - los resultados se ordenarán por `name`; `selectTriggers` - los resultados se ordenarán por `description`.
sortfield	string/array	Ordenar el resultado por las propiedades indicadas. Valores posibles: `itemid`, `name`, `key_`, `delay`, `type`, `status`, `history`, `trends`, `discovered`.

Parámetro	Tipo	Descripción
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
editable	boolean	
excludeSearch	boolean	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperar prototipos de item de una regla LLD

Recuperar todos los prototipos de item para un ID de regla LLD específico.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.get",
    "params": {
        "output": "extend",
        "discoveryids": "27426"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "itemid": "23077",
            "type": "0",
            "snmp_oid": "",
            "hostid": "10079",
            "name": "Incoming network traffic on en0",
            "key_": "net.if.in[en0]",
            "delay": "1m",
            "history": "1w",
            "trends": "365d",
            "status": "0",
            "value_type": "3",
            "trapper_hosts": "",
            "units": "bps",
            "formula": "",
            "logtimefmt": "",
            "templateid": "0",
            "valuemapid": "0",
            "params": "",
            "ipmi_sensor": "",
            "authtype": "0",
            "username": "",
            "password": "",
```

```
            "publickey": "",
            "privatekey": "",
            "flags": "2",
            "interfaceid": "0",
            "description": "",
            "inventory_link": "0",
            "lifetime": "7d",
            "evaltype": "0",
            "jmx_endpoint": "",
            "master_itemid": "0",
            "timeout": "",
            "url": "",
            "query_fields": [],
            "posts": "",
            "status_codes": "200",
            "follow_redirects": "1",
            "post_type": "0",
            "http_proxy": "",
            "headers": [],
            "retrieve_mode": "0",
            "request_method": "0",
            "output_format": "0",
            "ssl_cert_file": "",
            "ssl_key_file": "",
            "ssl_key_password": "",
            "verify_peer": "0",
            "verify_host": "0",
            "allow_traps": "0",
            "discover": "0",
            "uuid": "",
            "lifetime_type": "0",
            "enabled_lifetime_type": "2",
            "enabled_lifetime": "0",
            "parameters": []
        },
        {
            "itemid": "10010",
            "type": "0",
            "snmp_oid": "",
            "hostid": "10001",
            "name": "Processor load (1 min average per core)",
            "key_": "system.cpu.load[percpu,avg1]",
            "delay": "1m",
            "history": "1w",
            "trends": "365d",
            "status": "0",
            "value_type": "0",
            "trapper_hosts": "",
            "units": "",
            "formula": "",
            "logtimefmt": "",
            "templateid": "0",
            "valuemapid": "0",
            "params": "",
            "ipmi_sensor": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "flags": "2",
            "interfaceid": "0",
```

```
            "description": "The processor load is calculated as system CPU load divided by number of CPU c
            "inventory_link": "0",
            "lifetime": "7d",
            "evaltype": "0",
            "jmx_endpoint": "",
            "master_itemid": "0",
            "timeout": "",
            "url": "",
            "query_fields": [],
            "posts": "",
            "status_codes": "200",
            "follow_redirects": "1",
            "post_type": "0",
            "http_proxy": "",
            "headers": [],
            "retrieve_mode": "0",
            "request_method": "0",
            "output_format": "0",
            "ssl_cert_file": "",
            "ssl_key_file": "",
            "ssl_key_password": "",
            "verify_peer": "0",
            "verify_host": "0",
            "allow_traps": "0",
            "discover": "0",
            "uuid": "",
            "lifetime_type": "0",
            "enabled_lifetime_type": "2",
            "enabled_lifetime": "0",
            "parameters": []
        }
    ],
    "id": 1
}
```

Buscar un item dependiente

Buscar un item dependiente para un ID de item específico.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.get",
    "params": {
        "output": "extend",
        "filter": {
            "type": 18,
            "master_itemid": "25545"
        },
        "limit": "1"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "itemid": "25547",
            "type": "18",
            "snmp_oid": "",
            "hostid": "10116",
```

```
            "name": "Seconds",
            "key_": "apache.status.uptime.seconds",
            "delay": "0",
            "history": "90d",
            "trends": "365d",
            "status": "0",
            "value_type": "3",
            "trapper_hosts": "",
            "units": "",
            "formula": "",
            "logtimefmt": "",
            "templateid": "0",
            "valuemapid": "0",
            "params": "",
            "ipmi_sensor": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "flags": "0",
            "interfaceid": "0",
            "description": "",
            "inventory_link": "0",
            "lifetime": "7d",
            "evaltype": "0",
            "jmx_endpoint": "",
            "master_itemid": "25545",
            "timeout": "",
            "url": "",
            "query_fields": [],
            "posts": "",
            "status_codes": "200",
            "follow_redirects": "1",
            "post_type": "0",
            "http_proxy": "",
            "headers": [],
            "retrieve_mode": "0",
            "request_method": "0",
            "output_format": "0",
            "ssl_cert_file": "",
            "ssl_key_file": "",
            "ssl_key_password": "",
            "verify_peer": "0",
            "verify_host": "0",
            "allow_traps": "0",
            "discover": "0",
            "uuid": "",
            "lifetime_type": "0",
            "enabled_lifetime_type": "2",
            "enabled_lifetime": "0",
            "parameters": []
        }
    ],
    "id": 1
}
```


Buscar prototipo de item HTTP agent

Buscar prototipo de item HTTP agent con método de solicitud HEAD para un ID de host específico.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.get",
    "params": {
        "hostids": "10254",
        "filter": {
            "type": 19,
            "request_method": 3
        }
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "itemid": "28257",
            "type": "19",
            "snmp_oid": "",
            "hostid": "10254",
            "name": "discovered",
            "key_": "item[{#INAME}]",
            "delay": "{#IUPDATE}",
            "history": "90d",
            "trends": "30d",
            "status": "0",
            "value_type": "3",
            "trapper_hosts": "",
            "units": "",
            "formula": "",
            "logtimefmt": "",
            "templateid": "28255",
            "valuemapid": "0",
            "params": "",
            "ipmi_sensor": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "flags": "2",
            "interfaceid": "2",
            "description": "",
            "inventory_link": "0",
            "lifetime": "7d",
            "evaltype": "0",
            "jmx_endpoint": "",
            "master_itemid": "0",
            "timeout": "",
            "url": "{#IURL}",
            "query_fields": [],
            "posts": "",
            "status_codes": "",
            "follow_redirects": "0",
            "post_type": "0",
            "http_proxy": "",
            "headers": [],
            "retrieve_mode": "0",
            "request_method": "3",
            "output_format": "0",
```

```
            "ssl_cert_file": "",
            "ssl_key_file": "",
            "ssl_key_password": "",
            "verify_peer": "0",
            "verify_host": "0",
            "allow_traps": "0",
            "discover": "0",
            "uuid": "",
            "lifetime_type": "0",
            "enabled_lifetime_type": "2",
            "enabled_lifetime": "0",
            "parameters": []
        }
    ],
    "id": 1
}
```

Ver también

- Host
- Prototipo de gráfico
- Prototipo de trigger

Fuente

CItemPrototype::get() en *ui/include/classes/api/services/CItemPrototype.php*.

itemprototype.update

Descripción

`object itemprototype.update(object/array itemPrototypes)`

Este método permite actualizar los item prototypes existentes.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object/array)` Propiedades de la item prototype que se actualizarán.

La propiedad `itemid` debe definirse para cada item prototype, todas las demás propiedades son opcionales. Sólo se actualizarán las propiedades pasadas, todas las demás permanecerán sin cambios.

Además de las propiedades estándar de la item prototype, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
preprocessing	array	Opciones de preprocesamiento de la item prototype para reemplazar las opciones de preprocesamiento actuales. Comportamiento del parámetro: - *sólo lectura* para objetos heredados
tags	array	Etiquetas de la item prototype.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las métricas prototipo actualizadas bajo la propiedad `itemids`.

Ejemplos

Cambiar la interfaz de una métrica prototipo

Cambie la interfaz del equipo que será utilizada por las métricas descubiertas.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.update",
    "params": {
        "itemid": "27428",
        "interfaceid": "132"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "27428"
        ]
    },
    "id": 1
}
```

Actualizar prototipo de métrica dependiente

Actualizar el prototipo de métrica dependiente con un nuevo ID de prototipo de métrica principal. Solo se permiten dependencias en el mismo equipo (plantilla/regla de descubrimiento), por lo tanto, la métrica principal y la dependiente deben tener el mismo hostid y ruleid.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.update",
    "params": {
        "master_itemid": "25570",
        "itemid": "189030"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "189030"
        ]
    },
    "id": 1
}
```

Actualizar prototipo de métrica de agente HTTP

Cambiar los campos de consulta y eliminar todas las cabeceras personalizadas.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.update",
    "params": {
        "itemid":"28305",
        "query_fields": [
            {
                "name": "random",
                "value": "qwertyuiopasdfghjklzxcvbnm"
            }
```

```
        ],
        "headers": []
    }
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "28305"
        ]
    },
    "id": 1
}
```

Actualización de las opciones de preprocesamiento de la métrica

Actualice una plantilla de métrica con la regla de preprocesamiento de métrica “Multiplicador personalizado”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.update",
    "params": {
        "itemid": "44211",
        "preprocessing": [
            {
                "type": 1,
                "params": "4",
                "error_handler": 2,
                "error_handler_params": "5"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "44211"
        ]
    },
    "id": 1
}
```

Actualizar un prototipo de métrica de script

Actualice un prototipo de métrica de script con un script diferente y elimine los parámetros innecesarios que fueron utilizados por el script anterior.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.update",
    "params": {
        "itemid": "23865",
        "parameters": [],
        "script": "Zabbix.log(3, 'Log test');\nreturn 1;"
    },
```

```
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "23865"
        ]
    },
    "id": 1
}
```

Fuente

CItemPrototype::update() en *ui/include/classes/api/services/CItemPrototype.php*.

Prototipo de regla LLD

Esta clase está diseñada para trabajar con prototipos de reglas de bajo nivel de descubrimiento.

Referencias de objetos:

- Prototipo de regla LLD
  - Cabecera HTTP
  - Campo de consulta HTTP
- Filtro de prototipo de regla LLD
  - Condición de filtro de prototipo de regla LLD
- Ruta de macro de prototipo LLD
- Preprocesamiento de prototipo de regla LLD
- Anulaciones de prototipo de regla LLD
  - Filtro de anulación de prototipo de regla LLD
    - Condición de filtro de anulación de prototipo de regla LLD
  - Operación de anulación de prototipo de regla LLD
    - Estado de operación de anulación de prototipo de regla LLD
    - Descubrimiento de operación de anulación de prototipo de regla LLD
    - Periodo de operación de anulación de prototipo de regla LLD
    - Histórico de operación de anulación de prototipo de regla LLD
    - Tendencias de operación de anulación de prototipo de regla LLD
    - Severidad de operación de anulación de prototipo de regla LLD
    - Etiqueta de operación de anulación de prototipo de regla LLD
    - Template de operación de anulación de prototipo de regla LLD
    - Inventario de operación de anulación de prototipo de regla LLD

Métodos disponibles:

- discoveryruleprototype.create - crear nuevos prototipos de reglas LLD
- discoveryruleprototype.delete - eliminar prototipos de reglas LLD
- discoveryruleprototype.get - recuperar prototipos de reglas LLD
- discoveryruleprototype.update - actualizar prototipos de reglas LLD

Objeto prototipo de regla LLD

Los siguientes objetos están directamente relacionados con la API `discoveryruleprototype`.

Prototipo de regla de LLD

El objeto de prototipo de regla de descubrimiento de bajo nivel tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
itemid	ID	ID del prototipo de regla de LLD. Comportamiento de la propiedad: - *solo lectura* - *obligatoria* para operaciones de actualización
ruleid	ID	ID de la regla de LLD o del prototipo de regla de LLD principal. Comportamiento de la propiedad: - *solo lectura* - *obligatoria* para operaciones de creación
delay	string	Intervalo de actualización del prototipo de regla de LLD. Acepta un número de segundos (por ejemplo, 1, 60, 300), un valor con un sufijo de tiempo (por ejemplo, 30s, 1m, 2h, 1d) o, de forma opcional, uno o más intervalos personalizados, todos separados por punto y coma. Los intervalos personalizados pueden ser una combinación de intervalos flexibles y de planificación. Acepta macros de usuario. Si se utiliza una macro, el valor debe ser una única macro. No se admiten varias macros ni macros combinadas con texto. Los intervalos flexibles pueden escribirse como dos macros separadas por una barra diagonal (por ejemplo, `{$FLEX_INTERVAL}/{$FLEX_PERIOD}`). Ejemplo: `1h;wd1-5h9-18;{$Macro1}/1-7,00:00-24:00;0/6-7,12:00-24:00;{$Macr` Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”Zabbix agent” (0), ”Simple check” (3), ”Zabbix internal” (5), ”External check” (10), ”Database monitor” (11), ”IPMI agent” (12), ”SSH agent” (13), ”TELNET agent” (14), ”JMX agent” (16), ”HTTP agent” (19), ”SNMP agent” (20), ”Script” (21), ”Browser” (22), o si `type` está establecido en ”Zabbix agent (active)” (7) y `key_` no contiene ”mqtt.get”
hostid	ID	ID del host al que pertenece el prototipo de regla de LLD. Comportamiento de la propiedad: - *constante* - *obligatoria* para operaciones de creación
flags	integer	Origen del prototipo de regla de LLD. Valores posibles: 3 - un prototipo de regla de descubrimiento de bajo nivel; 7 - un prototipo de regla de descubrimiento de bajo nivel descubierto. Comportamiento de la propiedad: - *solo lectura*
interfaceid	ID	ID de la interfaz del host del prototipo de regla de LLD. Comportamiento de la propiedad: - *obligatoria* si el prototipo de regla de LLD pertenece a un host y `type` está establecido en ”Zabbix agent”, ”IPMI agent”, ”JMX agent” o ”SNMP agent” - *compatible* si el prototipo de regla de LLD pertenece a un host y `type` está establecido en ”Simple check”, ”External check”, ”SSH agent”, ”TELNET agent” o ”HTTP agent”
key_	string	Clave del prototipo de regla de LLD. Se requiere al menos una macro de LLD. Comportamiento de la propiedad: - *obligatoria* para operaciones de creación - *solo lectura* para objetos heredados

Propiedad	Tipo	Descripción
name	string	Nombre del prototipo de regla de LLD. Comportamiento de la propiedad: - *obligatoria* para operaciones de creación - *solo lectura* para objetos heredados
type	integer	Tipo del prototipo de regla de LLD. Valores posibles: 0 - Zabbix agent; 2 - Zabbix trapper; 3 - Simple check; 5 - Zabbix internal; 7 - Zabbix agent (active); 10 - External check; 11 - Database monitor; 12 - IPMI agent; 13 - SSH agent; 14 - TELNET agent; 16 - JMX agent; 18 - Dependent item; 19 - HTTP agent; 20 - SNMP agent; 21 - Script; 22 - Browser; 23 - Nested. Comportamiento de la propiedad: - *obligatoria* para operaciones de creación - *solo lectura* para objetos heredados
url	string	Cadena de URL. Admite macros de usuario, {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {HOST.PORT}, {ITEM.ID}, {ITEM.KEY}. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados
allow_traps	integer	Permite que el valor se complete de forma similar al item trapper. Valores posibles: 0 - *(predeterminado)* No aceptar datos entrantes; 1 - Aceptar datos entrantes. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent”
authtype	integer	Método de autenticación. Valores posibles si `type` está establecido en ”SSH agent”: 0 - *(predeterminado)* contraseña; 1 - clave pública. Valores posibles si `type` está establecido en ”HTTP agent”: 0 - *(predeterminado)* ninguno; 1 - básico; 2 - NTLM. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”SSH agent” o ”HTTP agent” - *solo lectura* para objetos heredados (si `type` está establecido en ”HTTP agent”)
description	string	Descripción del prototipo de regla de LLD.

Propiedad	Tipo	Descripción
error	string	Texto del error si existen problemas al actualizar el valor del prototipo de regla de LLD. Comportamiento de la propiedad: - *solo lectura*
follow_redirects	integer	Seguir las redirecciones de respuesta al sondear datos. Valores posibles: 0 - No seguir redirecciones; 1 - *(predeterminado)* Seguir redirecciones. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados
headers	array	Matriz de encabezados que se enviarán al realizar una solicitud HTTP. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados
http_proxy	string	Cadena de conexión del proxy HTTP(S). Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados
ipmi_sensor	string	Sensor IPMI. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”IPMI agent” y `key_` no está establecido en ”ipmi.get” - *compatible* si `type` está establecido en ”IPMI agent” - *solo lectura* para objetos heredados
jmx_endpoint	string	Cadena de conexión personalizada del JMX agent. Predeterminado: service:jmx:rmi:///jndi/rmi://{HOST.CONN}:{HOST.PORT}/jmxrmi Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”JMX agent”
lifetime	string	Periodo tras el cual se eliminarán los items que ya no se descubran. Acepta un número de segundos (por ejemplo, `1`, `60`, `300`), un valor con un sufijo de tiempo (por ejemplo, `30s`, `1m`, `2h`, `1d`) o una macro de usuario. Predeterminado: `7d`.
lifetime_type	integer	Escenario para eliminar los recursos de LLD perdidos. Valores posibles: 0 - *(predeterminado)* Eliminar cuando se alcance el umbral de duración; 1 - No eliminar; 2 - Eliminar inmediatamente.
enabled_lifetime	string	Periodo tras el cual se deshabilitarán los items que ya no se descubran. Acepta un número de segundos (por ejemplo, `1`, `60`, `300`), un valor con un sufijo de tiempo (por ejemplo, `30s`, `1m`, `2h`, `1d`) o una macro de usuario. Predeterminado: `0`.

Propiedad	Tipo	Descripción
enabled_lifetime_type	integer	Escenario para deshabilitar los recursos de LLD perdidos. Valores posibles: 0 - Deshabilitar cuando se alcance el umbral de duración; 1 - No deshabilitar; 2 - *(predeterminado)* Deshabilitar inmediatamente.
master_itemid	ID	ID del item maestro. La regla de descubrimiento no puede ser el item maestro de otra regla de descubrimiento. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”Dependent item” - *solo lectura* para objetos heredados
output_format	integer	Indica si la respuesta debe convertirse a JSON. Valores posibles: 0 - *(predeterminado)* Almacenar sin procesar; 1 - Convertir a JSON. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados
params	string	Parámetros adicionales según el tipo de prototipo de regla de LLD: - script ejecutado para reglas de LLD SSH y Telnet; - consulta SQL para reglas de LLD de monitor de base de datos; - fórmula para reglas de LLD calculadas; - script para reglas de LLD de script y navegador. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”Database monitor”, ”SSH agent”, ”TELNET agent”, ”Script” o ”Browser” - *solo lectura* para objetos heredados (si `type` está establecido en ”Script” o ”Browser”)
parameters	object/array	Parámetros adicionales si `type` está establecido en ”Script” o ”Browser”. Matriz de objetos con las propiedades `name` y `value`, donde `name` debe ser único. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”Script” o ”Browser” - *solo lectura* para objetos heredados
password	string	Contraseña para la autenticación. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”JMX agent” y `username` está establecido - *compatible* si `type` está establecido en ”Simple check”, ”Database monitor”, ”SSH agent”, ”TELNET agent” o ”HTTP agent” - *solo lectura* para objetos heredados (si `type` está establecido en ”HTTP agent”)
post_type	integer	Tipo de cuerpo de datos de publicación almacenado en la propiedad `posts`. Valores posibles: 0 - *(predeterminado)* Datos sin procesar; 2 - Datos JSON; 3 - Datos XML. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados

Propiedad	Tipo	Descripción
posts	string	Datos del cuerpo de la solicitud HTTP(S). Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”HTTP agent” y `post_type` está establecido en ”JSON data” o ”XML data” - *compatible* si `type` está establecido en ”HTTP agent” y `post_type` está establecido en ”Raw data” - *solo lectura* para objetos heredados
privatekey	string	Nombre del archivo de clave privada. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”SSH agent” y `authtype` está establecido en ”public key”
publickey	string	Nombre del archivo de clave pública. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”SSH agent” y `authtype` está establecido en ”public key”
query_fields	array	Matriz de campos de consulta que se enviarán al realizar una solicitud HTTP. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados
request_method	integer	Tipo de método de solicitud. Valores posibles: 0 - *(predeterminado)* GET; 1 - POST; 2 - PUT; 3 - HEAD. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados
retrieve_mode	integer	Parte de la respuesta que debe almacenarse. Valores posibles si `request_method` está establecido en ”GET”, ”POST” o ”PUT”: 0 - *(predeterminado)* Cuerpo; 1 - Encabezados; 2 - Se almacenarán el cuerpo y los encabezados. Valores posibles si `request_method` está establecido en ”HEAD”: 1 - Encabezados. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados
snmp_oid	string	OID de SNMP. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”SNMP agent” - *solo lectura* para objetos heredados
ssl_cert_file	string	Ruta del archivo de clave SSL pública. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados

Propiedad	Tipo	Descripción
ssl_key_file	string	Ruta del archivo de clave SSL privada. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados
ssl_key_password	string	Contraseña del archivo de clave SSL. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados
state	integer	Estado del prototipo de regla de LLD. Valores posibles: 0 - *(predeterminado)* normal; 1 - no compatible. Comportamiento de la propiedad: - *solo lectura*
status	integer	Estado del prototipo de regla de LLD. Valores posibles: 0 - *(predeterminado)* prototipo de regla de LLD habilitado; 1 - prototipo de regla de LLD deshabilitado.
status_codes	string	Rangos de códigos de estado HTTP requeridos, separados por comas. También admite macros de usuario como parte de una lista separada por comas. Ejemplo: 200,200-{$M},{$M},200-400 Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados
templateid	ID	ID del prototipo de regla de LLD principal del template. Comportamiento de la propiedad: - *solo lectura*
timeout	string	Tiempo de espera de la solicitud de sondeo de datos del item. Acepta un número de segundos (por ejemplo, 1, 60, 300) o un valor con un sufijo de tiempo (por ejemplo, 30s, 1m). También acepta macros de usuario. Rango de valores posibles: 1-600s. Predeterminado: ”” - usar la configuración del proxy/global. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”Zabbix agent” (0), ”Simple check” (3) y `key_` no comienza por ”`vmware.`” ni ”`icmpping`”, ”Zabbix agent (active)” (7), ”External check” (10), ”Database monitor” (11), ”SSH agent” (13), ”TELNET agent” (14), ”HTTP agent” (19), ”SNMP agent” (20) y `snmp_oid` comienza por ”`walk[`” o ”`get[`”, ”Script” (21), ”Browser” (22) - *solo lectura* para objetos heredados

Propiedad	Tipo	Descripción
trapper_hosts	string	Hosts permitidos. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”Zabbix trapper”, o si `type` está establecido en ”HTTP agent” y `allow_traps` está establecido en ”Allow incoming data”. Si se omite la propiedad, se establece automáticamente en la macro de usuario `{$TRAPPER.ALLOWED_HOSTS}` (valor predeterminado). De forma predeterminada, `{$TRAPPER.ALLOWED_HOSTS}` es `127.0.0.1,::1`. Si la propiedad se establece en una cadena vacía, se denegarán las conexiones entrantes de todos los hosts.
username	string	Nombre de usuario para la autenticación. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”SSH agent”, ”TELNET agent”, o si `type` está establecido en ”JMX agent” y `password` está establecido - *compatible* si `type` está establecido en ”Simple check”, ”Database monitor” o ”HTTP agent” - *solo lectura* para objetos heredados (si `type` está establecido en ”HTTP agent”)
uuid	string	Identificador único universal, utilizado para vincular prototipos de reglas de LLD importados con los ya existentes. Se genera automáticamente si no se proporciona. Comportamiento de la propiedad: - *compatible* si el prototipo de regla de LLD pertenece a un template
verify_host	integer	Indica si se debe validar que el nombre del host de la conexión coincida con el del certificado del host. Valores posibles: 0 - *(predeterminado)* No validar; 1 - Validar. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados
verify_peer	integer	Indica si se debe validar que el certificado del host sea auténtico. Valores posibles: 0 - *(predeterminado)* No validar; 1 - Validar. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados

Encabezado HTTP

El objeto header tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
name	string	Nombre del encabezado HTTP. Comportamiento de la propiedad: - *obligatorio*

Propiedad	Tipo	Descripción
value	string	Valor del encabezado. Comportamiento de la propiedad: - *obligatorio*

Campo de consulta HTTP

El objeto de campo de consulta define un nombre y un valor que se utilizan para especificar un parámetro de URL. Tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
name	string	Nombre del parámetro. Comportamiento de la propiedad: - *obligatorio*
value	string	Valor del parámetro. Comportamiento de la propiedad: - *obligatorio*

Filtro de prototipo de regla LLD

El objeto de filtro de prototipo de regla LLD define un conjunto de condiciones que pueden utilizarse para filtrar objetos descubiertos. Tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
conditions	object/array	Conjunto de condiciones de filtro a utilizar para filtrar resultados. Las condiciones se ordenarán en el orden de su colocación en la fórmula. Comportamiento de la propiedad: - *obligatorio*
evaltype	integer	Método de evaluación de la condición de filtro. Valores posibles: 0 - And/Or; 1 - And; 2 - Or; 3 - Expresión personalizada. Comportamiento de la propiedad: - *obligatorio*
eval_formula	string	Expresión generada que se utilizará para evaluar las condiciones del filtro. La expresión contiene IDs que hacen referencia a condiciones de filtro específicas por su `formulaid`. El valor de `eval_formula` es igual al valor de `formula` para los filtros con una expresión personalizada. Comportamiento de la propiedad: - *sólo lectura*

Propiedad	Tipo	Descripción
formula	string	Expresión definida por el usuario que se utilizará para evaluar las condiciones de los filtros con una expresión personalizada. La expresión debe contener IDs que hagan referencia a condiciones de filtro específicas por su `formulaid`. Los IDs utilizados en la expresión deben coincidir exactamente con los definidos en las condiciones del filtro: ninguna condición puede quedar sin usar u omitida. Comportamiento de la propiedad: - *obligatorio* si `evaltype` está establecido en ”expresión personalizada”

Condición de filtro de la regla LLD prototipo

El objeto de condición de filtro de la regla LLD prototipo define una comprobación separada que se realiza sobre el valor de una macro LLD. Tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
macro	string	Macro LLD sobre la que realizar la comprobación. Comportamiento de la propiedad: - *obligatorio*
value	string	Valor con el que comparar. Comportamiento de la propiedad: - *obligatorio* si `operator` está establecido en ”coincide con la expresión regular” o ”no coincide con la expresión regular”
formulaid	string	ID único arbitrario que se utiliza para hacer referencia a la condición desde una expresión personalizada. Solo puede contener letras mayúsculas. El ID debe ser definido por el usuario al modificar las condiciones del filtro, pero se generará de nuevo al solicitarlas posteriormente. Comportamiento de la propiedad: - *obligatorio* si `evaltype` del objeto de filtro de la regla LLD prototipo está establecido en ”expresión personalizada”
operator	integer	Operador de condición. Valores posibles: 8 - *(por defecto)* coincide con la expresión regular; 9 - no coincide con la expresión regular; 12 - existe; 13 - no existe.

Note:
Para comprender mejor cómo utilizar los filtros con varios tipos de expresiones, consulte los ejemplos en las páginas de los métodos discoveryruleprototype.get y discoveryruleprototype.create.

Ruta de macro LLD

La ruta de la macro LLD tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
lld_macro	string	Macro LLD. Comportamiento de la propiedad: - *obligatorio*

Propiedad	Tipo	Descripción
path	string	Selector para el valor que se asignará a la macro correspondiente. Comportamiento de la propiedad: - *obligatorio*

Preprocesamiento del prototipo de regla LLD

El objeto de preprocesamiento del prototipo de regla LLD tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
type	integer	El tipo de opción de preprocesamiento. Valores posibles: 5 - Expresión regular; 11 - XML XPath; 12 - JSONPath; 14 - Coincide con la expresión regular; 15 - No coincide con la expresión regular; 16 - Comprobar si hay un error en JSON; 17 - Comprobar si hay un error en XML; 20 - Descartar si no ha cambiado con latido; 21 - JavaScript; 23 - Prometheus a JSON; 24 - CSV a JSON; 25 - Reemplazar; 27 - XML a JSON; 28 - Valor de recorrido SNMP; 29 - Recorrido SNMP a JSON; 30 - Obtener valor mediante SNMP. Comportamiento de la propiedad: - *obligatoria*
params	string	Parámetros adicionales utilizados por la opción de preprocesamiento. Los parámetros múltiples se separan mediante el carácter de nueva línea (\n). Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”Expresión regular” (5), ”XML XPath” (11), ”JSONPath” (12), ”Coincide con la expresión regular” (14), ”No coincide con la expresión regular” (15), ”Comprobar si hay un error en JSON” (16), ”Comprobar si hay un error en XML” (17), ”Descartar si no ha cambiado con latido” (20), ”JavaScript” (21), ”Prometheus a JSON” (23), ”CSV a JSON” (24), ”Reemplazar” (25), ”Valor de recorrido SNMP” (28), ”Recorrido SNMP a JSON” (29) u ”Obtener valor mediante SNMP” (30)

Propiedad	Tipo	Descripción
error_handler	integer	Tipo de acción utilizado en caso de fallo del paso de preprocesamiento. Valores posibles: 0 - El mensaje de error lo establece el server de Zabbix; 1 - Descartar el valor; 2 - Establecer un valor personalizado; 3 - Establecer un mensaje de error personalizado. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”Expresión regular” (5), ”XML XPath” (11), ”JSONPath” (12), ”Coincide con la expresión regular” (14), ”No coincide con la expresión regular” (15), ”Comprobar si hay un error en JSON” (16), ”Comprobar si hay un error en XML” (17), ”Prometheus a JSON” (23), ”CSV a JSON” (24), ”XML a JSON” (27), ”Valor de recorrido SNMP” (28), ”Recorrido SNMP a JSON” (29) u ”Obtener valor mediante SNMP” (30)
error_handler_params	string	Parámetros del gestor de errores. Comportamiento de la propiedad: - *obligatoria* si `error_handler` está establecido en ”Establecer un valor personalizado” o ”Establecer un mensaje de error personalizado”

Los siguientes parámetros y gestores de errores son compatibles con cada tipo de preprocesamiento.

Tipo de preprocesamiento	Nombre	Parámetro 1	Parámetro 2	Parámetro 3	Gestores de errores compatibles
5	Expresión regular	patrón[1]	salida[2]		0, 1, 2, 3
11	XML XPath	ruta[3]			0, 1, 2, 3
12	JSONPath	ruta[3]			0, 1, 2, 3
14	Coincide con la expresión regular	patrón[1]			0, 1, 2, 3
15	No coincide con la expresión regular	patrón[1]			0, 1, 2, 3
16	Comprobar si hay un error en JSON	ruta[3]			0, 1, 2, 3

Tipo de preprocesamiento	Nombre	Parámetro 1	Parámetro 2	Parámetro 3	Gestores de errores compatibles
17	Comprobar si hay un error en XML	ruta[3]			0, 1, 2, 3
20	Descartar si no ha cambiado con latido	segundos^[4], 5^			
21	JavaScript	script[2]			
23	Prometheus a JSON	patrón^5, 6^			0, 1, 2, 3
24	CSV a JSON	carácter[2]	carácter[2]	0,1	0, 1, 2, 3
25	Reemplazar	cadena de búsqueda[2]	reemplazo[2]		
27	XML a JSON				0, 1, 2, 3
28	Valor de recorrido SNMP	OID[2]	Formato: 0 - Sin cambios 1 - UTF-8 desde Hex-STRING 2 - MAC desde Hex-STRING 3 - Entero desde BITS		0, 1, 2, 3
29	Recorrido SNMP a JSON[7]	Nombre del campo[2]	Prefijo OID[2]	Formato: 0 - Sin cambios 1 - UTF-8 desde Hex-STRING 2 - MAC desde Hex-STRING 3 - Entero desde BITS	0, 1, 2, 3
30	Obtener valor mediante SNMP	Formato: 1 - UTF-8 desde Hex-STRING 2 - MAC desde Hex-STRING 3 - Entero desde BITS			0, 1, 2, 3

[1] expresión regular
 [2] cadena
 [3] JSONPath o XML XPath
 [4] entero positivo (con compatibilidad con sufijos de tiempo, por ejemplo, 30s, 1m, 2h, 1d)
 [5] macro de usuario
 [6] patrón de Prometheus que sigue la sintaxis: `<metric name>{<label name>="<label value>", ...} == <value>`. Cada componente del patrón de Prometheus (métrica, nombre de etiqueta, valor de etiqueta y valor de métrica) puede ser una macro de usuario.
 [7] Admite varios registros de "Nombre del campo,Prefijo OID,Formato" delimitados por un carácter de nueva línea.

Anulaciones de prototipos de reglas LLD

El objeto de anulaciones de prototipos de reglas LLD define un conjunto de reglas (filtros, condiciones y operaciones) que se utilizan para anular las propiedades de diferentes objetos prototipo. Tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
name	string	Nombre único de la anulación. Comportamiento de la propiedad: - *obligatorio*
step	integer	Número de orden único de la anulación. Comportamiento de la propiedad: - *obligatorio*
stop	integer	Detener el procesamiento de las siguientes anulaciones si coincide. Valores posibles: 0 - *(por defecto)* no detener el procesamiento de las anulaciones; 1 - detener el procesamiento de las anulaciones si el filtro coincide.
filter	object	Filtro de anulación.
operations	object/array	Operaciones de anulación.

Filtro de anulación de prototipo de regla LLD

El objeto de filtro de anulación de prototipo de regla LLD define un conjunto de condiciones que, si coinciden con el objeto descubierto, se aplica la anulación. Tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
conditions	object/array	Conjunto de condiciones de filtro de anulación a utilizar para hacer coincidir los objetos descubiertos. Las condiciones se ordenarán en el orden de su colocación en la fórmula. Comportamiento de la propiedad: - *requerido*
evaltype	integer	Método de evaluación de la condición de filtro de anulación. Valores posibles: 0 - And/Or; 1 - And; 2 - Or; 3 - Expresión personalizada. Comportamiento de la propiedad: - *requerido*
eval_formula	string	Expresión generada que se utilizará para evaluar las condiciones del filtro de anulación. La expresión contiene IDs que hacen referencia a condiciones específicas del filtro de anulación por su `formulaid`. El valor de `eval_formula` es igual al valor de `formula` para los filtros con una expresión personalizada. Comportamiento de la propiedad: - *solo lectura*
formula	string	Expresión definida por el usuario que se utilizará para evaluar las condiciones de los filtros de anulación con una expresión personalizada. La expresión debe contener IDs que hagan referencia a condiciones específicas del filtro de anulación por su `formulaid`. Los IDs utilizados en la expresión deben coincidir exactamente con los definidos en las condiciones del filtro de anulación: ninguna condición puede quedar sin usar u omitirse. Comportamiento de la propiedad: - *requerido* si `evaltype` está establecido en "expresión personalizada"

Condición de filtro de anulación de la regla LLD

El objeto de condición de filtro de anulación de la regla LLD define una comprobación separada para realizar sobre el valor de una macro LLD. Tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
macro	string	Macro LLD sobre la que realizar la comprobación. Comportamiento de la propiedad: - *obligatorio*
value	string	Valor con el que comparar. Comportamiento de la propiedad: - *obligatorio* si `operator` está establecido en ”coincide con la expresión regular” o ”no coincide con la expresión regular”
formulaid	string	ID único arbitrario que se utiliza para hacer referencia a la condición desde una expresión personalizada. Sólo puede contener letras mayúsculas. El ID debe ser definido por el usuario al modificar las condiciones del filtro, pero se generará de nuevo al solicitarlas posteriormente. Comportamiento de la propiedad: - *obligatorio* si `evaltype` del objeto de filtro de anulación de la regla LLD está establecido en ”expresión personalizada”
operator	integer	Operador de condición. Valores posibles: 8 - *(por defecto)* coincide con la expresión regular; 9 - no coincide con la expresión regular; 12 - existe; 13 - no existe.

Operación de anulación de la regla LLD

La operación de anulación de la regla LLD es una combinación de condiciones y acciones a realizar sobre el objeto prototipo. Tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
operationobject	integer	Tipo de objeto descubierto sobre el que realizar la acción. Valores posibles: 0 - Item prototype; 1 - Trigger prototype; 2 - Graph prototype; 3 - Host prototype. Comportamiento de la propiedad: - *obligatorio*
operator	integer	Operador de condición de anulación. Valores posibles: 0 - *(por defecto)* igual; 1 - no igual; 2 - contiene; 3 - no contiene; 8 - coincide; 9 - no coincide.
value	string	Patrón para hacer coincidir el nombre del item, trigger, gráfico o prototipo de host dependiendo del objeto seleccionado.
opstatus	object	Objeto de estado de operación de anulación para objetos item, trigger y host prototype.
opdiscover	object	Objeto de estado de descubrimiento de operación de anulación (todos los tipos de objeto).
opperiod	object	Objeto de periodo de operación de anulación (intervalo de actualización) para el objeto item prototype.

Propiedad	Tipo	Descripción
ophistory	object	Objeto de historial de operación de anulación para el objeto item prototype.
optrends	object	Objeto de tendencias de operación de anulación para el objeto item prototype.
opseverity	object	Objeto de severidad de operación de anulación para el objeto trigger prototype.
optag	object/array	Objeto de etiqueta de operación de anulación para los objetos trigger y host prototype.
optemplate	object/array	Objeto de template de operación de anulación para el objeto host prototype.
opinventory	object	Objeto de inventario de operación de anulación para el objeto host prototype.

Estado de la operación de anulación de la regla LLD prototipo

Estado de la operación de anulación de la regla LLD prototipo que se establece en el objeto descubierto. Tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
status	integer	Anula el estado para el objeto seleccionado. Valores posibles: 0 - Crear habilitado; 1 - Crear deshabilitado. Comportamiento de la propiedad: - *obligatorio*

Operación de descubrimiento de prototipo de regla LLD de anulación

El estado de descubrimiento de la operación de anulación de prototipo de regla LLD que se establece en el objeto descubierto. Tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
discover	integer	Anula el estado de descubrimiento para el objeto seleccionado. Valores posibles: 0 - Sí, continuar descubriendo los objetos; 1 - No, no se descubrirán nuevos objetos y los existentes se marcarán como perdidos. Comportamiento de la propiedad: - *obligatorio*

Período de operación de la sobrescritura del prototipo de regla LLD

El período de operación de la sobrescritura del prototipo de regla LLD es un valor de intervalo de actualización que se establece en el item descubierto. Tiene las siguientes propiedades:

Property	Type	Description
delay	string	Sobrescribe el intervalo de actualización del prototipo de item. Acepta un número de segundos (por ejemplo, 1, 60, 300), un valor con un sufijo de tiempo (por ejemplo, 30s, 1m, 2h, 1d) o, opcionalmente, uno o varios intervalos personalizados, todos separados por punto y coma. Los intervalos personalizados pueden ser una combinación de intervalos flexibles y de planificación. Acepta macros de usuario o macros LLD. Si se utiliza, el valor debe ser una única macro. No se admiten varias macros ni macros combinadas con texto. Los intervalos flexibles pueden escribirse como dos macros separadas por una barra diagonal (por ejemplo, {$FLEX_INTERVAL}/{$FLEX_PERIOD}). Ejemplo: 1h;wd1-5h9-18;{$Macro1}/1-7,00:00-24:00;0/6-7,12:00-24:00;{$Macr Comportamiento de la propiedad: - *required*

Historial de operaciones de anulación de prototipos de reglas LLD

Valor del historial de operaciones de anulación de prototipos de reglas LLD que se establece en el item descubierto. Tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
history	string	Anula el historial del prototipo de item, que es una unidad de tiempo de cuánto tiempo se deben almacenar los datos históricos. También acepta macro de usuario y macro LLD. Comportamiento de la propiedad: - *obligatorio*

Tendencias de la operación de anulación de la regla LLD

Valor de tendencias de la operación de anulación de la regla LLD que se establece en el item descubierto. Tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
trends	string	Anula las tendencias del prototipo de item, que es una unidad de tiempo de cuánto tiempo deben almacenarse los datos de tendencias. También acepta macro de usuario y macro LLD. Comportamiento de la propiedad: - *requerido*

Severidad de la operación de anulación de la regla LLD

Valor de severidad de la operación de anulación de la regla LLD que se establece en el trigger descubierto. Tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
severity	integer	Anula la severidad del trigger prototype. Valores posibles: 0 - *(por defecto)* no clasificado; 1 - información; 2 - advertencia; 3 - media; 4 - alta; 5 - desastre. Comportamiento de la propiedad: - *requerido*

Etiqueta de operación de anulación de prototipo de regla LLD

El objeto de etiqueta de operación de anulación de prototipo de regla LLD contiene el nombre y el valor de la etiqueta que se establecen en el objeto descubierto. Tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
tag	string	Nuevo nombre de etiqueta. Comportamiento de la propiedad: - *requerido*
value	string	Nuevo valor de la etiqueta.

Plantilla de operación de anulación de la regla LLD

Objeto de plantilla de operación de anulación de la regla LLD que está vinculado al host descubierto. Tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
templateid	ID	Anula la plantilla de las plantillas vinculadas al prototipo de host. Comportamiento de la propiedad: - *obligatorio*

Inventario de la operación de anulación de la regla LLD

Valor del modo de inventario de la operación de anulación de la regla LLD que se establece en el host descubierto. Tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
inventory_mode	integer	Anula el modo de inventario del prototipo de host. Valores posibles: -1 - deshabilitado; 0 - *(por defecto)* manual; 1 - automático. Comportamiento de la propiedad: - *requerido*

discoveryruleprototype.create

Descripción

```
object discoveryruleprototype.create(object/array lldRules)
```

Este método le permite crear nuevos prototipos de reglas LLD.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object/array)` Prototipos de reglas LLD a crear.

Además de las propiedades estándar de los prototipos de reglas LLD, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
filter	object	Filtro de prototipo de regla LLD para la regla LLD.
preprocessing	object/array	Opciones de preprocesamiento de prototipo de regla LLD.
lld_macro_paths	object/array	Opciones de lld_macro_path del prototipo de regla LLD.
overrides	object/array	Opciones de anulaciones de prototipo de regla LLD.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las prototipos de reglas LLD creadas bajo la propiedad `itemids`. El orden de los IDs devueltos coincide con el orden de las prototipos de reglas LLD pasadas.

Ejemplos

Creando una prototipo de regla LLD

Cree una prototipo de regla LLD (Tipo: Anidado) para descubrir tablespaces en la instancia de base de datos.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryruleprototype.create",
    "params": {
        "name": "Descubrir tablespaces para {#DB}",
        "key_": "db.tablespace.discovery[{#DB}]",
        "hostid": "10084",
        "ruleid": "47251",
        "type": 23
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "47252"
        ]
    },
    "id": 1
}
```

Ver también

- Filtro de prototipo de regla LLD
- Rutas de macro LLD
- Preprocesamiento de prototipo de regla LLD

Fuente

CDiscoveryRulePrototype::create() en *ui/include/classes/api/services/CDiscoveryRulePrototype.php*.

discoveryruleprototype.delete

Descripción

`object discoveryruleprototype.delete(array lldRuleIds)`

Este método permite eliminar prototipos de reglas LLD.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(array)` IDs de las reglas prototipo de LLD a eliminar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las reglas LLD eliminadas bajo la propiedad `ruleids`.

Ejemplos

Eliminando múltiples prototipos de reglas LLD

Eliminar dos prototipos de reglas LLD.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryruleprototype.delete",
    "params": [
        "47252",
        "47253"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "ruleids": [
            "47252",
            "47253"
        ]
    },
    "id": 1
}
```

Fuente

CDiscoveryRulePrototype::delete() en *ui/include/classes/api/services/CDiscoveryRulePrototype.php*.

discoveryruleprototype.get

Descripción

`integer/array discoveryruleprototype.get(object parameters)`

El método permite recuperar prototipos de reglas LLD según los parámetros proporcionados.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(objeto)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
discoveryids	ID/array	Devolver solo prototipos de reglas LLD que pertenezcan a las reglas LLD o prototipos de reglas LLD indicados.
itemids	ID/array	Devolver solo prototipos de reglas LLD con los IDs indicados.
groupids	ID/array	Devolver solo prototipos de reglas LLD que pertenezcan a los hosts de los grupos indicados.
hostids	ID/array	Devolver solo prototipos de reglas LLD que pertenezcan a los hosts indicados.
inherited	boolean	Si se establece en `true`, devolver solo prototipos de reglas LLD heredados de un template.
interfaceids	ID/array	Devolver solo prototipos de reglas LLD que utilicen las interfaces de host indicadas.
monitored	boolean	Si se establece en `true`, devolver solo prototipos de reglas LLD habilitados que pertenezcan a hosts monitorizados.
templated	boolean	Si se establece en `true`, devolver solo prototipos de reglas LLD que pertenezcan a templates.
templateids	ID/array	Devolver solo prototipos de reglas LLD que pertenezcan a los templates indicados.
selectDiscoveryRule	query	Devolver una propiedad `discoveryRule` con la regla LLD padre del prototipo de regla LLD. Devolverá una respuesta vacía si el padre es un prototipo de regla de descubrimiento.
selectDiscoveryRulePrototype	query	Devolver una propiedad `discoveryRulePrototype` con el prototipo de regla LLD padre del prototipo de regla LLD. Devolverá una respuesta vacía si el padre es una regla de descubrimiento (no prototipo).
selectDiscoveryRulePrototypes	query	Devolver una propiedad `discoveryRulePrototypes` con los prototipos de reglas LLD hijos que pertenecen al prototipo de regla LLD. Admite `count`.
selectFilter	query	Devolver una propiedad `filter` con los datos del filtro utilizado por el prototipo de regla LLD.
selectGraphs	query	Devolver una propiedad `graphs` con los prototipos de gráficos que pertenecen al prototipo de regla LLD. Admite `count`.
selectHostPrototypes	query	Devolver una propiedad `hostPrototypes` con los prototipos de host que pertenecen al prototipo de regla LLD. Admite `count`.
selectHosts	query	Devolver una propiedad `hosts` con un array de hosts a los que pertenece el prototipo de regla LLD.
selectItems	query	Devolver una propiedad `items` con los prototipos de item que pertenecen al prototipo de regla LLD. Admite `count`.
selectTriggers	query	Devolver una propiedad `triggers` con los prototipos de trigger que pertenecen al prototipo de regla LLD. Admite `count`.
selectLLDMacroPaths	query	Devolver una propiedad `lld_macro_paths` con una lista de macros LLD y rutas a los valores asignados a cada macro correspondiente.
selectPreprocessing	query	Devolver una propiedad `preprocessing` con las opciones de preprocesamiento del prototipo de regla LLD.
selectOverrides	query	Devolver una propiedad `lld_rule_overrides` con una lista de filtros de override, condiciones y operaciones que se realizan sobre los objetos prototipo.

Parámetro	Tipo	Descripción
filter	object	Devolver solo aquellos resultados que coincidan exactamente con el filtro indicado. Acepta un objeto, donde las claves son nombres de propiedades y los valores son un único valor o un array de valores con los que comparar. No admite propiedades de tipo de dato `text` data type. Admite propiedades adicionales: `host` - nombre técnico del host al que pertenece el prototipo de regla LLD.
limitSelects	integer	Limita el número de registros devueltos por subconsultas. Se aplica a las siguientes subconsultas: `selectItems`, `selectGraphs`, `selectTriggers`.
sortfield	string/array	Ordenar el resultado por las propiedades indicadas. Valores posibles: `itemid`, `name`, `key_`, `delay`, `type`, `status`.
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
editable	boolean	
excludeSearch	boolean	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

`(entero/array)` devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperar prototipos de reglas de descubrimiento de un equipo

Recupera todos los prototipos de reglas de descubrimiento para un ID de equipo específico.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryruleprototype.get",
    "params": {
        "hostids": "10084"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "itemid": "47253",
            "type": "23",
            "snmp_oid": "",
            "hostid": "10084",
            "name": "Descubrir tablespaces para {#DB}",
```

```
            "key_": "db.tablespace.discovery[{#DB}]",
            "delay": "0",
            "history": "31d",
            "trends": "365d",
            "status": "0",
            "value_type": "4",
            "trapper_hosts": "",
            "units": "",
            "logtimefmt": "",
            "templateid": "0",
            "valuemapid": "0",
            "params": "",
            "ipmi_sensor": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "flags": "3",
            "interfaceid": "0",
            "description": "",
            "inventory_link": "0",
            "lifetime": "7d",
            "jmx_endpoint": "",
            "master_itemid": "0",
            "timeout": "",
            "url": "",
            "query_fields": [],
            "posts": "",
            "status_codes": "200",
            "follow_redirects": "1",
            "post_type": "0",
            "http_proxy": "",
            "headers": [],
            "retrieve_mode": "0",
            "request_method": "0",
            "output_format": "0",
            "ssl_cert_file": "",
            "ssl_key_file": "",
            "ssl_key_password": "",
            "verify_peer": "0",
            "verify_host": "0",
            "allow_traps": "0",
            "discover": "0",
            "uuid": "",
            "lifetime_type": "0",
            "enabled_lifetime_type": "2",
            "enabled_lifetime": "0",
            "parameters": []
        }
    ],
    "id": 1
}
```


Ver también

- Prototipo de gráfico
- Host
- Prototipo de item
- Filtro de prototipo de regla LLD
- Prototipo de trigger

Fuente

CDiscoveryRulePrototype::get() en *ui/include/classes/api/services/CDiscoveryRulePrototype.php*.

discoveryruleprototype.update

Descripción

`object discoveryruleprototype.update(object/array lldRules)`

Este método le permite actualizar prototipos de reglas LLD existentes.

Tenga en cuenta que la actualización de prototipos ya descubiertos es limitada.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object/array)` Propiedades de la regla LLD a actualizar.

La propiedad `itemid` debe estar definida para cada regla LLD, todas las demás propiedades son opcionales. Sólo se actualizarán las propiedades pasadas, todas las demás permanecerán sin cambios.

Además de las propiedades estándar de la regla LLD, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
filter	object	Filtro de la regla LLD para reemplazar el filtro existente.
preprocessing	object/array	Opciones de preprocesamiento de la regla LLD para reemplazar las opciones de preprocesamiento existentes. Comportamiento del parámetro: - *sólo lectura* para objetos heredados
lld_macro_paths	object/array	Opciones de lld_macro_path de la regla LLD para reemplazar las opciones de lld_macro_path existentes. Comportamiento del parámetro: - *sólo lectura* para objetos heredados
overrides	object/array	Opciones de overrides de la regla LLD para reemplazar las opciones de overrides existentes. Comportamiento del parámetro: - *sólo lectura* para objetos heredados

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las reglas LLD actualizadas bajo la propiedad `itemids`.

Ejemplos

Actualización de las opciones de preprocesamiento de la regla prototipo de LLD

Actualice una regla prototipo de LLD con una regla de preprocesamiento JSONPath. Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryruleprototype.update",
    "params": {
        "itemid": "47253",
        "preprocessing": [
            {
                "type": 12,
                "params": "$.tablespaces",
                "error_handler": 1
            }
        ]
    },
```

```
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "47253"
        ]
    },
    "id": 1
}
```

Fuente

CDiscoveryRulePrototype::update() en *ui/include/classes/api/services/CDiscoveryRulePrototype.php*.

Prototipo de trigger

Esta clase está diseñada para trabajar con prototipos de trigger.

Referencias de objetos:

- Prototipo de trigger
- Etiqueta de prototipo de trigger

Métodos disponibles:

- triggerprototype.create - crear nuevos prototipos de trigger
- triggerprototype.delete - eliminar prototipos de trigger
- triggerprototype.get - obtener prototipos de trigger
- triggerprototype.update - actualizar prototipos de trigger

Objeto prototipo de trigger

Los siguientes objetos están directamente relacionados con la API `triggerprototype`.

Prototipo de trigger

El objeto de prototipo de trigger tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
triggerid	ID	ID del prototipo de trigger. Comportamiento de la propiedad: - *solo lectura* - *requerido* para operaciones de actualización
description	string	Nombre del prototipo de trigger. Comportamiento de la propiedad: - *requerido* para operaciones de creación
expression	string	Expresión del trigger. Debe contener al menos un prototipo de item. Los prototipos de item con `value_type` establecido en 5 (binario) o 6 (JSON) no pueden usarse en expresiones de trigger. Comportamiento de la propiedad: - *requerido* para operaciones de creación
event_name	string	Nombre del evento generado por el trigger.
opdata	string	Datos operativos.
comments	string	Comentarios adicionales al prototipo de trigger.

Propiedad	Tipo	Descripción
priority	integer	Severidad del prototipo de trigger. Valores posibles: 0 - *(por defecto)* no clasificado; 1 - información; 2 - advertencia; 3 - media; 4 - alta; 5 - desastre.
status	integer	Indica si el prototipo de trigger está habilitado o deshabilitado. Valores posibles: 0 - *(por defecto)* habilitado; 1 - deshabilitado.
flags	integer	Origen del prototipo de trigger. Valores posibles: 2 - un prototipo de trigger; 6 - un prototipo de trigger descubierto. Comportamiento de la propiedad: - *solo lectura*
templateid	ID	ID del prototipo de trigger de la template padre. Comportamiento de la propiedad: - *solo lectura*
type	integer	Indica si el prototipo de trigger puede generar múltiples eventos de problema. Valores posibles: 0 - *(por defecto)* no generar múltiples eventos; 1 - generar múltiples eventos.
url	string	URL asociada al prototipo de trigger.
url_name	string	Etiqueta para la URL asociada al prototipo de trigger.
recovery_mode	integer	Modo de generación de evento OK. Valores posibles: 0 - *(por defecto)* Expresión; 1 - Expresión de recuperación; 2 - Ninguno.
recovery_expression	string	Expresión de recuperación del trigger. Debe contener al menos un prototipo de item. Los prototipos de item con `value_type` establecido en 5 (binario) o 6 (JSON) no pueden usarse en expresiones de recuperación de trigger.
correlation_mode	integer	El evento OK cierra. Valores posibles: 0 - *(por defecto)* Todos los problemas; 1 - Todos los problemas si los valores de las etiquetas coinciden.
correlation_tag	string	Etiqueta para coincidencia.
manual_close	integer	Permitir cierre manual. Valores posibles: 0 - *(por defecto)* No; 1 - Sí.
discover	integer	Estado de descubrimiento del prototipo de trigger. Valores posibles: 0 - *(por defecto)* se descubrirán nuevos triggers; 1 - no se descubrirán nuevos triggers y los triggers existentes se marcarán como perdidos.

Propiedad	Tipo	Descripción
uuid	string	Identificador único universal, utilizado para vincular prototipos de trigger importados con los ya existentes. Se genera automáticamente si no se proporciona. Comportamiento de la propiedad: - *soportado* si el prototipo de trigger pertenece a una template

Etiqueta de prototipo de trigger

El objeto de etiqueta de prototipo de trigger tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
tag	string	Nombre de la etiqueta del prototipo de trigger. Comportamiento de la propiedad: - *requerido*
value	string	Valor de la etiqueta del prototipo de trigger. Comportamiento de la propiedad: - *solo lectura*.
object	integer	Tipo de objeto del que se heredó la etiqueta. Valores posibles: 0 - template; 1 - host; 3 - item; 4 - prototipo de item. Comportamiento de la propiedad: - *solo lectura*.
objectid	ID	ID del objeto del que se heredó la etiqueta. Comportamiento de la propiedad: - *solo lectura*.

triggerprototype.create

Descripción

`object triggerprototype.create(object/array triggerPrototypes)`

Este método le permite crear nuevos prototipos de trigger.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(objeto/array)` Prototipos de trigger a crear.

Además de las propiedades estándar de los prototipos de trigger, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
dependencies	array	Triggers y prototipos de trigger de los que depende el prototipo de trigger. Los triggers deben tener definida únicamente la propiedad `triggerid`.
tags	array	Etiquetas de prototipo de trigger.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las prototipos de disparadores creados bajo la propiedad `triggerids`. El orden de los IDs devueltos coincide con el orden de las prototipos de disparadores pasadas.

Ejemplos

Creando un prototipo de disparador

Cree un prototipo de disparador para detectar cuando un sistema de archivos tiene menos del 20% de espacio libre en disco.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "triggerprototype.create",
    "params": {
        "description": "El espacio libre en disco es menor al 20% en el volumen {#FSNAME}",
        "expression": "last(/Zabbix server/vfs.fs.size[{#FSNAME},pfree])<20",
        "tags": [
            {
                "tag": "volume",
                "value": "{#FSNAME}"
            },
            {
                "tag": "type",
                "value": "{#FSTYPE}"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "triggerids": [
            "17372"
        ]
    },
    "id": 1
}
```

Fuente

CTriggerPrototype::create() en *ui/include/classes/api/services/CTriggerPrototype.php*.

triggerprototype.delete

Descripción

`object triggerprototype.delete(array triggerPrototypeIds)`

Este método permite eliminar prototipos de trigger.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(array)` IDs de las prototipos de disparadores a eliminar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las prototipos de disparadores eliminadas bajo la propiedad `triggerids`.

Ejemplos

Eliminando múltiples prototipos de disparadores

Eliminar dos prototipos de disparadores.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "triggerprototype.delete",
    "params": [
        "12002",
        "12003"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "triggerids": [
            "12002",
            "12003"
        ]
    },
    "id": 1
}
```

Fuente

CTriggerPrototype::delete() en *ui/include/classes/api/services/CTriggerPrototype.php*.

triggerprototype.get

Descripción

`integer/array triggerprototype.get(object parameters)`

El método permite recuperar prototipos de trigger según los parámetros proporcionados.

Note:
Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
active	flag	Devolver solo prototipos de trigger habilitados que pertenezcan a hosts monitorizados.
discoveryids	ID/array	Devolver solo prototipos de trigger que pertenezcan a las reglas LLD indicadas.
functions	string/array	Devolver solo triggers que utilicen las funciones indicadas. Consulte la página de Funciones soportadas para ver la lista de funciones soportadas.
group	string	Devolver solo prototipos de trigger que pertenezcan a hosts o templates de los grupos de hosts o grupos de templates con el nombre indicado.
groupids	ID/array	Devolver solo prototipos de trigger que pertenezcan a hosts o templates de los grupos de hosts o grupos de templates indicados.
host	string	Devolver solo prototipos de trigger que pertenezcan a hosts con el nombre indicado.

Parámetro	Tipo	Descripción
hostids	ID/array	Devolver solo prototipos de trigger que pertenezcan a los hosts indicados.
inherited	boolean	Si se establece en `true`, devolver solo prototipos de trigger heredados de un template.
maintenance	boolean	Si se establece en `true`, devolver solo prototipos de trigger habilitados que pertenezcan a hosts en mantenimiento.
min_severity	integer	Devolver solo prototipos de trigger con una severidad mayor o igual a la severidad indicada.
monitored	flag	Devolver solo prototipos de trigger habilitados que pertenezcan a hosts monitorizados y contengan solo items habilitados.
templated	boolean	Si se establece en `true`, devolver solo prototipos de trigger que pertenezcan a templates.
templateids	ID/array	Devolver solo prototipos de trigger que pertenezcan a los templates indicados.
triggerids	ID/array	Devolver solo prototipos de trigger con los IDs indicados.
expandExpression	flag	Expandir funciones y macros en la expresión del trigger.
selectDependencies	query	Devolver prototipos de trigger y triggers de los que depende el prototipo de trigger en la propiedad `dependencies`.
selectDiscoveryData	query	Devolver una propiedad `discoveryData` con los datos del objeto de descubrimiento del prototipo de trigger. El objeto de descubrimiento del prototipo de trigger vincula un prototipo de trigger descubierto a un prototipo de trigger del que se descubrió. Tiene las siguientes propiedades: `parent_triggerid` - `(ID)` ID del prototipo de trigger del que se ha creado el prototipo de trigger; `status` - `(int)` estado de descubrimiento del prototipo de trigger: 0 - *(por defecto)* el prototipo de trigger está descubierto, 1 - el prototipo de trigger ya no está descubierto; `ts_delete` - `(timestamp)` hora en la que se eliminará un prototipo de trigger que ya no se descubre; `ts_disable` - `(timestamp)` hora en la que se deshabilitará un prototipo de trigger que ya no se descubre; `disable_source` - `(int)` indicador de si el prototipo de trigger fue deshabilitado por una regla LLD o manualmente: 0 - *(por defecto)* deshabilitado automáticamente, 1 - deshabilitado por una regla LLD.
selectDiscoveryRule	query	Devolver la regla LLD a la que pertenece el prototipo de trigger en la propiedad `discoveryRule`.
selectDiscoveryRulePrototype	query	Devolver una propiedad `discoveryRulePrototype` con el prototipo de regla LLD principal al que pertenece el prototipo de trigger.
selectFunctions	query	Devolver las funciones utilizadas en el prototipo de trigger en la propiedad `functions`. Los objetos de función representan las funciones utilizadas en la expresión del trigger y tienen las siguientes propiedades: `functionid` - `(ID)` ID de la función; `itemid` - `(ID)` ID del item utilizado en la función; `function` - `(string)` nombre de la función; `parameter` - `(string)` parámetro pasado a la función. El parámetro de consulta se reemplaza por el símbolo $ en la cadena devuelta.
selectHostGroups	query	Devolver los grupos de hosts a los que pertenece el prototipo de trigger en la propiedad `hostgroups`.
selectHosts	query	Devolver los hosts a los que pertenece el prototipo de trigger en la propiedad `hosts`.
selectInheritedTags	query	Devolver una propiedad `inheritedTags` con las etiquetas heredadas de templates, templates vinculados y hosts, y de prototipos de item referenciados en expresiones de trigger o de recuperación.
selectItems	query	Devolver los items y prototipos de item utilizados por el prototipo de trigger en la propiedad `items`.
selectTags	query	Devolver las etiquetas del prototipo de trigger en la propiedad `tags`.

Parámetro	Tipo	Descripción
selectTemplateGroups	query	Devolver los grupos de templates a los que pertenece el prototipo de trigger en la propiedad `templategroups`.
filter	object	Devolver solo aquellos resultados que coincidan exactamente con el filtro indicado. Acepta un objeto, donde las claves son nombres de propiedades y los valores son un único valor o un array de valores con los que comparar. No admite propiedades del tipo de datos `text` data type. Admite propiedades adicionales: `host` - nombre técnico del host al que pertenece el prototipo de trigger; `hostid` - ID del host al que pertenece el prototipo de trigger.
limitSelects	integer	Limita el número de registros devueltos por subconsultas. Se aplica a las siguientes subconsultas: `selectHosts` - los resultados se ordenarán por `host`.
sortfield	string/array	Ordenar el resultado por las propiedades indicadas. Valores posibles: `triggerid`, `description`, `status`, `priority`, `discovered`.
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
editable	boolean	
excludeSearch	boolean	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperar prototipos de trigger de una regla LLD

Recuperar todos los prototipos de trigger y sus funciones de una regla LLD.

Request:

```
{
    "jsonrpc": "2.0",
    "method": "triggerprototype.get",
    "params": {
        "output": "extend",
        "selectFunctions": "extend",
        "discoveryids": "22450"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
```

```
        {
            "triggerid": "13272",
            "expression": "{12598}<20",
            "description": "Free inodes is less than 20% on volume {#FSNAME}",
            "url": "",
            "status": "0",
            "value": "0",
            "priority": "2",
            "lastchange": "0",
            "comments": "",
            "error": "",
            "templateid": "0",
            "type": "0",
            "state": "0",
            "flags": "2",
            "recovery_mode": "0",
            "recovery_expression": "",
            "correlation_mode": "0",
            "correlation_tag": "",
            "manual_close": "0",
            "opdata": "",
            "discover": "0",
            "event_name": "",
            "uuid": "6ce467d05e8745409a177799bed34bb3",
            "url_name": "",
            "functions": [
                {
                    "functionid": "12598",
                    "itemid": "22454",
                    "triggerid": "13272",
                    "parameter": "$",
                    "function": "last"
                }
            ]
        },
        {
            "triggerid": "13266",
            "expression": "{13500}<20",
            "description": "Free disk space is less than 20% on volume {#FSNAME}",
            "url": "",
            "status": "0",
            "value": "0",
            "priority": "2",
            "lastchange": "0",
            "comments": "",
            "error": "",
            "templateid": "0",
            "type": "0",
            "state": "0",
            "flags": "2",
            "recovery_mode": "0",
            "recovery_expression": "",
            "correlation_mode": "0",
            "correlation_tag": "",
            "manual_close": "0",
            "opdata": "",
            "discover": "0",
            "event_name": "",
            "uuid": "74a1fc62bfe24b7eabe4e244c70dc384",
            "url_name": "",
            "functions": [
                {
```

```
                    "functionid": "13500",
                    "itemid": "22686",
                    "triggerid": "13266",
                    "parameter": "$",
                    "function": "last"
                }
            ]
        }
    ],
    "id": 1
}
```

Recuperar una prototipo de disparador específica con etiquetas

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "triggerprototype.get",
    "params": {
        "output": [
            "triggerid",
            "description"
        ],
        "selectTags": "extend",
        "triggerids": [
            "17373"
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "triggerid": "17373",
            "description": "El espacio libre en disco es menor al 20% en el volumen {#FSNAME}",
            "tags": [
                {
                    "tag": "volume",
                    "value": "{#FSNAME}"
                },
                {
                    "tag": "type",
                    "value": "{#FSTYPE}"
                }
            ]
        }
    ],
    "id": 1
}
```

Ver también

- Regla de descubrimiento
- Item
- Host
- Grupo de hosts
- Grupo de templates

Fuente

CTriggerPrototype::get() en *ui/include/classes/api/services/CTriggerPrototype.php*.

triggerprototype.update

Descripción

`object triggerprototype.update(object/array triggerPrototypes)`

Este método le permite actualizar prototipos de trigger existentes.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object/array)` Propiedades de la trigger prototype que se actualizarán.

La propiedad `triggerid` debe estar definida para cada trigger prototype, todas las demás propiedades son opcionales. Sólo se actualizarán las propiedades pasadas, todas las demás permanecerán sin cambios.

Además de las propiedades estándar de trigger prototype, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
dependencies	array	Triggers y trigger prototypes de las que depende la trigger prototype. Las triggers deben tener definida sólo la propiedad `triggerid`.
tags	array	Etiquetas de trigger prototype.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las prototipos de disparadores actualizadas bajo la propiedad `triggerids`.

Ejemplos

Habilitar una prototipo de disparador

Habilite una prototipo de disparador, es decir, establezca su estado en ”0”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "triggerprototype.update",
    "params": {
        "triggerid": "13938",
        "status": 0
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "triggerids": [
            "13938"
        ]
    },
    "id": 1
}
```

Reemplazar etiquetas de prototipo de disparador

Reemplazar etiquetas para un prototipo de disparador.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "triggerprototype.update",
    "params": {
        "triggerid": "17373",
        "tags": [
            {
                "tag": "volume",
                "value": "{#FSNAME}"
            },
            {
                "tag": "type",
                "value": "{#FSTYPE}"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "triggerids": [
            "17373"
        ]
    },
    "id": 1
}
```

Fuente

CTriggerPrototype::update() en *ui/include/classes/api/services/CTriggerPrototype.php*.

Proxy

Esta clase está diseñada para trabajar con proxies.

Referencias de objetos:

- Proxy

Métodos disponibles:

- proxy.create - crear nuevos proxies
- proxy.delete - eliminar proxies
- proxy.get - obtener proxies
- proxy.update - actualizar proxies

Objeto proxy

Los siguientes objetos están directamente relacionados con la API `proxy`.

Proxy

El objeto proxy tiene las siguientes propiedades.

Property	Type	Description
proxyid	ID	ID del proxy. Comportamiento de la propiedad: - *solo lectura* - *obligatoria* para las operaciones de actualización

Property	Type	Description
name	string	Nombre del proxy. Comportamiento de la propiedad: - *obligatoria* para las operaciones de creación
proxy_groupid	ID	ID del grupo de proxies. 0 - el proxy no está asignado a ningún grupo de proxies.
local_address	string	Dirección para los agentes activos. Dirección IP o nombre DNS al que conectarse. Comportamiento de la propiedad: - *obligatoria* si `proxy_groupid` no es 0
local_port	string	Número de puerto local del proxy al que conectarse. Se admiten macros de usuario. Predeterminado: `10051`. Comportamiento de la propiedad: - *admitida* si `proxy_groupid` no es 0
operating_mode	integer	Tipo de proxy. Valores posibles: 0 - proxy activo; 1 - proxy pasivo. Comportamiento de la propiedad: - *obligatoria* para las operaciones de creación
description	text	Descripción del proxy.
lastaccess	timestamp	Momento en que el proxy se conectó por última vez al server. Comportamiento de la propiedad: - *solo lectura*
address	string	Dirección IP o nombre DNS al que conectarse. Se admiten macros de usuario. Comportamiento de la propiedad: - *obligatoria* si el modo de funcionamiento del proxy de Zabbix es pasivo
port	string	Número de puerto al que conectarse. Se admiten macros de usuario. Predeterminado: `10051`. Comportamiento de la propiedad: - *admitida* si el modo de funcionamiento del proxy de Zabbix es pasivo
allowed_addresses	string	Direcciones IP o nombres DNS delimitados por comas del proxy activo de Zabbix.
tls_connect	integer	Conexiones al proxy. Valores posibles: 1 - *(predeterminado)* sin cifrado; 2 - PSK; 4 - certificado.

Property	Type	Description
tls_accept	integer	Conexiones desde el proxy. Valores posibles de mapa de bits: 1 - *(predeterminado)* sin cifrado; 2 - PSK; 4 - certificado. Este es un campo de máscara de bits; se acepta cualquier suma de los valores posibles del mapa de bits (por ejemplo, 6 para PSK y certificado).
tls_issuer	string	Emisor del certificado.
tls_subject	string	Asunto del certificado.
tls_psk_identity	string	Identidad PSK; debe estar asociada únicamente a una PSK (entre autoregistro, hosts y proxies). No incluya información confidencial en la identidad PSK, ya que se envía sin cifrar a través de la red para informar al receptor de qué PSK debe utilizar. Comportamiento de la propiedad: - *solo escritura* - *obligatoria* si `tls_connect` está establecido en ”PSK” o si `tls_accept` contiene el bit ”PSK”
tls_psk	string	Clave precompartida (PSK); debe tener al menos 32 dígitos hexadecimales. Comportamiento de la propiedad: - *solo escritura* - *obligatoria* si `tls_connect` está establecido en ”PSK” o si `tls_accept` contiene el bit ”PSK”
custom_timeouts	integer	Indica si se deben sobrescribir los tiempos de espera globales de los item en el nivel del proxy. Valores posibles: 0 - *(predeterminado)* utilizar la configuración global; 1 - sobrescribir los tiempos de espera.
timeout_zabbix_agent	string	No dedicar más de `timeout_*` segundos al procesamiento. Acepta un número de segundos (por ejemplo, 1, 60, 300) o un valor con un sufijo de tiempo (por ejemplo, 30s, 1m). También acepta macros de usuario. Se utiliza únicamente si `custom_timeouts` está establecido en ”override timeouts”; de lo contrario, el valor se ignora. Rango de valores posibles: 1-600s. Comportamiento de la propiedad: - *obligatoria* si `custom_timeouts` está establecido en ”override timeouts”
timeout_simple_check		
timeout_snmp_agent		
timeout_external_check		
timeout_db_monitor		
timeout_http_agent		
timeout_ssh_agent		
timeout_telnet_agent		
timeout_script		
timeout_browser		
timeout_telemetry_query		

Property	Type	Description
version	integer	Versión del proxy. Número de versión de Zabbix de tres partes, donde se utilizan dos dígitos decimales para cada parte; por ejemplo, 60401 para la versión 6.4.1, 70002 para la versión 7.0.2, etc. 0 - Versión del proxy desconocida. Comportamiento de la propiedad: - *solo lectura*
compatibility	integer	Versión del proxy en comparación con la versión del server de Zabbix. Valores posibles: 0 - No definida; 1 - Versión actual (el proxy y el server tienen la misma versión principal); 2 - Versión obsoleta (la versión del proxy es anterior a la del server, pero es parcialmente compatible); 3 - Versión no compatible (la versión del proxy es anterior a la versión de la versión LTS anterior del server o la versión principal del server es anterior a la versión principal del proxy). Comportamiento de la propiedad: - *solo lectura*
state	integer	Estado del proxy. Valores posibles: 0 - Desconocido; 1 - Fuera de línea; 2 - En línea. Comportamiento de la propiedad: - *solo lectura*

Supervisión del rendimiento de las aplicaciones (APM)

El objeto APM tiene las siguientes propiedades.

Property	Type	Description
data_collection_status	integer	Indica si la recopilación de datos de telemetría está habilitada en el proxy. Valores posibles: 0 - *(predeterminado)* Deshabilitada; 1 - Habilitada.
max_messages_per_second	integer	Número máximo de mensajes por segundo que acepta el proxy. Rango de valores posibles: 0 (ilimitado) – 2147483647. Valor predeterminado: 0. Comportamiento de la propiedad: - *supported* si `data_collection_status` está establecido en ”Habilitada”

Property	Type	Description
additional_resource_attributes	array	Matriz de atributos de recursos que se añadirán a los datos de telemetría entrantes. Si los datos entrantes ya contienen un atributo de recurso con el mismo nombre, se conserva el atributo entrante en lugar del especificado aquí. Comportamiento de la propiedad: - *supported* si `data_collection_status` está establecido en "Habilitada"

Atributos del recurso

El objeto de atributos del recurso tiene las siguientes propiedades.

Property	Type	Description
key	string	Nombre del atributo del recurso. Debe ser único para el `signal_type` especificado. Comportamiento de la propiedad: - *required*
value	string	Valor del atributo del recurso. Puede estar vacío.
signal_type	integer	Tipo de datos de telemetría al que se aplica el atributo. Valores posibles: 0 - Trazas; 1 - Métricas; 2 - Registros. Comportamiento de la propiedad: - *required*

proxy.create

Descripción

`object proxy.create(object/array proxies)`

Este método permite crear nuevos proxies.

Note:
Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración del rol de usuario. Consulte User roles para obtener más información.

Parámetros

`(object/array)` Proxies que se crearán.

Además de las propiedades estándar del proxy, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
hosts	array	Hosts que serán supervisados por el proxy. Si un host ya está siendo supervisado por otro proxy, se reasignará al proxy actual. Los hosts solo deben tener definida la propiedad `hostid`.
apm	object	Configuración de supervisión del rendimiento de las aplicaciones (APM) del proxy.

Valores de retorno

(object) Devuelve un objeto que contiene los IDs de los proxies creados bajo la propiedad `proxyids`. El orden de los IDs devueltos coincide con el orden de los proxies enviados.

Ejemplos

Crear un proxy activo

Cree un proxy de acción "Proxy activo" y asigne un equipo para que sea monitorizado por él.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "proxy.create",
    "params": {
        "name": "Active proxy",
        "operating_mode": "0",
        "hosts": [
            {
                "hostid": "10279"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxyids": [
            "10280"
        ]
    },
    "id": 1
}
```

Crear un proxy pasivo

Cree un proxy pasivo "Proxy pasivo" y asigne dos equipos para que sean monitorizados por él.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "proxy.create",
    "params": {
        "name": "Proxy pasivo",
        "operating_mode": "1",
        "address": "127.0.0.1",
        "port": "10051",
        "hosts": [
            {
                "hostid": "10192"
            },
            {
                "hostid": "10139"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
```

```
        "proxyids": [
            "10284"
        ]
    },
    "id": 1
}
```

Creación de un proxy y adición a un grupo de proxies

Cree un proxy activo "Active proxy" y agréguelo al grupo de proxies con ID "1".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "proxy.create",
    "params": {
        "name": "Active proxy",
        "proxy_groupid": "1",
        "operating_mode": "0"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxyids": [
            "5"
        ]
    },
    "id": 1
}
```

Creación de un proxy con cifrado basado en certificado

Cree un proxy activo con cifrado basado en certificado. El proxy está configurado para supervisar el host "10785".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "proxy.create",
    "params": {
        "name": "Active proxy",
        "operating_mode": "0",
        "hosts": [
            {
                "hostid": "10785"
            }
        ],
        "tls_accept": "4",
        "tls_issuer": "CN=Signing CA,OU=Development group,O=Zabbix SIA,DC=zabbix,DC=com",
        "tls_subject": "CN=Zabbix server,OU=Development group,O=Zabbix SIA,DC=zabbix,DC=com"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxyids": [
            "2"
        ]
```

```
    },
    "id": 1
}
```

Creación de un proxy con APM habilitado

Cree un proxy activo "Active proxy" con APM habilitado.
El proxy acepta hasta 1000 mensajes por segundo, añade el atributo de recurso `deployment.environment` a las trazas y añade el atributo de recurso `cloud.region` a las métricas.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "proxy.create",
    "params": {
        "name": "Active proxy",
        "operating_mode": "0",
        "apm": {
            "data_collection_status": "1",
            "max_messages_per_second": "1000",
            "additional_resource_attributes": [
                {
                    "key": "deployment.environment",
                    "value": "production",
                    "signal_type": "0"
                },
                {
                    "key": "cloud.region",
                    "value": "eu-central-1",
                    "signal_type": "1"
                }
            ]
        }
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxyids": [
            "3"
        ]
    },
    "id": 1
}
```

Ver también

- Host
- Grupo de proxy

Fuente

CProxy::create() en *ui/include/classes/api/services/CProxy.php*.

proxy.delete

Descripción

`object proxy.delete(array proxies)`

Este método permite eliminar proxies.

Note:
Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración del rol de usuario. Consulte User roles para obtener más información.

Parámetros

`(array)` IDs de los proxies a eliminar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los proxies eliminados bajo la propiedad `proxyids`.

Ejemplos

Eliminar múltiples proxies

Eliminar dos proxies.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "proxy.delete",
    "params": [
        "10286",
        "10285"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxyids": [
            "10286",
            "10285"
        ]
    },
    "id": 1
}
```

Origen

CProxy::delete() en *ui/include/classes/api/services/CProxy.php*.

proxy.get

Descripción

`integer/array proxy.get(object parameters)`

El método permite recuperar proxies según los parámetros especificados.

Note:
Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración del rol de usuario. Consulte Roles de usuario para obtener más información.
 Solo se devuelven los proxies permitidos por la *Lista de acceso de proxy* de los grupos de usuarios del usuario.

Parámetros

`(object)` Parámetros que definen el resultado deseado.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
proxyids	ID/array	Devuelve solo los proxy con los ID indicados.

Parámetro	Tipo	Descripción
proxy_groupids	ID/array	Devuelve solo los proxy que pertenecen a los grupos de proxy indicados.
selectApm	query	Devuelve una propiedad `apm` con la configuración de monitorización del rendimiento de las aplicaciones (APM) del proxy.
selectAssignedHosts	query	Devuelve una propiedad `assignedHosts` con los host asignados al proxy. Admite `count`.
selectHosts	query	Devuelve una propiedad `hosts` con los host monitorizados por el proxy. Admite `count`.
selectProxyGroup	query	Devuelve una propiedad `proxyGroup` con el objeto del grupo de proxy.
sortfield	string/array	Ordena el resultado por las propiedades indicadas. Valores posibles: `proxyid`, `name`, `operating_mode`.
countOutput	boolean	Estos parámetros se describen en los comentarios de referencia.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperar todos los proxies

Recuperar todos los proxies configurados y sus interfaces.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "proxy.get",
    "params": {
        "output": "extend"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "proxyid": "11",
            "name": "Active proxy",
            "proxy_groupid": "0",
            "local_address": "",
            "local_port": "10051",
            "operating_mode": "0",
```

```
            "description": "",
            "allowed_addresses": "",
            "address": "127.0.0.1",
            "port": "10051",
            "tls_connect": "1",
            "tls_accept": "1",
            "tls_issuer": "",
            "tls_subject": "",
            "custom_timeouts": "0",
            "timeout_zabbix_agent": "",
            "timeout_simple_check": "",
            "timeout_snmp_agent": "",
            "timeout_external_check": "",
            "timeout_db_monitor": "",
            "timeout_http_agent": "",
            "timeout_ssh_agent": "",
            "timeout_telnet_agent": "",
            "timeout_script": "",
            "last_access": "1693391880",
            "version": "70000",
            "compatibility": "1",
            "state": "1"
        },
        {
            "proxyid": "12",
            "name": "Passive proxy",
            "proxy_groupid": "1",
            "local_address": "127.0.0.1",
            "local_port": "10051",
            "operating_mode": "1",
            "description": "",
            "allowed_addresses": "",
            "address": "127.0.0.1",
            "port": "10051",
            "tls_connect": "1",
            "tls_accept": "1",
            "tls_issuer": "",
            "tls_subject": "",
            "custom_timeouts": "1",
            "timeout_zabbix_agent": "5s",
            "timeout_simple_check": "5s",
            "timeout_snmp_agent": "5s",
            "timeout_external_check": "5s",
            "timeout_db_monitor": "5s",
            "timeout_http_agent": "5s",
            "timeout_ssh_agent": "5s",
            "timeout_telnet_agent": "5s",
            "timeout_script": "5s",
            "lastaccess": "1693391875",
            "version": "60400",
            "compatibility": "2",
            "state": "2"
        }
    ],
    "id": 1
}
```


Ver también

- Host
- Grupo de proxy

Origen

CProxy::get() en *ui/include/classes/api/services/CProxy.php*.

proxy.update

Descripción

`object proxy.update(object/array proxies)`

Este método le permite actualizar proxies existentes.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración del rol de usuario. Consulte User roles para obtener más información.

Parámetros

`(object/array)` Propiedades del proxy que se actualizarán.

La propiedad `proxyid` debe definirse para cada proxy; todas las demás propiedades son opcionales. Solo se actualizarán las propiedades proporcionadas; las demás permanecerán sin cambios.

Además de las propiedades estándar del proxy, el método acepta los siguientes parámetros.

Parameter	Type	Description
hosts	array	Hosts que serán supervisados por el proxy. Si un host ya está siendo supervisado por otro proxy, se reasignará al proxy actual. Los hosts solo deben tener definida la propiedad `hostid`.
apm	object	Configuración de supervisión del rendimiento de las aplicaciones (APM) del proxy.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los proxies actualizados bajo la propiedad `proxyids`.

Ejemplos

Cambiar los equipos monitorizados por un proxy

Actualice el proxy para monitorizar los dos equipos indicados.

Petición:

```
{
    "jsonrpc": "2.0",
    "method": "proxy.update",
    "params": {
        "proxyid": "10293",
        "hosts": [
            {
                "hostid": "10294"
            },
            {
                "hostid": "10295"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxyids": [
```

```
            "10293"
        ]
    },
    "id": 1
}
```

Cambiar el estado del proxy

Cambiar el proxy a un proxy activo y renombrarlo a "Active proxy".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "proxy.update",
    "params": {
        "proxyid": "10293",
        "name": "Active proxy",
        "operating_mode": "0"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxyids": [
            "10293"
        ]
    },
    "id": 1
}
```

Añadir un proxy a un grupo de proxies

Actualizar el proxy con ID "5" y añadirlo al grupo de proxies con ID "1".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "proxy.create",
    "params": {
        "proxyid": "5",
        "proxy_groupid": "1",
        "local_address": "127.0.0.1"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxyids": [
            "5"
        ]
    },
    "id": 1
}
```

Ver también

- Host
- Grupo de proxy

Origen

CProxy::update() en *ui/include/classes/api/services/CProxy.php*.

Registro de APM

Esta clase está diseñada para trabajar con registros recopilados para la monitorización del rendimiento de aplicaciones (APM).

Referencias de objetos:

- Registro de APM

Métodos disponibles:

- apmlog.get: recuperar registros de log

Objeto de registro de APM

Los siguientes objetos están directamente relacionados con la API `apmlog`.

Note:
Cada propiedad corresponde a la columna con el mismo nombre en la tabla `otel_logs` de la fuente de datos de telemetría (por ejemplo, la propiedad `severity_text` corresponde a la columna `SeverityText`). Para consultar la lista de columnas, consulte Columnas de datos de telemetría compatibles.

Registro de APM

El objeto de registro de APM representa un registro de eventos (un registro de texto de un evento, como un pago fallido o un tiempo de espera agotado de una base de datos). Un registro de eventos contiene información sobre un evento ocurrido en un momento específico, incluida su gravedad, mensaje y atributos. Si el registro de eventos se produjo durante una operación que forma parte de un trace, también contiene los ID de ese trace y span.

El objeto de registro de APM tiene las siguientes propiedades:

Property	Type	Description
timestamp	integer	Hora Unix (en nanosegundos) en la que ocurrió el evento. Comportamiento de la propiedad: - *solo lectura*
traceid	string	ID, si existe, del trace durante el cual se produjo el registro de eventos. Comportamiento de la propiedad: - *solo lectura*
spanid	string	ID, si existe, del span durante el cual se produjo el registro de eventos. Comportamiento de la propiedad: - *solo lectura*
trace_flags	integer	Indicadores del trace del contexto del trace del registro de eventos, como una máscara de bits. Si el bit menos significativo está establecido (valor 1), el trace se muestrea. Comportamiento de la propiedad: - *solo lectura*
severity_text	string	Nivel de gravedad del registro de eventos, proporcionado por la aplicación (por ejemplo, `INFO`, `WARN` o `ERROR`). Comportamiento de la propiedad: - *solo lectura*

Property	Type	Description
severity_number	integer	Nivel de gravedad del registro de eventos, como un número. Valores posibles: 0 - UNSPECIFIED; 1-4 - TRACE; 5-8 - DEBUG; 9-12 - INFO; 13-16 - WARN; 17-20 - ERROR; 21-24 - FATAL. Comportamiento de la propiedad: - *solo lectura*
service_name	string	Nombre del servicio que produjo el registro de eventos (por ejemplo, `zabbix-frontend`), obtenido del atributo de recurso `service.name` de OpenTelemetry. Comportamiento de la propiedad: - *solo lectura*
body	string	Mensaje del registro de eventos. Comportamiento de la propiedad: - *solo lectura*
resource_schema_url	string	URL, si existe, del esquema de OpenTelemetry de los atributos del recurso. Comportamiento de la propiedad: - *solo lectura*
resource_attributes	object	Atributos del recurso clave-valor del registro de eventos (por ejemplo, `host.name: eabf8c2a5ddc`). Comportamiento de la propiedad: - *solo lectura*
scope_schema_url	string	URL, si existe, del esquema de OpenTelemetry del ámbito de instrumentación y sus atributos. Comportamiento de la propiedad: - *solo lectura*
scope_name	string	Nombre del ámbito de instrumentación que produjo el registro de eventos. Comportamiento de la propiedad: - *solo lectura*
scope_version	string	Versión, si existe, del ámbito de instrumentación. Comportamiento de la propiedad: - *solo lectura*
scope_attributes	object	Atributos clave-valor del ámbito de instrumentación. Comportamiento de la propiedad: - *solo lectura*
log_attributes	object	Atributos del registro clave-valor del registro de eventos (por ejemplo, `log.file.path: /var/log/zabbix/zabbix_frontend.log`). Comportamiento de la propiedad: - *solo lectura*
event_name	string	Nombre, si existe, del tipo de evento que representa el registro de eventos. Comportamiento de la propiedad: - *solo lectura*

apmlog.get

Descripción

`integer/array apmlog.get(object parameters)`

El método permite recuperar registros de log según los parámetros indicados.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object)` Parámetros que definen el resultado deseado.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
time_from	timestamp	Devuelve únicamente los registros de log de eventos que ocurrieron en el momento indicado o después. Comportamiento del parámetro: - *obligatorio*
time_till	timestamp	Devuelve únicamente los registros de log de eventos que ocurrieron antes del momento indicado. Comportamiento del parámetro: - *obligatorio*
traceids	string/array	Devuelve únicamente los registros de log generados durante los traces indicados.
spanids	string/array	Devuelve únicamente los registros de log generados durante los spans indicados.
with_flags_on	integer	Devuelve únicamente los registros de log que tienen establecidos todos los trace flags indicados. Para especificar varios flags, utilice la suma de sus valores.
with_flags_off	integer	Devuelve únicamente los registros de log que no tienen establecido ninguno de los trace flags indicados. Para especificar varios flags, utilice la suma de sus valores.
resource_attributes	object/array	Devuelve únicamente los registros de log con los atributos de recurso indicados. Formato: `[{"key": "<key>", "value": "<value>", "operator": "<operator>"}, ...]`. Un array vacío devuelve todos los registros de log. Valores posibles de operator: 0 - *(predeterminado)* Contiene; 1 - Es igual a; 2 - No contiene; 3 - No es igual a; 4 - Existe; 5 - No existe.
resource_attributes_evaltype	integer	Método de evaluación de las condiciones de `resource_attributes`. Valores posibles: 0 - *(predeterminado)* And/Or; 2 - Or.

Parámetro	Tipo	Descripción
scope_attributes	object/array	Devuelve únicamente los registros de log con los atributos de scope indicados. Formato: `[{"key": "<key>", "value": "<value>", "operator": "<operator>"}, ...]`. Un array vacío devuelve todos los registros de log. Valores posibles de operator: 0 - *(predeterminado)* Contiene; 1 - Es igual a; 2 - No contiene; 3 - No es igual a; 4 - Existe; 5 - No existe.
scope_attributes_evaltype	integer	Método de evaluación de las condiciones de `scope_attributes`. Valores posibles: 0 - *(predeterminado)* And/Or; 2 - Or.
log_attributes	object/array	Devuelve únicamente los registros de log con los atributos de log indicados. Formato: `[{"key": "<key>", "value": "<value>", "operator": "<operator>"}, ...]`. Un array vacío devuelve todos los registros de log. Valores posibles de operator: 0 - *(predeterminado)* Contiene; 1 - Es igual a; 2 - No contiene; 3 - No es igual a; 4 - Existe; 5 - No existe.
log_attributes_evaltype	integer	Método de evaluación de las condiciones de `log_attributes`. Valores posibles: 0 - *(predeterminado)* And/Or; 2 - Or.
filter	object	Devuelve únicamente los resultados que coinciden exactamente con el filtro indicado. Acepta un objeto cuyas claves son nombres de propiedades y cuyos valores son un único valor o un array de valores con los que se debe comparar. Propiedades compatibles: `traceid`, `spanid`, `trace_flags`, `severity_text`, `severity_number`, `service_name`, `resource_schema_url`, `scope_schema_url`, `scope_name`, `scope_version`, `event_name`.
search	object	Devuelve los resultados que coinciden con el patrón indicado (sin distinguir entre mayúsculas y minúsculas). Acepta un objeto cuyas claves son nombres de propiedades y cuyos valores son cadenas que se deben buscar. Si no se proporcionan opciones adicionales, se realizará una búsqueda `LIKE "%…%"`. Propiedades compatibles: `severity_text`, `service_name`, `body`, `resource_schema_url`, `scope_schema_url`, `scope_name`, `scope_version`, `event_name`.

Parámetro	Tipo	Descripción
sortfield	string/array	Ordena el resultado según las propiedades indicadas. Valores posibles: `timestamp`, `traceid`, `spanid`, `trace_flags`, `severity_text`, `severity_number`, `service_name`, `resource_schema_url`, `scope_schema_url`, `scope_name`, `scope_version`, `event_name`.
offset	integer	Número de registros que se deben omitir antes de devolver el resultado.
countOutput	boolean	Estos parámetros se describen en la referencia.
excludeSearch	boolean	
limit	integer	
output	query	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores devueltos

`(integer/array)` Devuelve uno de los siguientes valores:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperación de registros de error de un trace

Recupere los registros con la gravedad `ERROR` (números de gravedad 17-20) asociados al trace "8a4f1c7e2b9d4e6fa0c3b5d7e9f1a2c4".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "apmlog.get",
    "params": {
        "output": "extend",
        "time_from": 1790848800,
        "time_till": 1790852400,
        "traceids": "8a4f1c7e2b9d4e6fa0c3b5d7e9f1a2c4",
        "filter": {
            "severity_number": [17, 18, 19, 20]
        },
        "sortfield": "timestamp",
        "sortorder": "ASC"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "timestamp": "1790849305091452376",
            "traceid": "8a4f1c7e2b9d4e6fa0c3b5d7e9f1a2c4",
            "spanid": "5d2e8f1a9c3b7046",
            "trace_flags": "1",
            "severity_text": "ERROR",
            "severity_number": "17",
            "service_name": "zabbix-frontend",
            "body": "mysqli_query(): (HY000/1205): Lock wait timeout exceeded; try restarting transaction"
            "resource_schema_url": "https://opentelemetry.io/schemas/1.26.0",
            "resource_attributes": {
```

```
                "service.name": "zabbix-frontend",
                "host.name": "6583ceeb5f94",
                "host.arch": "x86_64",
                "os.type": "linux"
            },
            "scope_schema_url": "",
            "scope_name": "zabbix-frontend",
            "scope_version": "",
            "scope_attributes": {},
            "log_attributes": {
                "log.file.path": "/var/log/zabbix/zabbix_frontend.log",
                "code.filepath": "/usr/share/zabbix/include/db.inc.php"
            },
            "event_name": ""
        }
    ],
    "id": 1
}
```


Búsqueda de mensajes de registro

Recupere la hora, el servicio y el mensaje de los 10 registros de log más recientes cuyo mensaje contenga ”timeout”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "apmlog.get",
    "params": {
        "output": ["timestamp", "service_name", "severity_text", "body"],
        "time_from": 1790848800,
        "time_till": 1790852400,
        "search": {
            "body": "timeout"
        },
        "sortfield": "timestamp",
        "sortorder": "DESC",
        "limit": 10
    },
    "id": 1
}
```

Respuesta:


```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "timestamp": "1790849305091452376",
            "service_name": "zabbix-frontend",
            "severity_text": "ERROR",
            "body": "mysqli_query(): (HY000/1205): Lock wait timeout exceeded; try restarting transaction"
        },
        {
            "timestamp": "1790848961730284115",
            "service_name": "zabbix-frontend",
            "severity_text": "ERROR",
            "body": "mysqli_query(): (HY000/1205): Lock wait timeout exceeded; try restarting transaction"
        }
    ],
    "id": 1
}
```


Véase también

- Traza de APM

- Span de APM
- Métrica de APM

Fuente

CApmLog::get() en *ui/include/classes/api/services/CApmLog.php*.

Registro de auditoría

Esta clase está diseñada para trabajar con el registro de auditoría.

Referencias de objetos:

- Registro de auditoría

Métodos disponibles:

- auditlog.get - recuperar registros de auditoría

Objeto de registro de auditoría

Los siguientes objetos están directamente relacionados con la API `auditlog`.

Registro de auditoría

El objeto de registro de auditoría contiene información sobre las acciones del usuario. Tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
auditid	ID	ID de la entrada del registro de auditoría. Generado usando el algoritmo CUID.
userid	ID	ID de usuario del autor de la entrada del registro de auditoría.
username	string	Nombre de usuario del autor de la entrada del registro de auditoría.
clock	timestamp	Marca de tiempo de creación de la entrada del registro de auditoría.
ip	string	Dirección IP del autor de la entrada del registro de auditoría.
action	integer	Acción de la entrada del registro de auditoría. Valores posibles: 0 - Añadir; 1 - Actualizar; 2 - Eliminar; 4 - Cerrar sesión; 7 - Ejecutar; 8 - Iniciar sesión; 9 - Fallo de inicio de sesión; 10 - Borrar historial; 11 - Refrescar configuración; 12 - Push.

Propiedad	Tipo	Descripción
resourcetype	integer	Tipo de recurso de la entrada del registro de auditoría. Valores posibles: 0 - Usuario; 3 - Tipo de medio; 4 - Host; 5 - Acción; 6 - Gráfico; 11 - Grupo de usuarios; 13 - Trigger; 14 - Grupo de hosts; 15 - Item; 16 - Imagen; 17 - Mapa de valores; 18 - Servicio; 19 - Mapa; 22 - Escenario web; 23 - Regla de descubrimiento; 25 - Script; 26 - Proxy; 27 - Mantenimiento; 28 - Expresión regular; 29 - Macro; 30 - Template; 31 - Prototipo de trigger; 32 - Asignación de iconos; 33 - Dashboard; 34 - Correlación de eventos; 35 - Prototipo de gráfico; 36 - Prototipo de item; 37 - Prototipo de host; 38 - Autoregistro; 39 - Módulo; 40 - Configuración; 41 - Housekeeping; 42 - Autenticación; 43 - Dashboard de template; 44 - Rol de usuario; 45 - Token de API; 46 - Informe programado; 47 - Nodo de alta disponibilidad; 48 - SLA; 49 - Directorio de usuarios; 50 - Grupo de templates; 51 - Conector; 52 - Regla LLD; 53 - Historial; 54 - Autenticación multifactor; 55 - Grupo de proxies; 56 - Prototipo de regla LLD.
resourceid	ID	Identificador de recurso de la entrada del registro de auditoría.
resource_cuid	ID	Identificador único de recurso de la entrada del registro de auditoría generado usando el algoritmo CUID.
resourcename	string	Nombre legible del recurso de la entrada del registro de auditoría.
recordsetid	ID	ID del conjunto de registros de la entrada del registro de auditoría. Las entradas del registro de auditoría creadas durante la misma operación tendrán el mismo ID de conjunto de registros. Generado usando el algoritmo CUID.

Propiedad	Tipo	Descripción
details	text	Detalles de la entrada del registro de auditoría. Los detalles se almacenan como un objeto JSON, donde cada nombre de propiedad es una ruta a la propiedad u objeto anidado en el que ocurrió el cambio, y donde cada valor contiene los datos (en formato de array) sobre el cambio en esta propiedad u objeto anidado. Formatos de valor posibles: ["add"] - Se ha añadido un objeto anidado; ["add", "<valor>"] - La propiedad del objeto añadido es igual a <valor>; ["update"] - Se ha actualizado un objeto anidado; ["update", "<nuevo valor>", "<valor anterior>"] - La propiedad del objeto actualizado cambió de <valor anterior> a <nuevo valor>; ["delete"] - Se ha eliminado un objeto anidado.

auditlog.get

Descripción

`integer/array auditlog.get(object parameters)`

El método permite recuperar registros del audit log según los parámetros proporcionados.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método soporta los siguientes parámetros.

Parámetro	Tipo	Descripción
auditids	ID/array	Devolver solo los registros de auditoría con los IDs dados.
userids	ID/array	Devolver solo los registros de auditoría que fueron creados por los usuarios dados.
time_from	timestamp	Devuelve solo las entradas del registro de auditoría que se hayan creado después o en el momento dado.
time_till	timestamp	Devuelve solo las entradas del registro de auditoría que se hayan creado antes o en el momento dado.
sortfield	string/array	Ordena el resultado por las propiedades dadas. Valores posibles: `auditid`, `userid`, `clock`.
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos;

- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperar el registro de auditoría

Recuperar los dos últimos registros del registro de auditoría.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "auditlog.get",
    "params": {
        "output": "extend",
        "sortfield": "clock",
        "sortorder": "DESC",
        "limit": 2
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "auditid": "cksstgfam0001yhdcc41y20q2",
            "userid": "1",
            "username": "Admin",
            "clock": "1629975715",
            "ip": "127.0.0.1",
            "action": "1",
            "resourcetype": "0",
            "resourceid": "0",
            "resource_cuid": "0",
            "resourcename": "Jim",
            "recordsetid": "cksstgfal0000yhdcso67ondl",
            "details": "{\"user.name\":[\"update\",\"Jim\",\"\"],\"user.medias[37]\":[\"add\"],\"user.medi
        },
        {
            "auditid": "ckssofl0p0001yhdcqxclsg8r",
            "userid": "1",
            "username": "Admin",
            "clock": "1629967278",
            "ip": "127.0.0.1",
            "action": "0",
            "resourcetype": "0",
            "resourceid": "20",
            "resource_cuid": "0",
            "resourcename": "John",
            "recordsetid": "ckssofl0p0000yhdcpxyo1jgo",
            "details": "{\"user.username\":[\"add\",\"John\"], \"user.userid:\":[\"add\",\"20\"],\"user.us
        }
    ],
    "id": 1
}
```

Véase también

- Objeto de registro de auditoría

Fuente

CAuditLog::get() en *ui/include/classes/api/services/CAuditLog.php*.

Regla de descubrimiento

Esta clase está diseñada para trabajar con reglas de descubrimiento de red.

> **Note:**
> Esta API está destinada a trabajar con reglas de descubrimiento de red. Para reglas de descubrimiento de bajo nivel, consulte la API de reglas LLD.

Referencias de objetos:

- Regla de descubrimiento

Métodos disponibles:

- drule.create - crear nuevas reglas de descubrimiento de red
- drule.delete - eliminar reglas de descubrimiento de red
- drule.get - recuperar reglas de descubrimiento de red
- drule.update - actualizar reglas de descubrimiento de red

Objeto de regla de descubrimiento

Los siguientes objetos están directamente relacionados con la API `drule`.

Regla de descubrimiento

El objeto de regla de descubrimiento define una regla de descubrimiento de red. Tiene las siguientes propiedades.

Property	Type	Description
druleid	ID	ID de la regla de descubrimiento. Comportamiento de la propiedad: - *solo lectura* - *obligatoria* para operaciones de actualización
iprange	string	Uno o varios rangos de IP que se comprobarán, separados por comas. Consulte la sección configuración del descubrimiento de red para obtener más información sobre los formatos de rangos de IP compatibles. Comportamiento de la propiedad: - *obligatoria* para operaciones de creación
name	string	Nombre de la regla de descubrimiento. Comportamiento de la propiedad: - *obligatoria* para operaciones de creación
delay	string	Intervalo de ejecución de la regla de descubrimiento. Acepta un número de segundos (por ejemplo, `1`, `60`, `300`), un valor con un sufijo de tiempo (por ejemplo, `30s`, `1m`, `2h`, `1d`) o una macro de usuario. Valor predeterminado: `1h`.
proxyid	ID	ID del proxy utilizado para el descubrimiento.
status	integer	Indica si la regla de descubrimiento está habilitada. Valores posibles: 0 - *(predeterminado)* habilitada; 1 - deshabilitada.

Property	Type	Description
concurrency_max	integer	Número máximo de comprobaciones simultáneas por regla de descubrimiento. Valores posibles: 0 - *(predeterminado)* número ilimitado de comprobaciones; 1 - una comprobación; 2-999 - número personalizado de comprobaciones.
error	string	Texto del error si se ha producido algún problema al ejecutar la regla de descubrimiento. Comportamiento de la propiedad: *- solo lectura*

drule.create

Descripción

`object drule.create(object/array discoveryRules)`

Este método permite crear nuevas reglas de descubrimiento.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(objeto/array)` Reglas de descubrimiento a crear.

Además de las propiedades estándar de las reglas de descubrimiento, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
dchecks	array	Comprobaciones de descubrimiento a crear para la regla de descubrimiento. Comportamiento del parámetro: *- obligatorio*

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las reglas de descubrimiento creadas bajo la propiedad `druleids`. El orden de los IDs devueltos coincide con el orden de las reglas de descubrimiento enviadas.

Ejemplos

Crear una regla de descubrimiento

Cree una regla de descubrimiento para encontrar máquinas que ejecutan el agente de Zabbix en la red local. La regla debe usar una única comprobación de agente de Zabbix en el puerto 10050.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "drule.create",
    "params": {
        "name": "Descubrimiento de agente de Zabbix",
        "iprange": "192.168.1.1-255",
        "concurrency_max": "10",
        "dchecks": [
            {
                "type": "9",
                "key_": "system.uname",
```

```
                "ports": "10050",
                "uniq": "0"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "druleids": [
            "6"
        ]
    },
    "id": 1
}
```

Ver también

- Comprobación de descubrimiento

Fuente

CDRule::create() en *ui/include/classes/api/services/CDRule.php*.

drule.delete

Descripción

`object drule.delete(array discoveryRuleIds)`

Este método le permite eliminar reglas de descubrimiento.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración del rol de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(array)` IDs de las reglas de descubrimiento a eliminar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las reglas de descubrimiento eliminadas bajo la propiedad `druleids`.

Ejemplos

Eliminar varias reglas de descubrimiento

Eliminar dos reglas de descubrimiento.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "drule.delete",
    "params": [
        "4",
        "6"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "druleids": [
            "4",
            "6"
        ]
    },
    "id": 1
}
```

Fuente

CDRule::delete() en *ui/include/classes/api/services/CDRule.php*.

drule.get

Descripción

`integer/array drule.get(object parameters)`

El método permite recuperar reglas de descubrimiento según los parámetros proporcionados.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
dhostids	ID/array	Devolver solo las reglas de descubrimiento que crearon los hosts descubiertos indicados.
druleids	ID/array	Devolver solo las reglas de descubrimiento con los IDs indicados.
dserviceids	ID/array	Devolver solo las reglas de descubrimiento que crearon los servicios descubiertos indicados.
selectDChecks	query	Devuelve una propiedad `dchecks` con las comprobaciones de descubrimiento utilizadas por la regla de descubrimiento. Admite `count`.
selectDHosts	query	Devuelve una propiedad `dhosts` con los hosts descubiertos creados por la regla de descubrimiento. Admite `count`.
limitSelects	integer	Limita el número de registros devueltos por las subselecciones. Se aplica a las siguientes subselecciones: `selectDChecks` - los resultados se ordenarán por `dcheckid`; `selectDHosts` - los resultados se ordenarán por `dhostsid`.
sortfield	string/array	Ordena el resultado por las propiedades indicadas. Valores posibles: `druleid`, `name`.
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	

Parámetro	Tipo	Descripción
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperar todas las reglas de descubrimiento

Recupera todas las reglas de descubrimiento configuradas y las comprobaciones de descubrimiento que utilizan.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "drule.get",
    "params": {
        "output": "extend",
        "selectDChecks": "extend"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "druleid": "2",
            "proxyid": "0",
            "name": "Local network",
            "iprange": "192.168.3.1-255",
            "delay": "5s",
            "status": "0",
            "concurrency_max": "0",
            "error": "",
            "dchecks": [
                {
                    "dcheckid": "7",
                    "druleid": "2",
                    "type": "3",
                    "key_": "",
                    "snmp_community": "",
                    "ports": "21",
                    "snmpv3_securityname": "",
                    "snmpv3_securitylevel": "0",
                    "snmpv3_authpassphrase": "",
                    "snmpv3_privpassphrase": "",
                    "uniq": "0",
                    "snmpv3_authprotocol": "0",
                    "snmpv3_privprotocol": "0",
                    "snmpv3_contextname": "",
                    "host_source": "1",
                    "name_source": "0",
                    "allow_redirect": "0"
                },
                {
                    "dcheckid": "8",
```

```
                    "druleid": "2",
                    "type": "4",
                    "key_": "",
                    "snmp_community": "",
                    "ports": "80",
                    "snmpv3_securityname": "",
                    "snmpv3_securitylevel": "0",
                    "snmpv3_authpassphrase": "",
                    "snmpv3_privpassphrase": "",
                    "uniq": "0",
                    "snmpv3_authprotocol": "0",
                    "snmpv3_privprotocol": "0",
                    "snmpv3_contextname": "",
                    "host_source": "1",
                    "name_source": "0",
                    "allow_redirect": "0"
                }
            ]
        },
        {
            "druleid": "6",
            "proxyid": "0",
            "name": "Zabbix agent discovery",
            "iprange": "192.168.1.1-255",
            "delay": "1h",
            "status": "0",
            "concurrency_max": "10",
            "error": "",
            "dchecks": [
                {
                    "dcheckid": "10",
                    "druleid": "6",
                    "type": "9",
                    "key_": "system.uname",
                    "snmp_community": "",
                    "ports": "10050",
                    "snmpv3_securityname": "",
                    "snmpv3_securitylevel": "0",
                    "snmpv3_authpassphrase": "",
                    "snmpv3_privpassphrase": "",
                    "uniq": "0",
                    "snmpv3_authprotocol": "0",
                    "snmpv3_privprotocol": "0",
                    "snmpv3_contextname": "",
                    "host_source": "2",
                    "name_source": "3",
                    "allow_redirect": "0"
                }
            ]
        }
    ],
    "id": 1
}
```


Ver también

- Host descubierto
- Chequeo de descubrimiento

Fuente

CDRule::get() en *ui/include/classes/api/services/CDRule.php*.

drule.update

Descripción

`object drule.update(object/array discoveryRules)`

Este método permite actualizar reglas de descubrimiento existentes.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object/array)` Propiedades de la regla de descubrimiento que se actualizarán.

La propiedad `druleid` debe definirse para cada regla de descubrimiento, todas las demás propiedades son opcionales. Sólo se actualizarán las propiedades pasadas, todas las demás permanecerán sin cambios.

Además de las propiedades estándar de la regla de descubrimiento, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
dchecks	array	Comprobaciones de descubrimiento para reemplazar las comprobaciones existentes.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las reglas de descubrimiento actualizadas bajo la propiedad `druleids`.

Ejemplos

Cambiar el rango de IP de una regla de descubrimiento

Cambie el rango de IP de una regla de descubrimiento a "192.168.2.1-255".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "drule.update",
    "params": {
        "druleid": "6",
        "iprange": "192.168.2.1-255"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "druleids": [
            "6"
        ]
    },
    "id": 1
}
```

Ver también

- Comprobación de descubrimiento

Fuente

CDRule::update() en *ui/include/classes/api/services/CDRule.php*.

Regla LLD

Esta clase está diseñada para trabajar con reglas de descubrimiento de bajo nivel.

Referencias de objetos:

- Regla LLD
  - Cabecera HTTP
  - Campo de consulta HTTP
- Filtro de regla LLD
  - Condición de filtro de regla LLD
- Ruta de macro LLD
- Preprocesamiento de regla LLD
- Anulaciones de regla LLD
  - Filtro de anulación de regla LLD
    - Condición de filtro de anulación de regla LLD
  - Operación de anulación de regla LLD
    - Estado de la operación de anulación de regla LLD
    - Descubrimiento de la operación de anulación de regla LLD
    - Periodo de la operación de anulación de regla LLD
    - Histórico de la operación de anulación de regla LLD
    - Tendencias de la operación de anulación de regla LLD
    - Severidad de la operación de anulación de regla LLD
    - Etiqueta de la operación de anulación de regla LLD
    - Template de la operación de anulación de regla LLD
    - Inventario de la operación de anulación de regla LLD

Métodos disponibles:

- discoveryrule.create - crear nuevas reglas LLD
- discoveryrule.delete - eliminar reglas LLD
- discoveryrule.get - obtener reglas LLD
- discoveryrule.update - actualizar reglas LLD

Objeto de regla LLD

Los siguientes objetos están directamente relacionados con la API `discoveryrule`.

Regla de LLD

El objeto de regla de descubrimiento de bajo nivel tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
itemid	ID	ID de la regla de LLD. Comportamiento de la propiedad: - *solo lectura* - *obligatoria* para operaciones de actualización

Propiedad	Tipo	Descripción
delay	string	Intervalo de actualización de la regla de LLD. Acepta un número de segundos (por ejemplo, 1, 60, 300), un valor con un sufijo de tiempo (por ejemplo, 30s, 1m, 2h, 1d) o, opcionalmente, uno o más intervalos personalizados, todos separados por punto y coma. Los intervalos personalizados pueden ser una combinación de intervalos flexibles y de planificación. Acepta macros de usuario. Si se utiliza, el valor debe ser una única macro. No se admiten varias macros ni macros combinadas con texto. Los intervalos flexibles pueden escribirse como dos macros separadas por una barra diagonal (por ejemplo, {$FLEX_INTERVAL}/{$FLEX_PERIOD}). Ejemplo: 1h;wd1-5h9-18;{$Macro1}/1-7,00:00-24:00;0/6-7,12:00-24:00;{$Macr Comportamiento de la propiedad: - *obligatoria* si type está establecido en ”Zabbix agent” (0), ”Simple check” (3), ”Zabbix internal” (5), ”External check” (10), ”Database monitor” (11), ”IPMI agent” (12), ”SSH agent” (13), ”TELNET agent” (14), ”JMX agent” (16), ”HTTP agent” (19), ”SNMP agent” (20), ”Script” (21), ”Browser” (22), o si type está establecido en ”Zabbix agent (active)” (7) y key_ no contiene ”mqtt.get”
hostid	ID	ID del host al que pertenece la regla de LLD. Comportamiento de la propiedad: - *constante* - *obligatoria* para operaciones de creación
flags	integer	Origen de la regla de descubrimiento. Valores posibles: 1 - una regla de descubrimiento de bajo nivel; 5 - una regla de descubrimiento de bajo nivel convertida a partir de un prototipo. Comportamiento de la propiedad: - *solo lectura*
interfaceid	ID	ID de la interfaz del host de la regla de LLD. Comportamiento de la propiedad: - *obligatoria* si la regla de LLD pertenece a un host y type está establecido en ”Zabbix agent”, ”IPMI agent”, ”JMX agent” o ”SNMP agent” - *compatible* si la regla de LLD pertenece a un host y type está establecido en ”Simple check”, ”External check”, ”SSH agent”, ”TELNET agent” o ”HTTP agent”
key_	string	Clave de la regla de LLD. Comportamiento de la propiedad: - *obligatoria* para operaciones de creación - *solo lectura* para objetos heredados
name	string	Nombre de la regla de LLD. Comportamiento de la propiedad: - *obligatoria* para operaciones de creación - *solo lectura* para objetos heredados

Propiedad	Tipo	Descripción
type	integer	Tipo de la regla de LLD. Valores posibles: 0 - Zabbix agent; 2 - Zabbix trapper; 3 - Simple check; 5 - Zabbix internal; 7 - Zabbix agent (active); 10 - External check; 11 - Database monitor; 12 - IPMI agent; 13 - SSH agent; 14 - TELNET agent; 16 - JMX agent; 18 - Dependent item; 19 - HTTP agent; 20 - SNMP agent; 21 - Script; 22 - Browser; 23 - Nested. Este tipo solo se permite si el elemento principal de la regla es un template (que se supone que está vinculado a un prototipo de host durante el descubrimiento) o un host descubierto (que tiene una regla de LLD en su raíz). Comportamiento de la propiedad: - *obligatoria* para operaciones de creación - *solo lectura* para objetos heredados
url	string	Cadena de URL. Admite macros de usuario, {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {HOST.PORT}, {ITEM.ID}, {ITEM.KEY}. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados
allow_traps	integer	Permite que el valor se complete de forma similar al item trapper. Valores posibles: 0 - *(predeterminado)* No aceptar datos entrantes; 1 - Aceptar datos entrantes. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent”
authtype	integer	Método de autenticación. Valores posibles si `type` está establecido en ”SSH agent”: 0 - *(predeterminado)* contraseña; 1 - clave pública. Valores posibles si `type` está establecido en ”HTTP agent”: 0 - *(predeterminado)* ninguno; 1 - básico; 2 - NTLM; 3 - Kerberos; 4 - Digest. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”SSH agent” o ”HTTP agent” - *solo lectura* para objetos heredados (si `type` está establecido en ”HTTP agent”)
description	string	Descripción de la regla de LLD.

Propiedad	Tipo	Descripción
error	string	Texto del error si se producen problemas al actualizar el valor de la regla de LLD. Comportamiento de la propiedad: - *solo lectura*
follow_redirects	integer	Seguir las redirecciones de respuesta durante el sondeo de datos. Valores posibles: 0 - No seguir redirecciones; 1 - *(predeterminado)* Seguir redirecciones. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados
headers	array	Matriz de encabezados que se enviarán al realizar una solicitud HTTP. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados
http_proxy	string	Cadena de conexión del proxy HTTP(S). Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados
ipmi_sensor	string	Sensor IPMI. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”IPMI agent” y `key_` no está establecido en ”ipmi.get” - *compatible* si `type` está establecido en ”IPMI agent” - *solo lectura* para objetos heredados
jmx_endpoint	string	Cadena de conexión personalizada del JMX agent. Predeterminado: service:jmx:rmi:///jndi/rmi://{HOST.CONN}:{HOST.PORT}/jmxrmi Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”JMX agent”
lifetime	string	Periodo tras el cual se eliminarán los items que ya no se descubran. Acepta un número de segundos (por ejemplo, 3600), un valor con un sufijo de tiempo (por ejemplo, 3600`s`, 180`m`, 24`h`, 1`d`) o una macro de usuario. Predeterminado: 7`d`.
lifetime_type	integer	Escenario para eliminar los recursos de LLD perdidos. Valores posibles: 0 - *(predeterminado)* Eliminar después de alcanzar el umbral de tiempo de vida; 1 - No eliminar; 2 - Eliminar inmediatamente.
enabled_lifetime	string	Periodo tras el cual se deshabilitarán los items que ya no se descubran. Acepta un número de segundos (por ejemplo, 1, 60, 300), un valor con un sufijo de tiempo (por ejemplo, 30`s`, 1`m`, 2`h`, 1`d`) o una macro de usuario. Predeterminado: 0.

Propiedad	Tipo	Descripción
enabled_lifetime_type	integer	Escenario para deshabilitar los recursos de LLD perdidos. Valores posibles: 0 - Deshabilitar después de alcanzar el umbral de tiempo de vida; 1 - No deshabilitar; 2 - *(predeterminado)* Deshabilitar inmediatamente.
master_itemid	ID	ID del item maestro. La regla de descubrimiento no puede ser el item maestro de otra regla de descubrimiento. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”Dependent item” - *solo lectura* para objetos heredados
output_format	integer	Indica si la respuesta debe convertirse a JSON. Valores posibles: 0 - *(predeterminado)* Almacenar sin procesar; 1 - Convertir a JSON. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados
params	string	Parámetros adicionales según el tipo de regla de LLD: - script ejecutado para reglas de LLD SSH y Telnet; - consulta SQL para reglas de LLD de monitor de base de datos; - fórmula para reglas de LLD calculadas; - script para reglas de LLD de script y navegador. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”Database monitor”, ”SSH agent”, ”TELNET agent”, ”Script” o ”Browser” - *solo lectura* para objetos heredados (si `type` está establecido en ”Script” o ”Browser”)
parameters	object/array	Parámetros adicionales si `type` está establecido en ”Script” o ”Browser”. Matriz de objetos con las propiedades `name` y `value`, donde `name` debe ser único. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”Script” o ”Browser” - *solo lectura* para objetos heredados
password	string	Contraseña para la autenticación. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”JMX agent” y `username` está establecido - *compatible* si `type` está establecido en ”Simple check”, ”Database monitor”, ”SSH agent”, ”TELNET agent” o ”HTTP agent” - *solo lectura* para objetos heredados (si `type` está establecido en ”HTTP agent”)
post_type	integer	Tipo del cuerpo de datos de publicación almacenado en la propiedad `posts`. Valores posibles: 0 - *(predeterminado)* Datos sin procesar; 2 - Datos JSON; 3 - Datos XML. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados

Propiedad	Tipo	Descripción
posts	string	Datos del cuerpo de la solicitud HTTP(S). Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”HTTP agent” y `post_type` está establecido en ”JSON data” o ”XML data” - *compatible* si `type` está establecido en ”HTTP agent” y `post_type` está establecido en ”Raw data” - *solo lectura* para objetos heredados
privatekey	string	Nombre del archivo de clave privada. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”SSH agent” y `authtype` está establecido en ”public key”
publickey	string	Nombre del archivo de clave pública. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”SSH agent” y `authtype` está establecido en ”public key”
query_fields	array	Matriz de campos de consulta que se enviarán al realizar una solicitud HTTP. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados
request_method	integer	Tipo de método de solicitud. Valores posibles: 0 - *(predeterminado)* GET; 1 - POST; 2 - PUT; 3 - HEAD. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados
retrieve_mode	integer	Parte de la respuesta que debe almacenarse. Valores posibles si `request_method` está establecido en ”GET”, ”POST” o ”PUT”: 0 - *(predeterminado)* Cuerpo; 1 - Encabezados; 2 - Se almacenarán tanto el cuerpo como los encabezados. Valores posibles si `request_method` está establecido en ”HEAD”: 1 - Encabezados. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados
snmp_oid	string	OID de SNMP. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”SNMP agent” - *solo lectura* para objetos heredados
ssl_cert_file	string	Ruta del archivo de clave pública SSL. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados

Propiedad	Tipo	Descripción
ssl_key_file	string	Ruta del archivo de clave privada SSL. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados
ssl_key_password	string	Contraseña del archivo de clave SSL. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados
state	integer	Estado de la regla de LLD. Valores posibles: 0 - *(predeterminado)* normal; 1 - no compatible. Comportamiento de la propiedad: - *solo lectura*
status	integer	Estado de la regla de LLD. Valores posibles: 0 - *(predeterminado)* regla de LLD habilitada; 1 - regla de LLD deshabilitada.
status_codes	string	Rangos de códigos de estado HTTP requeridos, separados por comas. También admite macros de usuario como parte de una lista separada por comas. Ejemplo: 200,200-{$M},{$M},200-400 Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados
templateid	ID	ID de la regla de LLD del template principal. Comportamiento de la propiedad: - *solo lectura*
timeout	string	Tiempo de espera de la solicitud de sondeo de datos del item. Acepta un número de segundos (por ejemplo, 1, 60, 300) o un valor con un sufijo de tiempo (por ejemplo, 30s, 1m). También acepta macros de usuario. Rango de valores posibles: 1-600s. Predeterminado: ”” - utilizar la configuración del proxy/global. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”Zabbix agent” (0), ”Simple check” (3) y `key_` no comienza por ”`vmware.`” ni ”`icmpping`”, ”Zabbix agent (active)” (7), ”External check” (10), ”Database monitor” (11), ”SSH agent” (13), ”TELNET agent” (14), ”HTTP agent” (19), ”SNMP agent” (20) y `snmp_oid` comienza por ”`walk[`” o ”`get[`”, ”Script” (21), ”Browser” (22) - *solo lectura* para objetos heredados, - si se aplica el valor predeterminado y el item pertenece a un host supervisado por un proxy o grupo de proxies inaccesible, se utiliza el tiempo de espera global.

Propiedad	Tipo	Descripción
trapper_hosts	string	Hosts permitidos. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”Zabbix trapper”, o si `type` está establecido en ”HTTP agent” y `allow_traps` está establecido en ”Allow incoming data”. Si se omite la propiedad, se establece automáticamente en la macro de usuario `{$TRAPPER.ALLOWED_HOSTS}` (valor predeterminado). De forma predeterminada, `{$TRAPPER.ALLOWED_HOSTS}` es `127.0.0.1,::1`. Si la propiedad se establece en una cadena vacía, se denegarán las conexiones entrantes de todos los hosts.
username	string	Nombre de usuario para la autenticación. Comportamiento de la propiedad: - *obligatorio* si `type` está establecido en ”SSH agent”, ”TELNET agent”, o si `type` está establecido en ”JMX agent” y `password` está establecido - *compatible* si `type` está establecido en ”Simple check”, ”Database monitor” o ”HTTP agent” - *solo lectura* para objetos heredados (si `type` está establecido en ”HTTP agent”)
uuid	string	Identificador único universal, utilizado para vincular reglas de LLD importadas con las ya existentes. Se genera automáticamente si no se proporciona. Comportamiento de la propiedad: - *compatible* si la regla de LLD pertenece a un template
verify_host	integer	Indica si se debe validar que el nombre del host de la conexión coincida con el del certificado del host. Valores posibles: 0 - *(predeterminado)* No validar; 1 - Validar. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados
verify_peer	integer	Indica si se debe validar que el certificado del host sea auténtico. Valores posibles: 0 - *(predeterminado)* No validar; 1 - Validar. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”HTTP agent” - *solo lectura* para objetos heredados

Encabezado HTTP

El objeto header tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
name	string	Nombre del encabezado HTTP. Comportamiento de la propiedad: - *obligatorio*

Propiedad	Tipo	Descripción
value	string	Valor del encabezado. Comportamiento de la propiedad: - *obligatorio*

Campo de consulta HTTP

El objeto de campo de consulta define un nombre y un valor que se utilizan para especificar un parámetro de URL. Tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
name	string	Nombre del parámetro. Comportamiento de la propiedad: - *requerido*
value	string	Valor del parámetro. Comportamiento de la propiedad: - *requerido*

Filtro de regla LLD

El objeto de filtro de regla LLD define un conjunto de condiciones que se pueden utilizar para filtrar objetos descubiertos. Tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
conditions	object/array	Conjunto de condiciones de filtro a utilizar para filtrar resultados. Las condiciones se ordenarán en el orden de su colocación en la fórmula. Comportamiento de la propiedad: - *requerido*
evaltype	integer	Método de evaluación de la condición de filtro. Valores posibles: 0 - And/Or; 1 - And; 2 - Or; 3 - Expresión personalizada. Comportamiento de la propiedad: - *requerido*
eval_formula	string	Expresión generada que se utilizará para evaluar las condiciones del filtro. La expresión contiene IDs que hacen referencia a condiciones de filtro específicas por su `formulaid`. El valor de `eval_formula` es igual al valor de `formula` para los filtros con una expresión personalizada. Comportamiento de la propiedad: - *solo lectura*

Propiedad	Tipo	Descripción
formula	string	Expresión definida por el usuario que se utilizará para evaluar las condiciones de los filtros con una expresión personalizada. La expresión debe contener IDs que hagan referencia a condiciones de filtro específicas por su `formulaid`. Los IDs utilizados en la expresión deben coincidir exactamente con los definidos en las condiciones del filtro: ninguna condición puede quedar sin usar u omitida. Comportamiento de la propiedad: - *requerido* si `evaltype` está establecido en ”expresión personalizada”

Condición de filtro de la regla LLD

El objeto de condición de filtro de la regla LLD define una comprobación separada que se realiza sobre el valor de una macro LLD. Tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
macro	string	Macro LLD sobre la que realizar la comprobación. Comportamiento de la propiedad: - *obligatorio*
value	string	Valor con el que comparar. Comportamiento de la propiedad: - *obligatorio* si `operator` está establecido en ”coincide con la expresión regular” o ”no coincide con la expresión regular”
formulaid	string	ID único arbitrario que se utiliza para hacer referencia a la condición desde una expresión personalizada. Solo puede contener letras mayúsculas. El ID debe ser definido por el usuario al modificar las condiciones del filtro, pero se generará de nuevo al solicitarlas posteriormente. Comportamiento de la propiedad: - *obligatorio* si `evaltype` del objeto de filtro de regla LLD está establecido en ”expresión personalizada”
operator	integer	Operador de condición. Valores posibles: 8 - *(por defecto)* coincide con la expresión regular; 9 - no coincide con la expresión regular; 12 - existe; 13 - no existe.

Note:
Para comprender mejor cómo utilizar los filtros con varios tipos de expresiones, consulte los ejemplos en las páginas de los métodos discoveryrule.get y discoveryrule.create.

Ruta de macro LLD

La ruta de la macro LLD tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
lld_macro	string	Macro LLD. Comportamiento de la propiedad: - *requerido*

Propiedad	Tipo	Descripción
path	string	Selector para el valor que se asignará a la macro correspondiente. Comportamiento de la propiedad: - *requerido*

Preprocesamiento de la regla LLD

El objeto de preprocesamiento de la regla LLD tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
type	integer	El tipo de opción de preprocesamiento. Valores posibles: 5 - Expresión regular; 11 - XML XPath; 12 - JSONPath; 14 - Coincide con la expresión regular; 15 - No coincide con la expresión regular; 16 - Comprobar si hay un error en JSON; 17 - Comprobar si hay un error en XML; 20 - Descartar si no ha cambiado con latido; 21 - JavaScript; 23 - Prometheus a JSON; 24 - CSV a JSON; 25 - Reemplazar; 27 - XML a JSON; 28 - Valor de recorrido SNMP; 29 - Recorrido SNMP a JSON; 30 - Obtener valor SNMP. Comportamiento de la propiedad: - *required*
params	string	Parámetros adicionales utilizados por la opción de preprocesamiento. Los parámetros múltiples se separan mediante el carácter de nueva línea (\n). Comportamiento de la propiedad: - *required* si `type` está establecido en ”Expresión regular” (5), ”XML XPath” (11), ”JSONPath” (12), ”Coincide con la expresión regular” (14), ”No coincide con la expresión regular” (15), ”Comprobar si hay un error en JSON” (16), ”Comprobar si hay un error en XML” (17), ”Descartar si no ha cambiado con latido” (20), ”JavaScript” (21), ”Prometheus a JSON” (23), ”CSV a JSON” (24), ”Reemplazar” (25), ”Valor de recorrido SNMP” (28), ”Recorrido SNMP a JSON” (29) u ”Obtener valor SNMP” (30)

Propiedad	Tipo	Descripción
error_handler	integer	Tipo de acción utilizado en caso de fallo del paso de preprocesamiento. Valores posibles: 0 - El mensaje de error lo establece Zabbix server; 1 - Descartar el valor; 2 - Establecer un valor personalizado; 3 - Establecer un mensaje de error personalizado. Comportamiento de la propiedad: - *required* si `type` está establecido en ”Expresión regular” (5), ”XML XPath” (11), ”JSONPath” (12), ”Coincide con la expresión regular” (14), ”No coincide con la expresión regular” (15), ”Comprobar si hay un error en JSON” (16), ”Comprobar si hay un error en XML” (17), ”Prometheus a JSON” (23), ”CSV a JSON” (24), ”XML a JSON” (27), ”Valor de recorrido SNMP” (28), ”Recorrido SNMP a JSON” (29) u ”Obtener valor SNMP” (30)
error_handler_params	string	Parámetros del gestor de errores. Comportamiento de la propiedad: - *required* si `error_handler` está establecido en ”Establecer un valor personalizado” o ”Establecer un mensaje de error personalizado”

Los siguientes parámetros y gestores de errores son compatibles con cada tipo de preprocesamiento.

Tipo de preprocesamiento	Nombre	Parámetro 1	Parámetro 2	Parámetro 3	Gestores de errores compatibles
5	Expresión regular	pattern[1]	output[2]		0, 1, 2, 3
11	XML XPath	path[3]			0, 1, 2, 3
12	JSONPath	path[3]			0, 1, 2, 3
14	Coincide con la expresión regular	pattern[1]			0, 1, 2, 3
15	No coincide con la expresión regular	pattern[1]			0, 1, 2, 3
16	Comprobar si hay un error en JSON	path[3]			0, 1, 2, 3

Tipo de preprocesamiento	Nombre	Parámetro 1	Parámetro 2	Parámetro 3	Gestores de errores compatibles
17	Comprobar si hay un error en XML	path[3]			0, 1, 2, 3
20	Descartar si no ha cambiado con latido	seconds^4, 5^			
21	JavaScript	script[2]			
23	Prometheus a JSON	pattern^5, 6^			0, 1, 2, 3
24	CSV a JSON	character[2]	character[2]	0,1	0, 1, 2, 3
25	Reemplazar	search string[2]	replacement[2]		
27	XML a JSON				0, 1, 2, 3
28	Valor de recorrido SNMP	OID[2]	Formato: 0 - Sin cambios 1 - UTF-8 desde Hex-STRING 2 - MAC desde Hex-STRING 3 - Entero desde BITS		0, 1, 2, 3
29	Recorrido SNMP a JSON[7]	Field name[2]	OID prefix[2]	Formato: 0 - Sin cambios 1 - UTF-8 desde Hex-STRING 2 - MAC desde Hex-STRING 3 - Entero desde BITS	0, 1, 2, 3
30	Obtener valor SNMP	Formato: 1 - UTF-8 desde Hex-STRING 2 - MAC desde Hex-STRING 3 - Entero desde BITS			0, 1, 2, 3

[1] expresión regular
 [2] cadena
 [3] JSONPath o XML XPath
 [4] entero positivo (con compatibilidad con sufijos de tiempo, por ejemplo, 30s, 1m, 2h, 1d)
 [5] macro de usuario
 [6] patrón de Prometheus conforme a la sintaxis: `<metric name>{<label name>="<label value>", ...} == <value>`. Cada componente del patrón de Prometheus (métrica, nombre de etiqueta, valor de etiqueta y valor de métrica) puede ser una macro de usuario.
 [7] Admite varios registros de "Nombre del campo,Prefijo OID,Registros de formato" delimitados por un carácter de nueva línea.

Reglas de anulación de LLD

El objeto de reglas de anulación de LLD define un conjunto de reglas (filtros, condiciones y operaciones) que se utilizan para anular las propiedades de diferentes objetos prototipo. Tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
name	string	Nombre único de la anulación. Comportamiento de la propiedad: - *obligatorio*
step	integer	Número de orden único de la anulación. Comportamiento de la propiedad: - *obligatorio*
stop	integer	Detener el procesamiento de las siguientes anulaciones si coincide. Valores posibles: 0 - *(por defecto)* no detener el procesamiento de las anulaciones; 1 - detener el procesamiento de las anulaciones si el filtro coincide.
filter	object	Filtro de anulación.
operations	object/array	Operaciones de anulación.

Filtro de anulación de la regla LLD

El objeto de filtro de anulación de la regla LLD define un conjunto de condiciones que, si coinciden con el objeto descubierto, se aplica la anulación. Tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
conditions	object/array	Conjunto de condiciones de filtro de anulación para usar en la coincidencia de los objetos descubiertos. Las condiciones se ordenarán en el orden de su colocación en la fórmula. Comportamiento de la propiedad: - *requerido*
evaltype	integer	Método de evaluación de la condición de filtro de anulación. Valores posibles: 0 - Y/O; 1 - Y; 2 - O; 3 - Expresión personalizada. Comportamiento de la propiedad: - *requerido*
eval_formula	string	Expresión generada que se utilizará para evaluar las condiciones del filtro de anulación. La expresión contiene IDs que hacen referencia a condiciones específicas del filtro de anulación por su `formulaid`. El valor de `eval_formula` es igual al valor de `formula` para los filtros con una expresión personalizada. Comportamiento de la propiedad: - *solo lectura*
formula	string	Expresión definida por el usuario que se utilizará para evaluar las condiciones de los filtros de anulación con una expresión personalizada. La expresión debe contener IDs que hagan referencia a condiciones específicas del filtro de anulación por su `formulaid`. Los IDs utilizados en la expresión deben coincidir exactamente con los definidos en las condiciones del filtro de anulación: ninguna condición puede quedar sin usar u omitirse. Comportamiento de la propiedad: - *requerido* si `evaltype` está establecido en ”expresión personalizada”

Condición de filtro de anulación de la regla LLD

El objeto de condición de filtro de anulación de la regla LLD define una comprobación independiente que se realiza sobre el valor de una macro LLD. Tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
macro	string	Macro LLD sobre la que realizar la comprobación. Comportamiento de la propiedad: - *obligatorio*
value	string	Valor con el que comparar. Comportamiento de la propiedad: - *obligatorio* si `operator` está establecido en ”coincide con la expresión regular” o ”no coincide con la expresión regular”
formulaid	string	ID único arbitrario que se utiliza para hacer referencia a la condición desde una expresión personalizada. Solo puede contener letras mayúsculas. El ID debe ser definido por el usuario al modificar las condiciones del filtro, pero se generará de nuevo al solicitarlas posteriormente. Comportamiento de la propiedad: - *obligatorio* si `evaltype` del objeto de filtro de anulación de la regla LLD está establecido en ”expresión personalizada”
operator	integer	Operador de condición. Valores posibles: 8 - *(por defecto)* coincide con la expresión regular; 9 - no coincide con la expresión regular; 12 - existe; 13 - no existe.

Operación de anulación de la regla LLD

La operación de anulación de la regla LLD es una combinación de condiciones y acciones a realizar sobre el objeto prototipo. Tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
operationobject	integer	Tipo de objeto descubierto sobre el que realizar la acción. Valores posibles: 0 - Prototipo de item; 1 - Prototipo de trigger; 2 - Prototipo de gráfico; 3 - Prototipo de host; 4 - Prototipo de descubrimiento. Comportamiento de la propiedad: - *requerido*
operator	integer	Operador de la condición de anulación. Valores posibles: 0 - *(por defecto)* igual; 1 - no igual; 2 - contiene; 3 - no contiene; 8 - coincide; 9 - no coincide.
value	string	Patrón para hacer coincidir el nombre del prototipo de item, trigger, gráfico o host dependiendo del objeto seleccionado.
opstatus	object	Objeto de estado de operación de anulación para objetos prototipo de item, trigger y host.
opdiscover	object	Objeto de estado de descubrimiento de operación de anulación (todos los tipos de objeto).
opperiod	object	Objeto de periodo de operación de anulación (intervalo de actualización) para el objeto prototipo de item.

Propiedad	Tipo	Descripción
ophistory	object	Objeto de historial de operación de anulación para el objeto prototipo de item.
optrends	object	Objeto de tendencias de operación de anulación para el objeto prototipo de item.
opseverity	object	Objeto de severidad de operación de anulación para el objeto prototipo de trigger.
optag	object/array	Objeto de etiqueta de operación de anulación para objetos prototipo de trigger y host.
optemplate	object/array	Objeto de template de operación de anulación para el objeto prototipo de host.
opinventory	object	Objeto de inventario de operación de anulación para el objeto prototipo de host.

Estado de la operación de anulación de la regla LLD

Estado de la operación de anulación de la regla LLD que se establece en el objeto descubierto. Tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
status	integer	Anula el estado para el objeto seleccionado. Valores posibles: 0 - Crear habilitado; 1 - Crear deshabilitado. Comportamiento de la propiedad: - *obligatorio*

Operación de descubrimiento de anulación de regla LLD

El estado de descubrimiento de la operación de anulación de la regla LLD que se establece en el objeto descubierto. Tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
discover	integer	Anula el estado de descubrimiento para el objeto seleccionado. Valores posibles: 0 - Sí, continuar descubriendo los objetos; 1 - No, no se descubrirán nuevos objetos y los existentes se marcarán como perdidos. Comportamiento de la propiedad: - *requerido*

Período de operación de la sobrescritura de la regla LLD

El período de operación de la sobrescritura de la regla LLD es un valor de intervalo de actualización que se establece en el item descubierto. Tiene las siguientes propiedades:

Property	Type	Description
delay	string	Sobrescribe el intervalo de actualización del prototipo de item. Acepta un número de segundos (por ejemplo, 1, 60, 300), un valor con un sufijo de tiempo (por ejemplo, 30s, 1m, 2h, 1d) o, de forma opcional, uno o varios intervalos personalizados, todos separados por punto y coma. Los intervalos personalizados pueden ser una combinación de intervalos flexibles y de programación. Acepta macros de usuario o macros LLD. Si se utiliza, el valor debe ser una única macro. No se admiten varias macros ni macros combinadas con texto. Los intervalos flexibles pueden escribirse como dos macros separadas por una barra diagonal (por ejemplo, {$FLEX_INTERVAL}/{$FLEX_PERIOD}). Ejemplo: 1h;wd1-5h9-18;{$Macro1}/1-7,00:00-24:00;0/6-7,12:00-24:00;{$Macr Comportamiento de la propiedad: - *required*

Historial de la operación de anulación de la regla LLD

Valor del historial de la operación de anulación de la regla LLD que se establece en el item descubierto. Tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
history	string	Anula el historial de la plantilla de item, que es una unidad de tiempo de cuánto tiempo se deben almacenar los datos históricos. También acepta macro de usuario y macro LLD. Comportamiento de la propiedad: - *obligatorio*

Tendencias de la operación de anulación de la regla LLD

Valor de tendencias de la operación de anulación de la regla LLD que se establece en el item descubierto. Tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
trends	string	Anula las tendencias del prototipo de item, que es una unidad de tiempo de cuánto tiempo deben almacenarse los datos de tendencias. También acepta macro de usuario y macro LLD. Comportamiento de la propiedad: - *requerido*

Severidad de la operación de anulación de la regla LLD

Valor de severidad de la operación de anulación de la regla LLD que se establece en el trigger descubierto. Tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
severity	integer	Anula la severidad del trigger prototipo. Valores posibles: 0 - *(por defecto)* no clasificado; 1 - información; 2 - advertencia; 3 - media; 4 - alta; 5 - desastre. Comportamiento de la propiedad: - *requerido*

Etiqueta de operación de anulación de regla LLD

El objeto de etiqueta de operación de anulación de regla LLD contiene el nombre y el valor de la etiqueta que se establecen en el objeto descubierto. Tiene las siguientes propiedades:

Property	Type	Description
tag	string	Nuevo nombre de etiqueta. También acepta macro de usuario y macro LLD. Property behavior: - *required*
value	string	Nuevo valor de etiqueta. También acepta macro de usuario y macro LLD.

Plantilla de operación de anulación de regla LLD

Objeto de plantilla de operación de anulación de regla LLD que está vinculado al host descubierto. Tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
templateid	ID	Anula la plantilla de las plantillas vinculadas al prototipo de host. Comportamiento de la propiedad: - *requerido*

Inventario de operación de anulación de regla LLD

Valor del modo de inventario de operación de anulación de regla LLD que se establece en el host descubierto. Tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
inventory_mode	integer	Anula el modo de inventario del prototipo de host. Valores posibles: -1 - deshabilitado; 0 - *(por defecto)* manual; 1 - automático. Comportamiento de la propiedad: - *requerido*

discoveryrule.create

Descripción

```
object discoveryrule.create(object/array lldRules)
```

Este método permite crear nuevas reglas LLD.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object/array)` Reglas LLD a crear.

Además de las propiedades estándar de la regla LLD, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
filter	object	Filtro de regla LLD para la regla LLD.
preprocessing	object/array	Opciones de preprocesamiento de la regla LLD.
lld_macro_paths	object/array	Opciones de lld_macro_path de la regla LLD.
overrides	object/array	Opciones de anulaciones de la regla LLD.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las reglas LLD creadas bajo la propiedad `itemids`. El orden de los IDs devueltos coincide con el orden de las reglas LLD proporcionadas.

Ejemplos

Creando una regla LLD

Cree una regla LLD de agente Zabbix para descubrir sistemas de archivos montados. Las métricas descubiertas se actualizarán cada 30 segundos.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.create",
    "params": {
        "name": "Mounted filesystem discovery",
        "key_": "vfs.fs.discovery",
        "hostid": "10197",
        "type": 0,
        "interfaceid": "112",
        "delay": "30s"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "27665"
        ]
    },
    "id": 1
}
```

Usando un filtro

Cree una regla LLD con un conjunto de condiciones para filtrar los resultados. Las condiciones se agruparán utilizando el operador lógico ”and”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.create",
```

```
    "params": {
        "name": "Regla LLD filtrada",
        "key_": "lld",
        "hostid": "10116",
        "type": 0,
        "interfaceid": "13",
        "delay": "30s",
        "filter": {
            "evaltype": 1,
            "conditions": [
                {
                    "macro": "{#MACRO1}",
                    "value": "@regex1"
                },
                {
                    "macro": "{#MACRO2}",
                    "value": "@regex2",
                    "operator": "9"
                },
                {
                    "macro": "{#MACRO3}",
                    "value": "",
                    "operator": "12"
                },
                {
                    "macro": "{#MACRO4}",
                    "value": "",
                    "operator": "13"
                }
            ]
        }
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "27665"
        ]
    },
    "id": 1
}
```

Creando una regla LLD con rutas de macro

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.create",
    "params": {
        "name": "Regla LLD con rutas de macro LLD",
        "key_": "lld",
        "hostid": "10116",
        "type": 0,
        "interfaceid": "13",
        "delay": "30s",
        "lld_macro_paths": [
            {
                "lld_macro": "{#MACRO1}",
                "path": "$.path.1"
```

```
            },
            {
                "lld_macro": "{#MACRO2}",
                "path": "$.path.2"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "27665"
        ]
    },
    "id": 1
}
```

Usar un filtro de expresión personalizada

Cree una regla LLD con un filtro que utilice una expresión personalizada para evaluar las condiciones. La regla LLD solo debe descubrir objetos cuyo valor de macro "{#MACRO1}" coincida tanto con la expresión regular "regex1" como con "regex2", y el valor de "{#MACRO2}" coincida con "regex3" o "regex4". Los IDs de fórmula "A", "B", "C" y "D" se han elegido arbitrariamente.

Petición:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.create",
    "params": {
        "name": "Regla LLD filtrada",
        "key_": "lld",
        "hostid": "10116",
        "type": 0,
        "interfaceid": "13",
        "delay": "30s",
        "filter": {
            "evaltype": 3,
            "formula": "(A and B) and (C or D)",
            "conditions": [
                {
                    "macro": "{#MACRO1}",
                    "value": "@regex1",
                    "formulaid": "A"
                },
                {
                    "macro": "{#MACRO1}",
                    "value": "@regex2",
                    "formulaid": "B"
                },
                {
                    "macro": "{#MACRO2}",
                    "value": "@regex3",
                    "formulaid": "C"
                },
                {
                    "macro": "{#MACRO2}",
                    "value": "@regex4",
                    "formulaid": "D"
                }
            ]
```

```
        }
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "27665"
        ]
    },
    "id": 1
}
```

Uso de campos de consulta y cabeceras personalizadas

Cree una regla LLD con campos de consulta y cabeceras personalizadas.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.create",
    "params": {
        "hostid": "10257",
        "interfaceid": "5",
        "type": 19,
        "name": "API HTTP agent",
        "key_": "api_discovery_rule",
        "delay": "5s",
        "url": "http://127.0.0.1?discoverer.php",
        "query_fields": [
            {
                "name": "mode",
                "value": "json"
            },
            {
                "name": "elements",
                "value": "2"
            }
        ],
        "headers": [
            {
                "name": "X-Type",
                "value": "api"
            },
            {
                "name": "Authorization",
                "value": "Bearer mF_A.B5f-2.1JcM"
            }
        ],
        "allow_traps": 1,
        "trapper_hosts": "127.0.0.1"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
```

```
            "28336"
        ]
    },
    "id": 1
}
```

Creando una regla LLD con preprocesamiento

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.create",
    "params": {
        "name": "Regla de descubrimiento con preprocesamiento",
        "key_": "lld.with.preprocessing",
        "hostid": "10001",
        "ruleid": "27665",
        "type": 0,
        "delay": "60s",
        "interfaceid": "1155",
        "preprocessing": [
            {
                "type": 20,
                "params": "20",
                "error_handler": 0,
                "error_handler_params": ""
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "44211"
        ]
    },
    "id": 1
}
```

Creación de una regla LLD con sobrescrituras

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.create",
    "params": {
        "name": "Descubrir equipo de base de datos",
        "key_": "lld.with.overrides",
        "hostid": "10001",
        "type": 0,
        "delay": "60s",
        "interfaceid": "1155",
        "overrides": [
            {
                "name": "Descubrir equipo MySQL",
                "step": "1",
                "stop": "1",
                "filter": {
                    "evaltype": "2",
```

```
                "conditions": [
                    {
                        "macro": "{#UNIT.NAME}",
                        "operator": "8",
                        "value": "^mysqld\\.service$"
                    },
                    {
                        "macro": "{#UNIT.NAME}",
                        "operator": "8",
                        "value": "^mariadb\\.service$"
                    }
                ]
            },
            "operations": [
                {
                    "operationobject": "3",
                    "operator": "2",
                    "value": "Database host",
                    "opstatus": {
                        "status": "0"
                    },
                    "optemplate": [
                        {
                            "templateid": "10170"
                        }
                    ],
                    "optag": [
                        {
                            "tag": "database",
                            "value": "mysql"
                        }
                    ]
                }
            ]
        },
        {
            "name": "Descubrir equipo PostgreSQL",
            "step": "2",
            "stop": "1",
            "filter": {
                "evaltype": "0",
                "conditions": [
                    {
                        "macro": "{#UNIT.NAME}",
                        "operator": "8",
                        "value": "^postgresql\\.service$"
                    }
                ]
            },
            "operations": [
                {
                    "operationobject": "3",
                    "operator": "2",
                    "value": "Database host",
                    "opstatus": {
                        "status": "0"
                    },
                    "optemplate": [
                        {
                            "templateid": "10263"
                        }
                    ],
```

```
                        "optag": [
                            {
                                "tag": "database",
                                "value": "postgresql"
                            }
                        ]
                    }
                ]
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "30980"
        ]
    },
    "id": 1
}
```

Crear regla LLD de script

Cree una recopilación de datos simple utilizando una regla LLD de script.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.create",
    "params": {
        "name": "Ejemplo de script",
        "key_": "custom.script.lldrule",
        "hostid": "12345",
        "type": 21,
        "params": "var request = new HttpRequest();\nreturn request.post(\"https://postman-echo.com/post\"
        "parameters": [{
            "name": "host",
            "value": "{HOST.CONN}"
        }],
        "timeout": "6s",
        "delay": "30s"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "23865"
        ]
    },
    "id": 1
}
```

Crear una regla LLD con un período de tiempo especificado para deshabilitar y sin eliminación

Cree una regla LLD con un período de tiempo personalizado para deshabilitar la entidad después de que ya no se descubra, con la configuración de que nunca se eliminará.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.create",
    "params": {
        "name": "lld disable after 1h",
        "key_": "lld.disable",
        "hostid": "10001",
        "type": 2,
        "lifetime_type": 1,
        "enabled_lifetime_type": 0,
        "enabled_lifetime": "1h"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "46864"
        ]
    },
    "id": 1
}
```

Ver también

- Filtro de regla LLD
- Rutas de macro LLD
- Preprocesamiento de regla LLD

Fuente

CDiscoveryRule::create() en *ui/include/classes/api/services/CDiscoveryRule.php*.

discoveryrule.delete

Descripción

`object discoveryrule.delete(array lldRuleIds)`

Este método permite eliminar reglas de descubrimiento de bajo nivel (LLD).

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.
 El usuario no podrá eliminar una regla LLD que descubra entidades mediante un proxy inaccesible. Consulte configuración del grupo de usuarios (*Lista de acceso al proxy*) para obtener más información.

Parámetros

`(array)` IDs de las reglas LLD a eliminar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las reglas LLD eliminadas bajo la propiedad `ruleids`.

Ejemplos

Eliminando múltiples reglas LLD

Eliminar dos reglas LLD.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.delete",
    "params": [
        "27665",
        "27668"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "ruleids": [
            "27665",
            "27668"
        ]
    },
    "id": 1
}
```

Fuente

CDiscoveryRule::delete() en *ui/include/classes/api/services/CDiscoveryRule.php*.

discoveryrule.get

Descripción

`integer/array discoveryrule.get(object parameters)`

El método permite recuperar reglas de LLD según los parámetros proporcionados.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
itemids	ID/array	Devolver solo las reglas LLD con los IDs dados.
groupids	ID/array	Devolver solo las reglas LLD que pertenecen a los hosts de los grupos dados.
hostids	ID/array	Devolver solo las reglas LLD que pertenecen a los hosts dados.
inherited	boolean	Si se establece en `true`, devolver solo las reglas LLD heredadas de un template.
interfaceids	ID/array	Devolver solo las reglas LLD que usan las interfaces de host dadas.
monitored	boolean	Si se establece en `true`, devolver solo las reglas LLD habilitadas que pertenecen a hosts monitorizados.
templated	boolean	Si se establece en `true`, devolver solo las reglas LLD que pertenecen a templates.
templateids	ID/array	Devolver solo las reglas LLD que pertenecen a los templates dados.
selectDiscoveryData	query	Devolver una propiedad `discoveryData` con los datos del objeto de descubrimiento de la regla LLD. El objeto de descubrimiento de la regla LLD vincula una regla LLD descubierta a una regla LLD prototipo de la que fue descubierta.

Parámetro	Tipo	Descripción
selectDiscoveryRulePrototypes	query	Devolver una propiedad `discoveryRulePrototypes` con las reglas LLD prototipo que pertenecen a la regla LLD. Admite `count`.
selectFilter	query	Devolver una propiedad `filter` con los datos del filtro utilizado por la regla LLD.
selectGraphs	query	Devuelve una propiedad `graphs` con los gráficos prototipo que pertenecen a la regla LLD. Admite `count`.
selectHostPrototypes	query	Devolver una propiedad `hostPrototypes` con los hosts prototipo que pertenecen a la regla LLD. Admite `count`.
selectHosts	query	Devolver una propiedad `hosts` con un array de hosts a los que pertenece la regla LLD.
selectItems	query	Devolver una propiedad `items` con los items prototipo que pertenecen a la regla LLD. Admite `count`.
selectTriggers	query	Devolver una propiedad `triggers` con los triggers prototipo que pertenecen a la regla LLD. Admite `count`.
selectLLDMacroPaths	query	Devolver una propiedad `lld_macro_paths` con una lista de macros LLD y rutas a los valores asignados a cada macro correspondiente.
selectPreprocessing	query	Devolver una propiedad `preprocessing` con las opciones de preprocesamiento de la regla LLD.
selectOverrides	query	Devolver una propiedad `lld_rule_overrides` con una lista de filtros de override, condiciones y operaciones que se realizan en los objetos prototipo.
filter	object	Devolver solo aquellos resultados que coincidan exactamente con el filtro dado. Acepta un objeto, donde las claves son nombres de propiedades y los valores son un único valor o un array de valores con los que comparar. No admite propiedades de tipo de dato `text` data type. Admite propiedades adicionales: `host` - nombre técnico del host al que pertenece la regla LLD.
limitSelects	integer	Limita el número de registros devueltos por subconsultas. Se aplica a las siguientes subconsultas: `selectItems`, `selectGraphs`, `selectTriggers`.
sortfield	string/array	Ordenar el resultado por las propiedades dadas. Valores posibles: `itemid`, `name`, `key_`, `delay`, `type`, `status`.
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
editable	boolean	
excludeSearch	boolean	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

(entero/array) Devuelve:

- un array de objetos
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperar reglas de descubrimiento de un host

Recupera todas las reglas de descubrimiento para un ID de host específico.

Solicitud:


```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.get",
    "params": {
        "output": "extend",
        "hostids": "10202"
    },
    "id": 1
}
```


Respuesta:


```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "itemid": "27425",
            "type": "0",
            "snmp_oid": "",
            "hostid": "10202",
            "name": "Descubrimiento de interfaz de red",
            "key_": "net.if.discovery",
            "delay": "1h",
            "status": "0",
            "trapper_hosts": "",
            "templateid": "22444",
            "valuemapid": "0",
            "params": "",
            "ipmi_sensor": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "flags": "1",
            "interfaceid": "119",
            "description": "Descubrimiento de interfaces de red según lo definido en la expresión regular
            "lifetime": "30d",
            "jmx_endpoint": "",
            "master_itemid": "0",
            "timeout": "",
            "url": "",
            "query_fields": [],
            "posts": "",
            "status_codes": "200",
            "follow_redirects": "1",
            "post_type": "0",
            "http_proxy": "",
            "headers": [],
            "retrieve_mode": "0",
            "request_method": "0",
            "output_format": "0",
            "ssl_cert_file": "",
            "ssl_key_file": "",
```

```
            "ssl_key_password": "",
            "verify_peer": "0",
            "verify_host": "0",
            "allow_traps": "0",
            "uuid": "",
            "lifetime_type": "0",
            "enabled_lifetime_type": "2",
            "enabled_lifetime": "0",
            "state": "0",
            "error": "",
            "parameters": []
        },
        {
            "itemid": "27426",
            "type": "0",
            "snmp_oid": "",
            "hostid": "10202",
            "name": "Descubrimiento de sistemas de archivos montados",
            "key_": "vfs.fs.discovery",
            "delay": "1h",
            "status": "0",
            "trapper_hosts": "",
            "templateid": "22450",
            "valuemapid": "0",
            "params": "",
            "ipmi_sensor": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "flags": "1",
            "interfaceid": "119",
            "description": "Descubrimiento de sistemas de archivos de diferentes tipos según lo definido e
            "lifetime": "30d",
            "jmx_endpoint": "",
            "master_itemid": "0",
            "timeout": "",
            "url": "",
            "query_fields": [],
            "posts": "",
            "status_codes": "200",
            "follow_redirects": "1",
            "post_type": "0",
            "http_proxy": "",
            "headers": [],
            "retrieve_mode": "0",
            "request_method": "0",
            "output_format": "0",
            "ssl_cert_file": "",
            "ssl_key_file": "",
            "ssl_key_password": "",
            "verify_peer": "0",
            "verify_host": "0",
            "allow_traps": "0",
            "uuid": "",
            "lifetime_type": "0",
            "enabled_lifetime_type": "2",
            "enabled_lifetime": "0",
            "state": "0",
            "error": "",
            "parameters": []
```

```
        }
    ],
    "id": 1
}
```

Recuperando condiciones de filtro

Recupere el nombre de la regla LLD "24681" y sus condiciones de filtro. El filtro utiliza el tipo de evaluación "and", por lo que la propiedad `formula` está vacía y `eval_formula` se genera automáticamente.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.get",
    "params": {
        "output": ["name"],
        "selectFilter": "extend",
        "itemids": ["24681"]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "itemid": "24681",
            "name": "Filtered LLD rule",
            "filter": {
                "evaltype": "1",
                "formula": "",
                "conditions": [
                    {
                        "macro": "{#MACRO1}",
                        "value": "@regex1",
                        "operator": "8",
                        "formulaid": "A"
                    },
                    {
                        "macro": "{#MACRO2}",
                        "value": "@regex2",
                        "operator": "9",
                        "formulaid": "B"
                    },
                    {
                        "macro": "{#MACRO3}",
                        "value": "",
                        "operator": "12",
                        "formulaid": "C"
                    },
                    {
                        "macro": "{#MACRO4}",
                        "value": "",
                        "operator": "13",
                        "formulaid": "D"
                    }
                ],
                "eval_formula": "A and B and C and D"
            }
        }
    ],
    "id": 1
```

```
}
```

Recuperar la regla LLD por URL

Recupera la regla LLD para el host por el valor del campo URL de la regla. Solo se admite la coincidencia exacta de la cadena URL definida para la regla LLD.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.get",
    "params": {
        "hostids": "10257",
        "filter": {
            "type": 19,
            "url": "http://127.0.0.1/discoverer.php"
        }
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "itemid": "28336",
            "type": "19",
            "snmp_oid": "",
            "hostid": "10257",
            "name": "API HTTP agent",
            "key_": "api_discovery_rule",
            "delay": "5s",
            "status": "0",
            "trapper_hosts": "",
            "templateid": "0",
            "valuemapid": "0",
            "params": "",
            "ipmi_sensor": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "flags": "1",
            "interfaceid": "5",
            "description": "",
            "lifetime": "30d",
            "jmx_endpoint": "",
            "master_itemid": "0",
            "timeout": "",
            "url": "http://127.0.0.1/discoverer.php",
            "query_fields": [
                {
                    "name": "mode",
                    "value": "json"
                },
                {
                    "name": "elements",
                    "value": "2"
                }
            ],
            "posts": "",
```

```
            "status_codes": "200",
            "follow_redirects": "1",
            "post_type": "0",
            "http_proxy": "",
            "headers": [
                {
                    "name" : "X-Type",
                    "value": "api"
                },
                {
                    "name": "Authorization",
                    "value": "Bearer mF_A.B5f-2.1JcM"
                }
            ],
            "retrieve_mode": "0",
            "request_method": "1",
            "output_format": "0",
            "ssl_cert_file": "",
            "ssl_key_file": "",
            "ssl_key_password": "",
            "verify_peer": "0",
            "verify_host": "0",
            "allow_traps": "0",
            "uuid": "",
            "lifetime_type": "0",
            "enabled_lifetime_type": "2",
            "enabled_lifetime": "0",
            "state": "0",
            "error": "",
            "parameters": []
        }
    ],
    "id": 1
}
```

Recuperar regla LLD con sobrescrituras

Recuperar una regla LLD que tiene varias configuraciones de sobrescritura.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.get",
    "params": {
        "output": ["name"],
        "itemids": "30980",
        "selectOverrides": ["name", "step", "stop", "filter", "operations"]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "name": "Descubrir equipo de base de datos",
            "overrides": [
                {
                    "name": "Descubrir equipo MySQL",
                    "step": "1",
                    "stop": "1",
                    "filter": {
```

```
                "evaltype": "2",
                "formula": "",
                "conditions": [
                    {
                        "macro": "{#UNIT.NAME}",
                        "operator": "8",
                        "value": "^mysqld\\.service$",
                        "formulaid": "A"
                    },
                    {
                        "macro": "{#UNIT.NAME}",
                        "operator": "8",
                        "value": "^mariadb\\.service$",
                        "formulaid": "B"
                    }
                ],
                "eval_formula": "A or B"
            },
            "operations": [
                {
                    "operationobject": "3",
                    "operator": "2",
                    "value": "Equipo de base de datos",
                    "opstatus": {
                        "status": "0"
                    },
                    "optag": [
                        {
                            "tag": "database",
                            "value": "mysql"
                        }
                    ],
                    "optemplate": [
                        {
                            "templateid": "10170"
                        }
                    ]
                }
            ]
        },
        {
            "name": "Descubrir equipo PostgreSQL",
            "step": "2",
            "stop": "1",
            "filter": {
                "evaltype": "0",
                "formula": "",
                "conditions": [
                    {
                        "macro": "{#UNIT.NAME}",
                        "operator": "8",
                        "value": "^postgresql\\.service$",
                        "formulaid": "A"
                    }
                ],
                "eval_formula": "A"
            },
            "operations": [
                {
                    "operationobject": "3",
                    "operator": "2",
                    "value": "Equipo de base de datos",
```

```
                            "opstatus": {
                                "status": "0"
                            },
                            "optag": [
                                {
                                    "tag": "database",
                                    "value": "postgresql"
                                }
                            ],
                            "optemplate": [
                                {
                                    "templateid": "10263"
                                }
                            ]
                        }
                    ]
                }
            ]
        }
    ],
    "id": 1
}
```


Ver también

- Prototipo de gráfico
- Host
- Prototipo de item
- Filtro de regla LLD
- Prototipo de trigger

Fuente

CDiscoveryRule::get() en *ui/include/classes/api/services/CDiscoveryRule.php*.

discoveryrule.update

Descripción

`object discoveryrule.update(object/array lldRules)`

Este método le permite actualizar reglas LLD existentes.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object/array)` Propiedades de la regla LLD que se actualizarán.

La propiedad `itemid` debe estar definida para cada regla LLD, todas las demás propiedades son opcionales. Sólo se actualizarán las propiedades pasadas, todas las demás permanecerán sin cambios.

Además de las propiedades estándar de la regla LLD, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
filter	object	Filtro de regla LLD para reemplazar el filtro existente.
preprocessing	object/array	Opciones de preprocesamiento de la regla LLD para reemplazar las opciones de preprocesamiento existentes. Comportamiento del parámetro: - *sólo lectura* para objetos heredados

Parámetro	Tipo	Descripción
lld_macro_paths	object/array	Opciones de lld_macro_path de la regla LLD para reemplazar las opciones de lld_macro_path existentes. Comportamiento del parámetro: - *sólo lectura* para objetos heredados
overrides	object/array	Opciones de overrides de la regla LLD para reemplazar las opciones de overrides existentes. Comportamiento del parámetro: - *sólo lectura* para objetos heredados

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las reglas LLD actualizadas bajo la propiedad `itemids`.

Ejemplos

Agregar un filtro a una regla LLD

Agregue un filtro para que el contenido de la macro *{#FSTYPE}* coincida con la expresión regular *@File systems for discovery*.

Request:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.update",
    "params": {
        "itemid": "22450",
        "filter": {
            "evaltype": 1,
            "conditions": [
                {
                    "macro": "{#FSTYPE}",
                    "value": "@File systems for discovery"
                }
            ]
        }
    },
    "id": 1
}
```

Response:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "22450"
        ]
    },
    "id": 1
}
```

Añadiendo rutas de macro LLD

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.update",
    "params": {
        "itemid": "22450",
        "lld_macro_paths": [
            {
                "lld_macro": "{#MACRO1}",
                "path": "$.json.path"
```

```
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "22450"
        ]
    },
    "id": 1
}
```

Deshabilitar trapping

Deshabilitar trapping de LLD para la regla de descubrimiento.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.update",
    "params": {
        "itemid": "28336",
        "allow_traps": 0
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "28336"
        ]
    },
    "id": 1
}
```

Actualización de las opciones de preprocesamiento de la regla LLD

Actualizar una regla LLD con la regla de preprocesamiento ”JSONPath”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.update",
    "params": {
        "itemid": "44211",
        "preprocessing": [
            {
                "type": 12,
                "params": "$.path.to.json",
                "error_handler": 2,
                "error_handler_params": "5"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "44211"
        ]
    },
    "id": 1
}
```

Actualización del script de la regla LLD

Actualice un script de regla LLD con un script diferente y elimine los parámetros innecesarios que fueron utilizados por el script anterior.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.update",
    "params": {
        "itemid": "23865",
        "parameters": [],
        "script": "Zabbix.log(3, 'Log test');\nreturn 1;"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "23865"
        ]
    },
    "id": 1
}
```

Actualizar la vida útil de la regla LLD

Actualizar la regla LLD para deshabilitar la entidad que ya no se descubre después de 12 horas y eliminarla después de 7 días.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.update",
    "params": {
        "itemid": "46864",
        "lifetime_type": 0,
        "lifetime": "7d",
        "enabled_lifetime_type": 0,
        "enabled_lifetime": "12h"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "46864"
        ]
```

```
    },
    "id": 1
}
```

Fuente

CDiscoveryRule::update() en *ui/include/classes/api/services/CDiscoveryRule.php*.

Reporte

Esta clase está diseñada para trabajar con reportes programados.

Referencias de objetos:

- Reporte
- Usuarios
- Grupos de usuarios

Métodos disponibles:

- report.create - crear nuevos reportes programados
- report.delete - eliminar reportes programados
- report.get - recuperar reportes programados
- report.update - actualizar reportes programados

Objeto report

Los siguientes objetos están directamente relacionados con la API `report`.

Informe

El objeto de informe tiene las siguientes propiedades:

Property	Type	Description
reportid	ID	ID del informe. Comportamiento de la propiedad: - *solo lectura* - *obligatoria* para las operaciones de actualización
userid	ID	ID del usuario que creó el informe. Comportamiento de la propiedad: - *obligatoria* para las operaciones de creación
name	string	Nombre único del informe. Comportamiento de la propiedad: - *obligatoria* para las operaciones de creación
dashboardid	ID	ID del dashboard en el que se basa el informe. Comportamiento de la propiedad: - *obligatoria* para las operaciones de creación
period	integer	Período para el que se preparará el informe. Valores posibles: 0 - *(predeterminado)* día anterior; 1 - semana anterior; 2 - mes anterior; 3 - año anterior.
cycle	integer	Programación de repetición del período. Valores posibles: 0 - *(predeterminado)* diariamente; 1 - semanalmente; 2 - mensualmente; 3 - anualmente.

Property	Type	Description
start_time	integer	Hora del día, en segundos, a la que se preparará el informe para su envío. Predeterminado: 0.
weekdays	integer	Días de la semana para enviar el informe. Valores de bitmap posibles: 1 - lunes; 2 - martes; 4 - miércoles; 8 - jueves; 16 - viernes; 32 - sábado; 64 - domingo. Este es un campo de máscara de bits; se acepta cualquier suma de los valores de bitmap posibles (por ejemplo, 21 para lunes, miércoles y viernes). Predeterminado: 0. Comportamiento de la propiedad: - *obligatoria* si `cycle` está establecido en ”weekly”.
active_since	string	Fecha a partir de la cual se iniciará. Valores posibles: cadena vacía - *(predeterminado)* no especificada (se almacena como 0); fecha específica en formato YYYY-MM-DD (se almacena como una marca de tiempo del inicio del día (00:00:00)).
active_till	string	Fecha en la que finalizará. Valores posibles: cadena vacía - *(predeterminado)* no especificada (se almacena como 0); fecha específica en formato YYYY-MM-DD (se almacena como una marca de tiempo del final del día (23:59:59)).
subject	string	Asunto del mensaje del informe.
message	string	Texto del mensaje del informe.
status	integer	Indica si el informe está habilitado o deshabilitado. Valores posibles: 0 - *(predeterminado)* habilitado; 1 - deshabilitado.
description	text	Descripción del informe.
state	integer	Estado del informe. Valores posibles: 0 - *(predeterminado)* el informe aún no se ha procesado; 1 - el informe se generó y se envió correctamente a todos los destinatarios; 2 - no se pudo generar el informe; ”info” contiene información sobre el error; 3 - el informe se generó, pero no se pudo enviar a algunos (o a todos) los destinatarios; ”info” contiene información sobre el error. Comportamiento de la propiedad: - *solo lectura*
lastsent	timestamp	Marca de tiempo de Unix del último informe enviado correctamente. Comportamiento de la propiedad: - *solo lectura*

Property	Type	Description
info	string	Descripción del error o información adicional. Comportamiento de la propiedad: - *solo lectura*

Usuarios

El objeto usuarios tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
userid	ID	ID del usuario al que se enviará el informe. Comportamiento de la propiedad: - *obligatorio*
access_userid	ID	ID del usuario en cuyo nombre se generará el informe. 0 - *(por defecto)* Generar informe por destinatario.
exclude	integer	Si se debe excluir al usuario de la lista de correo. Valores posibles: 0 - *(por defecto)* Incluir; 1 - Excluir.

Grupos de usuarios

El objeto de grupos de usuarios tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
usrgrpid	ID	ID del grupo de usuarios al que se enviará el informe. Comportamiento de la propiedad: - *obligatorio*
access_userid	ID	ID del usuario en cuyo nombre se generará el informe. 0 - *(por defecto)* Generar informe por destinatario.

report.create

Descripción

`object report.create(object/array reports)`

Este método permite crear nuevos informes programados.

Note:
Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object/array)` Informes programados a crear.

Además de las propiedades estándar del informe programado, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
users	object/array	Usuarios a los que enviar el informe. Comportamiento del parámetro: - *obligatorio* si `user_groups` no está establecido

Parámetro	Tipo	Descripción
user_groups	object/array	Grupos de usuarios a los que enviar el informe. Comportamiento del parámetro: - *obligatorio* si `users` no está establecido

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los informes programados creados bajo la propiedad `reportids`. El orden de los IDs devueltos coincide con el orden de los informes programados enviados.

Ejemplos

Creación de un informe programado

Cree un informe semanal que se preparará para la semana anterior todos los lunes-viernes a las 12:00 desde 2021-04-01 hasta 2021-08-31.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "report.create",
    "params": {
        "userid": "1",
        "name": "Weekly report",
        "dashboardid": "1",
        "period": "1",
        "cycle": "1",
        "start_time": "43200",
        "weekdays": "31",
        "active_since": "2021-04-01",
        "active_till": "2021-08-31",
        "subject": "Weekly report",
        "message": "Report accompanying text",
        "status": "1",
        "description": "Report description",
        "users": [
            {
                "userid": "1",
                "access_userid": "1",
                "exclude": "0"
            },
            {
                "userid": "2",
                "access_userid": "0",
                "exclude": "1"
            }
        ],
        "user_groups": [
            {
                "usrgrpid": "7",
                "access_userid": "0"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "reportids": [
```

```
            "1"
        ]
    },
    "id": 1
}
```

Ver también

- Usuarios
- Grupos de usuarios

Fuente

CReport::create() en *ui/include/classes/api/services/CReport.php*.

report.delete

Descripción

`object report.delete(array reportids)`

Este método le permite eliminar informes programados.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración del rol de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(array)` IDs de los informes programados a eliminar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los informes programados eliminados bajo la propiedad `reportids`.

Ejemplos

Eliminando múltiples informes programados

Eliminar dos informes programados.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "report.delete",
    "params": [
        "1",
        "2"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "reportids": [
            "1",
            "2"
        ]
    },
    "id": 1
}
```

Fuente

CReport::delete() en *ui/include/classes/api/services/CReport.php*.

report.get

Descripción

`integer/array report.get(object parameters)`

El método permite recuperar informes programados según los parámetros proporcionados.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
reportids	ID/array	Devolver solo los informes programados con los IDs de informe dados.
expired	boolean	Si se establece en `true`, devuelve solo los informes programados caducados.
selectUsers	query	Devuelve una propiedad `users` con los usuarios a los que se ha configurado el envío del informe.
selectUserGroups	query	Devuelve una propiedad `user_groups` con los grupos de usuarios a los que se ha configurado el envío del informe.
sortfield	string/array	Ordena el resultado por las propiedades indicadas. Valores posibles: `reportid`, `name`, `status`.
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperación de datos de informes

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "report.get",
    "params": {
        "output": "extend",
        "selectUsers": "extend",
        "selectUserGroups": "extend",
        "reportids": ["1", "2"]
    },
    "id": 1
}
```

Respuesta:


```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "reportid": "1",
            "userid": "1",
            "name": "Informe semanal",
            "dashboardid": "1",
            "period": "1",
            "cycle": "1",
            "start_time": "43200",
            "weekdays": "31",
            "active_since": "2021-04-01",
            "active_till": "2021-08-31",
            "subject": "Informe semanal",
            "message": "Texto adjunto al informe",
            "status": "1",
            "description": "Descripción del informe",
            "state": "1",
            "lastsent": "1613563219",
            "info": "",
            "users": [
                {
                    "userid": "1",
                    "access_userid": "1",
                    "exclude": "0"
                },
                {
                    "userid": "2",
                    "access_userid": "0",
                    "exclude": "1"
                }
            ],
            "user_groups": [
                {
                    "usrgrpid": "7",
                    "access_userid": "0"
                }
            ]
        },
        {
            "reportid": "2",
            "userid": "1",
            "name": "Informe mensual",
            "dashboardid": "2",
            "period": "2",
            "cycle": "2",
            "start_time": "0",
            "weekdays": "0",
            "active_since": "2021-05-01",
            "active_till": "",
            "subject": "Informe mensual",
            "message": "Texto adjunto al informe",
            "status": "1",
            "description": "",
            "state": "0",
            "lastsent": "0",
            "info": "",
            "users": [
                {
                    "userid": "1",
```

```
                    "access_userid": "1",
                    "exclude": "0"
                }
            ],
            "user_groups": []
        }
    ],
    "id": 1
}
```

Ver también

- Usuarios
- Grupos de usuarios

Fuente

CReport::get() en *ui/include/classes/api/services/CReport.php*.

report.update

Descripción

`object report.update(object/array reports)`

Este método permite actualizar informes programados existentes.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración del rol de usuario. Consulte User roles para obtener más información.

Parámetros

`(object/array)` Propiedades del informe programado que se actualizarán.

La propiedad `reportid` debe estar definida para cada informe programado, todas las demás propiedades son opcionales. Sólo se actualizarán las propiedades pasadas, todas las demás permanecerán sin cambios.

Además de las propiedades estándar del informe programado, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
users	object/array	Usuarios que reemplazarán a los usuarios actuales asignados al informe programado. Comportamiento del parámetro: - *obligatorio* si `user_groups` no está definido
user_groups	object/array	Grupos de usuarios que reemplazarán a los grupos de usuarios actuales asignados al informe programado. Comportamiento del parámetro: - *obligatorio* si `users` no está definido

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los informes programados actualizados bajo la propiedad `reportids`.

Ejemplos

Deshabilitar informe programado

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "report.update",
    "params": {
        "reportid": "1",
```

```
        "status": "0"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "reportids": [
            "1"
        ]
    },
    "id": 1
}
```

Ver también

- Usuarios
- Grupos de usuarios

Fuente

CReport::update() en *ui/include/classes/api/services/CReport.php*.

Rol

Esta clase está diseñada para trabajar con roles de usuario.

Referencias de objetos:

- Rol
- Reglas de rol
  - Elemento de la interfaz
  - Servicio
  - Etiqueta de servicio
  - Módulo
  - Acción

Métodos disponibles:

- role.create - crear nuevos roles de usuario
- role.delete - eliminar roles de usuario
- role.get - obtener roles de usuario
- role.update - actualizar roles de usuario

Objeto de rol

Los siguientes objetos están directamente relacionados con la API de `role`.

Rol

El objeto rol tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
roleid	ID	ID del rol. Comportamiento de la propiedad: - *solo lectura* - *requerido* para operaciones de actualización
name	string	Nombre del rol. Comportamiento de la propiedad: - *requerido* para operaciones de creación

Propiedad	Tipo	Descripción
type	integer	Tipo de usuario. Valores posibles: 1 - *(por defecto)* Usuario; 2 - Admin; 3 - Super admin. Comportamiento de la propiedad: - *requerido* para operaciones de creación
readonly	integer	Si el rol es de solo lectura. Valores posibles: 0 - *(por defecto)* No; 1 - Sí. Comportamiento de la propiedad: - *solo lectura*

Reglas de rol

El objeto de reglas de rol tiene las siguientes propiedades:

Property	Type	Description
ui	array	Matriz de objetos de elementos de la interfaz de usuario.
ui.default_access	integer	Indica si el acceso a los nuevos elementos de la interfaz de usuario está habilitado. Valores posibles: 0 - Deshabilitado; 1 - *(predeterminado)* Habilitado.
profile.redirect.enforce	integer	Indica si se aplica la URL de redirección a nivel de rol después de la autenticación y si el usuario no puede anularla. Valores posibles: 0 - *(predeterminado)* Deshabilitado; 1 - Habilitado.
profile.redirect.url	string	URL utilizada para redirigir al usuario después de la autenticación.
services.read.mode	integer	Acceso de solo lectura a los servicios. Valores posibles: 0 - Acceso de solo lectura a los servicios especificados por las propiedades `services.read.list` o coincidentes con `services.read.tag`; 1 - *(predeterminado)* Acceso de solo lectura a todos los servicios.
services.read.list	array	Matriz de objetos Service. Los servicios especificados, incluidos los servicios secundarios, tendrán acceso de solo lectura para el rol de usuario. El acceso de solo lectura no anulará el acceso de lectura y escritura a los servicios. Comportamiento de la propiedad: - *compatible* si `services.read.mode` está establecido en ”0”.

Property	Type	Description
services.read.tag	object	Matriz de objetos Service tag. Los servicios coincidentes con la etiqueta, incluidos los servicios secundarios, tendrán acceso de solo lectura para el rol de usuario. El acceso de solo lectura no anulará el acceso de lectura y escritura a los servicios. Comportamiento de la propiedad: - *compatible* si `services.read.mode` está establecido en ”0”.
services.write.mode	integer	Acceso de lectura y escritura a los servicios. Valores posibles: 0 - *(predeterminado)* Acceso de lectura y escritura a los servicios especificados por las propiedades `services.write.list` o coincidentes con `services.write.tag`; 1 - Acceso de lectura y escritura a todos los servicios.
services.write.list	array	Matriz de objetos Service. Los servicios especificados, incluidos los servicios secundarios, tendrán acceso de lectura y escritura para el rol de usuario. El acceso de lectura y escritura anulará el acceso de solo lectura a los servicios. Comportamiento de la propiedad: - *compatible* si `services.write.mode` está establecido en ”0”.
services.write.tag	object	Matriz de objetos Service tag. Los servicios coincidentes con la etiqueta, incluidos los servicios secundarios, tendrán acceso de lectura y escritura para el rol de usuario. El acceso de lectura y escritura anulará el acceso de solo lectura a los servicios. Comportamiento de la propiedad: - *compatible* si `services.write.mode` está establecido en ”0”.
modules	array	Matriz de objetos de módulos.
modules.default_access	integer	Indica si el acceso a los nuevos módulos está habilitado. Valores posibles: 0 - Deshabilitado; 1 - *(predeterminado)* Habilitado.
api.access	integer	Indica si el acceso a la API está habilitado. Valores posibles: 0 - *(predeterminado)* Deshabilitado; 1 - Habilitado.
api.mode	integer	Modo para tratar los métodos de la API enumerados en la propiedad `api`. Valores posibles: 0 - *(predeterminado)* Lista de denegación; 1 - Lista de permitidos. Comportamiento de la propiedad: - *compatible* si `api.access` está establecido en ”1”.

Property	Type	Description
api	array	Matriz de métodos de la API. Se admiten comodines, por ejemplo: `*` (cualquier método), `dashboard.*`, `*.export`. Si no se especifican métodos, se deniega el acceso a todos los métodos de la API si `api.mode` está establecido en ”1” (Lista de permitidos), y se permite si `api.mode` está establecido en ”0” (Lista de denegación). Comportamiento de la propiedad: - *compatible* si `api.access` está establecido en ”1”.
actions	array	Matriz de objetos de acciones.
actions.default_access	integer	Indica si el acceso a las nuevas acciones está habilitado. Valores posibles: 0 - Deshabilitado; 1 - *(predeterminado)* Habilitado.
devices.access	integer	Indica si está habilitado el acceso a los dispositivos móviles propios del usuario para utilizarlos con la aplicación móvil de Zabbix. Valores posibles: 0 - (*predeterminado para User y Admin*) Deshabilitado; 1 - (*predeterminado para Super admin*) Habilitado. Comportamiento de la propiedad: - *compatible* si `EnableMobileDevices` está habilitado
devices.actions	array	Matriz de objetos de acciones de dispositivo. Comportamiento de la propiedad: - *compatible* si `devices.access` está establecido en ”1”
devices.actions.default_access	integer	Indica si el acceso a las nuevas acciones de dispositivo está habilitado. Valores posibles: 0 - (*predeterminado para User y Admin*) Deshabilitado; 1 - (*predeterminado para Super admin*) Habilitado. Comportamiento de la propiedad: - *compatible* si `EnableMobileDevices` está habilitado

Elemento de la interfaz de usuario

El objeto de elemento de la interfaz de usuario tiene las siguientes propiedades:

Property	Type	Description
name	string	Nombre del elemento de la interfaz de usuario. Valores posibles si `type` del objeto Role está establecido en ”User”, ”Admin” o ”Super admin”: `monitoring.dashboard` - *Dashboards*; `monitoring.problems` - *Monitoring > Problems*; `monitoring.hosts` - *Monitoring > Hosts*; `monitoring.latest_data` - *Monitoring > Latest data*; `monitoring.maps` - *Monitoring > Maps*; `services.services` - *Services > Services*; `services.sla_report` - *Services > SLA report*; `inventory.overview` - *Inventory > Overview*; `inventory.hosts` - *Inventory > Hosts*; `reports.availability_report` - *Reports > Availability report*; `reports.top_triggers` - *Reports > Triggers top 100*. Valores posibles si `type` del objeto Role está establecido en ”Admin” o ”Super admin”: `monitoring.discovery` - *Monitoring > Discovery*; `services.sla` - *Services > SLA*; `reports.scheduled_reports` - *Reports > Scheduled reports*; `reports.notifications` - *Reports > Notifications*; `configuration.template_groups` - *Data collection > Template groups*; `configuration.host_groups` - *Data collection > Host groups*; `configuration.templates` - *Data collection > Templates*; `configuration.hosts` - *Data collection > Hosts*; `configuration.maintenance` - *Data collection > Maintenance*; `configuration.discovery` - *Data collection > Discovery*; `configuration.trigger_actions` - *Alerts > Actions > Trigger actions*; `configuration.service_actions` - *Alerts > Actions > Service actions*; `configuration.discovery_actions` - *Alerts > Actions > Discovery actions*; `configuration.autoregistration_actions` - *Alerts > Actions > Autoregistration actions*; `configuration.internal_actions` - *Alerts > Actions > Internal actions*. Valores posibles si `type` del objeto Role está establecido en ”Super admin”: `reports.system_info` - *Reports > System information*; `reports.audit` - *Reports > Audit log*; `reports.action_log` - *Reports > Action log*; `configuration.event_correlation` - *Data collection > Event correlation*; `administration.media_types` - *Alerts > Media types*; `administration.scripts` - *Alerts > Scripts*; `administration.user_groups` - *Users > User groups*; `administration.user_roles` - *Users > User roles*; `administration.users` - *Users > Users*; `administration.api_tokens` - *Users > API tokens*; `administration.authentication` - *Users > Authentication*; `administration.linked_devices` - *Users > Devices*; `administration.general` - *Administration > General*; `administration.audit_log` - *Administration > Audit log*; `administration.housekeeping` - *Administration > Housekeeping*; `administration.proxy_groups` - *Administration > Proxy groups*; `administration.proxies` - *Administration > Proxies*; `administration.macros` - *Administration > Macros*; `administration.queue` - *Administration > Queue*. Comportamiento de la propiedad: *- required*

Property	Type	Description
status	integer	Indica si el acceso al elemento de la interfaz de usuario está habilitado. Valores posibles: 0 - Deshabilitado; 1 - *(default)* Habilitado.

Servicio

Propiedad	Tipo	Descripción
serviceid	ID	ID del servicio. Comportamiento de la propiedad: - *requerido*

Etiqueta de servicio

Propiedad	Tipo	Descripción
tag	string	Nombre de la etiqueta. Si se especifica una cadena vacía, la etiqueta de servicio no se utilizará para la coincidencia de servicios. Comportamiento de la propiedad: - *obligatorio*
value	string	Valor de la etiqueta. Si no se especifica ningún valor o una cadena vacía, sólo se utilizará el nombre de la etiqueta para la coincidencia de servicios.

Módulo

El objeto módulo tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
moduleid	ID	ID del módulo. Comportamiento de la propiedad: - *requerido*
status	integer	Si el acceso al módulo está habilitado. Valores posibles: 0 - Deshabilitado; 1 - *(por defecto)* Habilitado.

Acción

El objeto de acción tiene las siguientes propiedades:

Property	Type	Description
name	string	Nombre de la acción. Valores posibles si `type` del objeto Role está establecido en ”User”, ”Admin” o ”Super admin”: `edit_dashboards` - Crear y editar dashboards; `edit_maps` - Crear y editar mapas; `add_problem_comments` - Añadir comentarios a los problemas; `change_severity` - Cambiar la gravedad del problema; `acknowledge_problems` - Confirmar problemas; `suppress_problems` - Suprimir problemas; `close_problems` - Cerrar problemas; `execute_scripts` - Ejecutar scripts; `manage_api_tokens` - Gestionar tokens de API; `change_problem_ranking` - Cambiar la clasificación del problema de causa a síntoma y viceversa; `edit_own_media` - Permitir a los usuarios crear y editar sus propios medios. Valores posibles si `type` del objeto Role está establecido en ”Admin” o ”Super admin”: `edit_maintenance` - Crear y editar mantenimientos; `manage_scheduled_reports` - Gestionar informes programados, `manage_sla` - Gestionar SLA, `select_server_for_monitoring` - Permitir a los usuarios crear hosts supervisados por el server y por las reglas de descubrimiento de red ejecutadas por el server. Valores posibles si `type` del objeto Role está establecido en ”User” o ”Admin”: `invoke_execute_now` - Permitir a los usuarios que solo tienen permisos de lectura en un host ejecutar comprobaciones de item. Valores posibles si `type` del objeto Role está establecido en ”Super admin”: `edit_user_media` - Permitir a los usuarios crear y editar medios para otros usuarios. Comportamiento de la propiedad: - *required*
status	integer	Indica si está habilitado el acceso para realizar la acción. Valores posibles: 0 - Deshabilitado; 1 - *(predeterminado)* Habilitado.

Acción de dispositivo

El objeto de acción de dispositivo tiene las siguientes propiedades:

Property	Type	Description
name	string	Nombre de la acción de dispositivo. Valores posibles si `type` del objeto Role está establecido en ”User”, ”Admin” o ”Super admin”: `manage_own` - Permitir a los usuarios agregar o eliminar sus propios dispositivos móviles. Valores posibles si `type` del objeto Role está establecido en ”Super admin”: `manage_user` - Permitir a los usuarios agregar o eliminar dispositivos móviles para todos los usuarios. Comportamiento de la propiedad: - *required*
status	integer	Indica si está habilitado el acceso para realizar la acción. Valores posibles: 0 - Deshabilitado; 1 - (*default*) Habilitado.

role.create

Descripción

`object role.create(object/array roles)`

Este método le permite crear nuevos roles.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración del rol de usuario. Consulte User roles para obtener más información.

Parámetros

`(object/array)` Roles a crear.

Además de las propiedades estándar de los roles, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
rules	array	Reglas de rol que se crearán para el rol.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los roles creados bajo la propiedad `roleids`. El orden de los IDs devueltos coincide con el orden de los roles pasados.

Ejemplos

Creando un rol

Cree un rol con tipo ”Usuario” y acceso denegado a dos elementos de la interfaz de usuario.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "role.create",
    "params": {
        "name": "Operator",
        "type": "1",
        "rules": {
            "ui": [
                {
```

```
                    "name": "monitoring.hosts",
                    "status": "0"
                },
                {
                    "name": "monitoring.maps",
                    "status": "0"
                }
            ]
        }
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "roleids": [
            "5"
        ]
    },
    "id": 1
}
```

Ver también

- Reglas de rol
- Elemento de la interfaz
- Módulo
- Acción

Fuente

CRole::create() en *ui/include/classes/api/services/CRole.php*.

role.delete

Descripción

`object role.delete(array roleids)`

Este método permite eliminar roles.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración del rol de usuario. Consulte User roles para obtener más información.

Parámetros

`(array)` IDs de los roles a eliminar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los roles eliminados bajo la propiedad `roleids`.

Ejemplos

Eliminando varios roles de usuario

Eliminar dos roles de usuario.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "role.delete",
    "params": [
        "4",
```

```
        "5"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "roleids": [
            "4",
            "5"
        ]
    },
    "id": 1
}
```

Fuente

CRole::delete() en *ui/include/classes/api/services/CRole.php*.

role.get

Descripción

`integer/array role.get(object parameters)`

El método permite recuperar roles según los parámetros proporcionados.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object)` Parámetros que definen el resultado deseado.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
roleids	ID/array	Devuelve solo los roles con los ID indicados.
current	boolean	Si se establece en `true`, devuelve solo el rol del usuario actual.
selectRules	query	Devuelve una propiedad `rules` con las reglas del rol.
selectUsers	query	Devuelve una propiedad `users` con los usuarios a los que se ha asignado el rol. Consulte `user.get` para conocer las restricciones según el tipo de usuario.
sortfield	string/array	Ordena el resultado según las propiedades indicadas. Valores posibles: `roleid`, `name`.
countOutput	boolean	Estos parámetros se describen en los comentarios de referencia.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	

Parámetro	Tipo	Descripción
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperación de datos del rol

Recupere los datos del rol "Super admin role" y sus reglas de acceso.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "role.get",
    "params": {
        "output": "extend",
        "selectRules": "extend",
        "roleids": "3"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "roleid": "3",
            "name": "Super admin role",
            "type": "3",
            "readonly": "1",
            "rules": {
                "ui": [
                    {
                        "name": "monitoring.dashboard",
                        "status": "1"
                    },
                    {
                        "name": "monitoring.problems",
                        "status": "1"
                    },
                    {
                        "name": "monitoring.hosts",
                        "status": "1"
                    },
                    {
                        "name": "monitoring.latest_data",
                        "status": "1"
                    },
                    {
                        "name": "monitoring.maps",
                        "status": "1"
                    },
                    {
                        "name": "services.services",
                        "status": "1"
```

```
        },
        {
            "name": "services.sla_report",
            "status": "1"
        },
        {
            "name": "inventory.overview",
            "status": "1"
        },
        {
            "name": "inventory.hosts",
            "status": "1"
        },
        {
            "name": "reports.availability_report",
            "status": "1"
        },
        {
            "name": "reports.top_triggers",
            "status": "1"
        },
        {
            "name": "monitoring.discovery",
            "status": "1"
        },
        {
            "name": "services.sla",
            "status": "1"
        },
        {
            "name": "reports.scheduled_reports",
            "status": "1"
        },
        {
            "name": "reports.notifications",
            "status": "1"
        },
        {
            "name": "configuration.template_groups",
            "status": "1"
        },
        {
            "name": "configuration.host_groups",
            "status": "1"
        },
        {
            "name": "configuration.templates",
            "status": "1"
        },
        {
            "name": "configuration.hosts",
            "status": "1"
        },
        {
            "name": "configuration.maintenance",
            "status": "1"
        },
        {
            "name": "configuration.discovery",
            "status": "1"
        },
        {
```

```
            "name": "configuration.trigger_actions",
            "status": "1"
        },
        {
            "name": "configuration.service_actions",
            "status": "1"
        },
        {
            "name": "configuration.discovery_actions",
            "status": "1"
        },
        {
            "name": "configuration.autoregistration_actions",
            "status": "1"
        },
        {
            "name": "configuration.internal_actions",
            "status": "1"
        },
        {
            "name": "reports.system_info",
            "status": "1"
        },
        {
            "name": "reports.audit",
            "status": "1"
        },
        {
            "name": "reports.action_log",
            "status": "1"
        },
        {
            "name": "configuration.event_correlation",
            "status": "1"
        },
        {
            "name": "administration.media_types",
            "status": "1"
        },
        {
            "name": "administration.scripts",
            "status": "1"
        },
        {
            "name": "administration.user_groups",
            "status": "1"
        },
        {
            "name": "administration.user_roles",
            "status": "1"
        },
        {
            "name": "administration.users",
            "status": "1"
        },
        {
            "name": "administration.api_tokens",
            "status": "1"
        },
        {
            "name": "administration.authentication",
            "status": "1"
```

```
            },
            {
                "name": "administration.general",
                "status": "1"
            },
            {
                "name": "administration.audit_log",
                "status": "1"
            },
            {
                "name": "administration.housekeeping",
                "status": "1"
            },
            {
                "name": "administration.proxy_groups",
                "status": "1"
            },
            {
                "name": "administration.proxies",
                "status": "1"
            },
            {
                "name": "administration.macros",
                "status": "1"
            },
            {
                "name": "administration.queue",
                "status": "1"
            }
        ],
        "ui.default_access": "1",
        "profile.redirect.enforce": "0",
        "profile.redirect.url": "",
        "services.read.mode": "1",
        "services.read.list": [],
        "services.read.tag": {
            "tag": "",
            "value": ""
        },
        "services.write.mode": "1",
        "services.write.list": [],
        "services.write.tag": {
            "tag": "",
            "value": ""
        },
        "modules": [
            {
                "moduleid": 1,
                "status": "1"
            },
            {
                "moduleid": 2,
                "status": "1"
            },
            {
                "moduleid": 3,
                "status": "1"
            },
            {
                "moduleid": 4,
                "status": "1"
            },
```

```
                {
                    "moduleid": 5,
                    "status": "1"
                },
                {
                    "moduleid": 6,
                    "status": "1"
                },
                {
                    "moduleid": 7,
                    "status": "1"
                },
                {
                    "moduleid": 8,
                    "status": "1"
                },
                {
                    "moduleid": 9,
                    "status": "1"
                },
                {
                    "moduleid": 10,
                    "status": "1"
                },
                {
                    "moduleid": 11,
                    "status": "1"
                },
                {
                    "moduleid": 12,
                    "status": "1"
                },
                {
                    "moduleid": 13,
                    "status": "1"
                },
                {
                    "moduleid": 14,
                    "status": "1"
                },
                {
                    "moduleid": 15,
                    "status": "1"
                },
                {
                    "moduleid": 16,
                    "status": "1"
                },
                {
                    "moduleid": 17,
                    "status": "1"
                },
                {
                    "moduleid": 18,
                    "status": "1"
                },
                {
                    "moduleid": 19,
                    "status": "1"
                },
                {
                    "moduleid": 20,
```

```
                "status": "1"
            },
            {
                "moduleid": 21,
                "status": "1"
            },
            {
                "moduleid": 22,
                "status": "1"
            },
            {
                "moduleid": 23,
                "status": "1"
            },
            {
                "moduleid": 24,
                "status": "1"
            },
            {
                "moduleid": 25,
                "status": "1"
            },
            {
                "moduleid": 26,
                "status": "1"
            },
            {
                "moduleid": 27,
                "status": "1"
            },
            {
                "moduleid": 28,
                "status": "1"
            },
            {
                "moduleid": 29,
                "status": "1"
            },
            {
                "moduleid": 30,
                "status": "1"
            },
            {
                "moduleid": 31,
                "status": "1"
            },
            {
                "moduleid": 32,
                "status": "1"
            },
            {
                "moduleid": 33,
                "status": "1"
            }
        ],
        "modules.default_access": "1",
        "api.access": "1",
        "api.mode": "0",
        "api": [],
        "actions": [
            {
                "name": "edit_dashboards",
```

```
            "status": "1"
        },
        {
            "name": "edit_maps",
            "status": "1"
        },
        {
            "name": "acknowledge_problems",
            "status": "1"
        },
        {
            "name": "suppress_problems",
            "status": "1"
        },
        {
            "name": "close_problems",
            "status": "1"
        },
        {
            "name": "change_severity",
            "status": "1"
        },
        {
            "name": "add_problem_comments",
            "status": "1"
        },
        {
            "name": "execute_scripts",
            "status": "1"
        },
        {
            "name": "manage_api_tokens",
            "status": "1"
        },
        {
            "name": "edit_maintenance",
            "status": "1"
        },
        {
            "name": "manage_scheduled_reports",
            "status": "1"
        },
        {
            "name": "manage_sla",
            "status": "1"
        },
        {
            "name": "invoke_execute_now",
            "status": "1"
        },
        {
            "name": "change_problem_ranking",
            "status": "1"
        },
        {
            "name": "edit_own_media",
            "status": "1"
        },
        {
            "name": "edit_user_media",
            "status": "1"
        }
```

```
                ],
                "actions.default_access": "1"
            }
        }
    ],
    "id": 1
}
```

Ver también

- Reglas de rol
- Usuario

Fuente

CRole::get() en *ui/include/classes/api/services/CRole.php*.

role.update

Descripción

`object role.update(object/array roles)`

Este método le permite actualizar roles existentes.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración del rol de usuario. Consulte User roles para obtener más información.

Parámetros

`(object/array)` Propiedades del rol a actualizar.

La propiedad `roleid` debe estar definida para cada rol, todas las demás propiedades son opcionales. Sólo se actualizarán las propiedades pasadas, todas las demás permanecerán sin cambios.

Además de las propiedades estándar del rol, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
rules	array	Reglas de acceso a actualizar para el rol.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los roles actualizados bajo la propiedad `roleids`.

Ejemplos

Deshabilitar la capacidad de ejecutar scripts

Actualizar el rol con ID "5", deshabilitar la capacidad de ejecutar scripts.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "role.update",
    "params": [
        {
            "roleid": "5",
            "rules": {
                "actions": [
                    {
                        "name": "execute_scripts",
                        "status": "0"
                    }
                ]
            }
```

```
        }
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "roleids": [
            "5"
        ]
    },
    "id": 1
}
```

Limitando el acceso a la API

Actualizar el rol con ID ”5”, denegando la llamada a cualquier método ”create”, ”update” o ”delete”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "role.update",
    "params": [
        {
            "roleid": "5",
            "rules": {
                "api.access": "1",
                "api.mode": "0",
                "api": ["*.create", "*.update", "*.delete"]
            }
        }
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "roleids": [
            "5"
        ]
    },
    "id": 1
}
```

Fuente

CRole::update() en *ui/include/classes/api/services/CRole.php*.

Script

Esta clase está diseñada para trabajar con scripts.

Referencias de objetos:

- Script
  - Parámetros de webhook
- Debug
  - Entrada de registro

Métodos disponibles:

- script.create - crear nuevos scripts
- script.delete - eliminar scripts
- script.execute - ejecutar scripts
- script.get - recuperar scripts
- script.getscriptsbyevents - recuperar scripts para eventos
- script.getscriptsbyhosts - recuperar scripts para hosts
- script.update - actualizar scripts

Objeto script

Los siguientes objetos están directamente relacionados con la API de `script`.

Script

El objeto script tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
scriptid	ID	ID del script. Comportamiento de la propiedad: - *solo lectura* - *obligatoria* para operaciones de actualización
name	string	Nombre del script. Comportamiento de la propiedad: - *obligatoria* para operaciones de creación
type	integer	Tipo de script. Valores posibles si `scope` está establecido en ”action operation”: 0 - Script; 1 - IPMI; 2 - SSH; 3 - TELNET; 5 - Webhook. Valores posibles si `scope` está establecido en ”manual host action” o ”manual event action”: 6 - URL. Comportamiento de la propiedad: - *obligatoria* para operaciones de creación
command	string	Comando que se ejecutará. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”Script”, ”IPMI”, ”SSH”, ”TELNET” o ”Webhook”
scope	integer	Ámbito del script. Valores posibles: 1 - action operation; 2 - manual host action; 4 - manual event action. Comportamiento de la propiedad: - *obligatoria* para operaciones de creación

Propiedad	Tipo	Descripción
execute_on	integer	Ubicación donde se ejecutará el script. Valores posibles: 0 - ejecutar en Zabbix agent; 1 - ejecutar en Zabbix server. Solo está *admitido* si la ejecución de scripts globales está habilitada en Zabbix server; 2 - *(predeterminado)* ejecutar en Zabbix server o proxy.
menu_path	string	Comportamiento de la propiedad: - *admitida* si `type` está establecido en ”Script” Carpetas separadas por barras que forman una navegación de menú en frontend al hacer clic en un host o evento.
authtype	integer	Comportamiento de la propiedad: - *admitida* si `scope` está establecido en ”manual host action” o ”manual event action” Método de autenticación utilizado para el tipo de script SSH. Valores posibles: 0 - contraseña; 1 - clave pública.
username	string	Comportamiento de la propiedad: - *admitida* si `type` está establecido en ”SSH” Nombre de usuario utilizado para la autenticación.
password	string	Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”SSH” o ”TELNET” Contraseña utilizada para scripts SSH con autenticación mediante contraseña y para scripts TELNET.
publickey	string	Comportamiento de la propiedad: - *admitida* si `type` está establecido en ”SSH” y `authtype` está establecido en ”password”, o si `type` está establecido en ”TELNET” Nombre del archivo de clave pública utilizado para scripts SSH con autenticación mediante clave pública.
privatekey	string	Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”SSH” y `authtype` está establecido en ”public key” Nombre del archivo de clave privada utilizado para scripts SSH con autenticación mediante clave pública.
port	string	Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”SSH” y `authtype` está establecido en ”public key” Número de puerto utilizado para scripts SSH y TELNET.
groupid	ID	Comportamiento de la propiedad: - *admitida* si `type` está establecido en ”SSH” o ”TELNET” ID del grupo de hosts en el que se puede ejecutar el script. Si se establece en ”0”, el script estará disponible en todos los grupos de hosts. Predeterminado: 0.

Propiedad	Tipo	Descripción
usrgrpid	ID	ID del grupo de usuarios al que se permitirá ejecutar el script. Si se establece en ”0”, el script estará disponible para todos los grupos de usuarios. Predeterminado: 0. Comportamiento de la propiedad: - *admitida* si `scope` está establecido en ”manual host action” o ”manual event action”
host_access	integer	Permisos de host necesarios para ejecutar el script. Valores posibles: 2 - *(predeterminado)* lectura; 3 - escritura. Comportamiento de la propiedad: - *admitida* si `scope` está establecido en ”manual host action” o ”manual event action”
confirmation	string	Texto de la ventana emergente de confirmación. La ventana emergente aparecerá al intentar ejecutar el script desde el frontend de Zabbix. Comportamiento de la propiedad: - *admitida* si `scope` está establecido en ”manual host action” o ”manual event action”
timeout	string	Tiempo de espera de ejecución del script webhook en segundos. Se admiten sufijos de tiempo (por ejemplo, `30s`, `1m`). Valores posibles: 1-60s. Predeterminado: 30s. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”Webhook”
parameters	array	Matriz de parámetros de entrada del webhook. Comportamiento de la propiedad: - *admitida* si `type` está establecido en ”Webhook”
description	string	Descripción del script.
url	string	URL definida por el usuario. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”URL”
new_window	integer	Abrir la URL en una ventana nueva. Valores posibles: 0 - No; 1 - *(predeterminado)* Sí. Comportamiento de la propiedad: - *admitida* si `type` está establecido en ”URL”
manualinput	integer	Indica si el script acepta entradas proporcionadas por el usuario. Valores posibles: 0 - *(predeterminado)* Deshabilitada; 1 - Habilitada; Comportamiento de la propiedad: - *admitida* si `scope` está establecido en ”manual host action” o ”manual event action”

Propiedad	Tipo	Descripción
manualinput_prompt	string	Texto de la solicitud de entrada manual. Comportamiento de la propiedad: - *obligatoria* si `manualinput` está establecido en ”Enabled”
manualinput_validator	string	Campo de cadena de caracteres utilizado para validar la entrada proporcionada por el usuario. La cadena consta de una expresión regular o de un conjunto de valores separados por comas. Comportamiento de la propiedad: - *obligatoria* si `manualinput` está establecido en ”Enabled”
manualinput_validator_type	integer	Determina el tipo de entrada de usuario esperado. Valores posibles: 0 - *(predeterminado)* String. Indica que manualinput_validator debe tratarse como una expresión regular; 1 - List. Indica que manualinput_validator debe tratarse como una lista de posibles valores de entrada separados por comas.
manualinput_default_value	string	Valor predeterminado para completar automáticamente la entrada del usuario. Comportamiento de la propiedad: - *admitida* si `manualinput` está establecido en ”Enabled”
		Comportamiento de la propiedad: - *admitida* si `manualinput_validator_type` está establecido en ”String”

Parámetros de webhook

Los parámetros pasados al script del webhook cuando se llama tienen las siguientes propiedades.

Propiedad	Tipo	Descripción
name	string	Nombre del parámetro. Comportamiento de la propiedad: - *obligatorio*
value	string	Valor del parámetro. Soporta macros.

Depuración

Información de depuración del script webhook ejecutado. El objeto de depuración tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
logs	array	Array de entradas de registro.
ms	string	Duración de la ejecución del script en milisegundos.

Entrada de registro

El objeto de entrada de registro tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
level	integer	Nivel de registro.
ms	string	El tiempo transcurrido en milisegundos desde que se ejecutó el script antes de que se añadiera la entrada de registro.
message	string	Mensaje de registro.

script.create

Descripción

`object script.create(object/array scripts)`

Este método le permite crear nuevos scripts.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(objeto/array)` Scripts a crear.

El método acepta scripts con las propiedades estándar de script.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los scripts creados bajo la propiedad `scriptids`. El orden de los IDs devueltos coincide con el orden de los scripts enviados.

Ejemplos

Crear un script de webhook

Cree un script de webhook que envíe una solicitud HTTP a un servicio externo.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "script.create",
    "params": {
        "name": "Webhook script",
        "command": "try {\n var request = new HttpRequest(),\n response,\n data;\n\n request.addHeader('Co
        "scope": 1,
        "type": 5,
        "timeout": "40s",
        "parameters": [
            {
                "name": "token",
                "value": "{$WEBHOOK.TOKEN}"
            },
            {
                "name": "host",
                "value": "{HOST.HOST}"
            },
            {
                "name": "v",
                "value": "2.2"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "scriptids": [
            "3"
        ]
    },
```

```
    "id": 1
}
```

Crear un script SSH

Cree un script SSH con autenticación por clave pública que pueda ejecutarse en un equipo y tenga un menú contextual.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "script.create",
    "params": {
        "name": "SSH script",
        "command": "my script command",
        "type": 2,
        "authtype": 1,
        "username": "John",
        "publickey": "pub.key",
        "privatekey": "priv.key",
        "password": "secret",
        "port": "12345",
        "scope": 2,
        "menu_path": "All scripts/SSH",
        "usrgrpid": "7",
        "groupid": "4"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "scriptids": [
            "5"
        ]
    },
    "id": 1
}
```

Crear un script personalizado

Cree un script personalizado que reiniciará un servidor. El script requerirá acceso de escritura al equipo y solicitará al usuario una entrada manual. Tras el envío exitoso de la entrada, el script mostrará un mensaje de confirmación en la interfaz.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "script.create",
    "params": {
        "name": "Reboot server",
        "command": "reboot server {MANUALINPUT}",
        "type": 0,
        "scope": 2,
        "confirmation": "Are you sure you would like to reboot the server {MANUALINPUT}?",
        "manualinput": 1,
        "manualinput_prompt": "Which server you want to reboot?",
        "manualinput_validator": "[1-9]",
        "manualinput_validator_type": 0,
        "manualinput_default_value": "1"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "scriptids": [
            "4"
        ]
    },
    "id": 1
}
```

Crear un script de tipo URL

Cree un script de tipo URL para el ámbito de equipo que permanezca en la misma ventana y tenga un texto de confirmación.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "script.create",
    "params": {
        "name": "URL script",
        "type": 6,
        "scope": 2,
        "url": "http://zabbix/ui/zabbix.php?action=host.edit&hostid={HOST.ID}",
        "confirmation": "¿Editar equipo {HOST.NAME}?",
        "new_window": 0
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "scriptids": [
            "56"
        ]
    },
    "id": 1
}
```

Crear un script de tipo URL con entrada manual

Cree un script de tipo URL para el ámbito de evento que se abra en una nueva ventana y tenga entrada manual.

Petición:

```
{
    "jsonrpc": "2.0",
    "method": "script.create",
    "params": {
        "name": "Script URL con entrada manual",
        "type": 6,
        "scope": 4,
        "url": "http://zabbix/ui/zabbix.php?action={MANUALINPUT}",
        "new_window": 1,
        "manualinput": 1,
        "manualinput_prompt": "Seleccione una página para abrir:",
        "manualinput_validator": "dashboard.view,script.list,actionlog.list",
        "manualinput_validator_type": 1
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "scriptids": [
            "57"
        ]
    },
    "id": 1
}
```

Fuente

CScript::create() en *ui/include/classes/api/services/CScript.php*.

script.delete

Descripción

`object script.delete(array scriptIds)`

Este método permite eliminar scripts.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración del rol de usuario. Consulte User roles para obtener más información.

Parámetros

`(array)` IDs de los scripts a eliminar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los scripts eliminados bajo la propiedad `scriptids`.

Ejemplos

Eliminar múltiples scripts

Eliminar dos scripts.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "script.delete",
    "params": [
        "3",
        "4"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "scriptids": [
            "3",
            "4"
        ]
    },
    "id": 1
}
```

Fuente

CScript::delete() en *ui/include/classes/api/services/CScript.php*.

script.execute

Descripción

`object script.execute(object parameters)`

Este método le permite ejecutar un script en un host o evento. Excepto los scripts de tipo URL; esos no se pueden ejecutar.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para invocar el método se pueden revocar en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object)` Parámetros que contienen el ID del script a ejecutar, ya sea el ID del host o el ID del evento y el valor de manualinput.

Parámetro	Tipo	Descripción
scriptid	ID	ID del script a ejecutar. Comportamiento del parámetro: - *obligatorio*
hostid	ID	ID del host en el que ejecutar el script. Comportamiento del parámetro: - *obligatorio* si `eventid` no está establecido
eventid	ID	ID del evento en el que ejecutar el script. Comportamiento del parámetro: - *obligatorio* si `hostid` no está establecido
manualinput	string	Valor proporcionado por el usuario para ejecutar el script, sustituyendo la macro {MANUALINPUT}.

Valores de retorno

`(object)` Devuelve el resultado de la ejecución del script.

Propiedad	Tipo	Descripción
response	string	Indica si el script se ejecutó correctamente. Valor posible - `success`.
value	string	Salida del script.
debug	object	Contiene un objeto debug si se ejecuta un script de webhook. Para otros tipos de script, contiene un objeto vacío.

Ejemplos

Ejecutar un script de webhook

Ejecuta un script de webhook que envía una solicitud HTTP a un servicio externo.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "script.execute",
    "params": {
        "scriptid": "4",
        "hostid": "30079"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "response": "success",
        "value": "{\"status\":\"sent\",\"timestamp\":\"1611235391\"}",
        "debug": {
            "logs": [
                {
                    "level": 3,
                    "ms": 480,
                    "message": "[Webhook Script] HTTP status: 200."
                }
            ],
            "ms": 495
        }
    },
    "id": 1
}
```


Ejecutar un script personalizado

Ejecutar un script "ping" en un equipo.

Solicitud:


```
{
    "jsonrpc": "2.0",
    "method": "script.execute",
    "params": {
        "scriptid": "1",
        "hostid": "30079"
    },
    "id": 1
}
```


Respuesta:


```
{
    "jsonrpc": "2.0",
    "result": {
        "response": "success",
        "value": "PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.\n64 bytes from 127.0.0.1: icmp_req=1 tt
        "debug": []
    },
    "id": 1
}
```


Ejecutar un script personalizado con entrada manual

Ejecutar un script "ping" con el comando "ping -c {MANUALINPUT} {HOST.CONN}; case $? in [01]) true;; *) false;; esac" en un host.

Solicitud:


```
{
    "jsonrpc": "2.0",
    "method": "script.execute",
    "params": {
        "scriptid": "7",
        "hostid": "30079",
        "manualinput": "2"
    },
    "id": 1
}
```


Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "response": "success",
        "value": "PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.\n64 bytes from 127.0.0.1: icmp_seq=1 tt
        "debug": []
    },
    "id": 1
}
```


Fuente

CScript::execute() en *ui/include/classes/api/services/CScript.php*.

script.get

Descripción

`integer/array script.get(object parameters)`

El método permite recuperar scripts según los parámetros proporcionados.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
groupids	ID/array	Devolver solo los scripts que se pueden ejecutar en los grupos de hosts indicados.
hostids	ID/array	Devolver solo los scripts que se pueden ejecutar en los hosts indicados.
scriptids	ID/array	Devolver solo los scripts con los IDs indicados.
usrgrpids	ID/array	Devolver solo los scripts que pueden ser ejecutados por usuarios en los grupos de usuarios indicados.
selectHostGroups	query	Devolver una propiedad `hostgroups` con los grupos de hosts en los que se puede ejecutar el script.
selectHosts	query	Devolver una propiedad `hosts` con los hosts en los que se puede ejecutar el script.
selectActions	query	Devolver una propiedad `actions` con las acciones asociadas al script.
sortfield	string/array	Ordenar el resultado por las propiedades indicadas. Valores posibles: `scriptid`, `name`.
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperar todos los scripts

Recuperar todos los scripts configurados.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "script.get",
    "params": {
        "output": "extend"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "scriptid": "1",
            "name": "Ping",
            "command": "/bin/ping -c 3 {HOST.CONN} 2>&1",
            "host_access": "2",
            "usrgrpid": "0",
            "groupid": "0",
            "description": "",
            "confirmation": "",
            "type": "0",
            "execute_on": "1",
            "timeout": "30s",
            "scope": "2",
            "port": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "menu_path": "",
            "url": "",
            "new_window": "1",
            "manualinput": "0",
            "manualinput_prompt": "",
            "manualinput_validator": "",
            "manualinput_validator_type": "0",
            "manualinput_default_value": "",
            "parameters": []
        },
        {
            "scriptid": "2",
            "name": "Traceroute",
            "command": "/usr/bin/traceroute {HOST.CONN} 2>&1",
            "host_access": "2",
            "usrgrpid": "0",
            "groupid": "0",
            "description": "",
            "confirmation": "",
            "type": "0",
            "execute_on": "1",
            "timeout": "30s",
```

```
            "scope": "2",
            "port": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "menu_path": "",
            "url": "",
            "new_window": "1",
            "manualinput": "0",
            "manualinput_prompt": "",
            "manualinput_validator": "",
            "manualinput_validator_type": "0",
            "manualinput_default_value": "",
            "parameters": []
        },
        {
            "scriptid": "3",
            "name": "Detect operating system",
            "command": "sudo /usr/bin/nmap -O {HOST.CONN} 2>&1",
            "host_access": "2",
            "usrgrpid": "7",
            "groupid": "0",
            "description": "",
            "confirmation": "",
            "type": "0",
            "execute_on": "1",
            "timeout": "30s",
            "scope": "2",
            "port": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "menu_path": "",
            "url": "",
            "new_window": "1",
            "manualinput": "0",
            "manualinput_prompt": "",
            "manualinput_validator": "",
            "manualinput_validator_type": "0",
            "manualinput_default_value": "",
            "parameters": []
        },
        {
            "scriptid": "4",
            "name": "Webhook",
            "command": "try {\n var request = new HttpRequest(),\n response,\n data;\n\n request.addHeader
            "host_access": "2",
            "usrgrpid": "7",
            "groupid": "0",
            "description": "",
            "confirmation": "",
            "type": "5",
            "execute_on": "1",
            "timeout": "30s",
            "scope": "2",
            "port": "",
            "authtype": "0",
            "username": "",
```

```
            "password": "",
            "publickey": "",
            "privatekey": "",
            "menu_path": "",
            "url": "",
            "new_window": "1",
            "manualinput": "0",
            "manualinput_prompt": "",
            "manualinput_validator": "",
            "manualinput_validator_type": "0",
            "manualinput_default_value": "",
            "parameters": [
                {
                    "name": "token",
                    "value": "{$WEBHOOK.TOKEN}"
                },
                {
                    "name": "host",
                    "value": "{HOST.HOST}"
                },
                {
                    "name": "v",
                    "value": "2.2"
                }
            ]
        },
        {
            "scriptid": "5",
            "name": "URL",
            "command": "",
            "host_access": "2",
            "usrgrpid": "0",
            "groupid": "0",
            "description": "",
            "confirmation": "Go to {HOST.NAME}?",
            "type": "6",
            "execute_on": "1",
            "timeout": "30s",
            "scope": "4",
            "port": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "menu_path": "",
            "url": "http://zabbix/ui/zabbix.php?action=latest.view&hostids[]={HOST.ID}",
            "new_window": "0",
            "manualinput": "0",
            "manualinput_prompt": "",
            "manualinput_validator": "",
            "manualinput_validator_type": "0",
            "manualinput_default_value": "",
            "parameters": []
        },
        {
            "scriptid": "6",
            "name": "URL with user input",
            "command": "",
            "host_access": "2",
            "usrgrpid": "0",
            "groupid": "0",
```

```
            "description": "",
            "confirmation": "Open zabbix page {MANUALINPUT}?",
            "type": "6",
            "execute_on": "1",
            "timeout": "30s",
            "scope": "2",
            "port": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "menu_path": "",
            "url": "http://zabbix/ui/zabbix.php?action={MANUALINPUT}",
            "new_window": "0",
            "manualinput": "1",
            "manualinput_prompt": "Seleccione una página para abrir:",
            "manualinput_validator": "dashboard.view,script.list,actionlog.list",
            "manualinput_validator_type": "1",
            "parameters": []
        }
    ],
    "id": 1
}
```


Ver también

- Host
- Grupo de hosts

Fuente

CScript::get() en *ui/include/classes/api/services/CScript.php*.

script.getscriptsbyevents

Descripción

`object script.getscriptsbyevents(object parameters)`

Este método permite recuperar todos los scripts disponibles en el evento especificado o un script concreto si se proporciona el ID del script. Cuando se proporciona manualinput, sustituye la macro `{MANUALINPUT}` por el valor especificado.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para invocar el método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(objeto/array)` El método acepta un objeto o un array de objetos con los siguientes parámetros.

Parámetro	Tipo	Descripción
eventid	ID	ID del evento para devolver scripts. Debe ser único. Comportamiento del parámetro: - *requerido*
scriptid	ID	ID del script a devolver.
manualinput	string	Valor del macro {MANUALINPUT} proporcionado por el usuario.

Valores de retorno

`(object)` Devuelve un objeto con los IDs de evento como propiedades y arrays de scripts disponibles como valores. Si se proporciona el ID de script, el valor asociado es un array que contiene el script específico.

Note:
El método expandirá automáticamente las macros en el texto de `confirmation`, el texto de `manualinput prompt` y la `url`.
Si se proporciona el parámetro manualinput, la macro {MANUALINPUT} se resolverá al valor especificado.

Ejemplos

Recuperar scripts por IDs de evento

Recuperar todos los scripts disponibles para los eventos "632" y "614".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "script.getscriptsbyevents",
    "params": [
      {
         "eventid":  "632"
      },
      {
         "eventid":  "614"
      }
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "632": [
            {
                "scriptid": "3",
                "name": "Detect operating system",
                "command": "sudo /usr/bin/nmap -O {HOST.CONN} 2>&1",
                "host_access": "2",
                "usrgrpid": "7",
                "groupid": "0",
                "description": "",
                "confirmation": "",
                "type": "0",
                "execute_on": "1",
                "timeout": "30s",
                "scope": "4",
                "port": "",
                "authtype": "0",
                "username": "",
                "password": "",
                "publickey": "",
                "privatekey": "",
                "menu_path": "",
                "url": "",
                "new_window": "1",
                "manualinput": "0",
                "manualinput_prompt": "",
                "manualinput_validator_type": "0",
                "manualinput_validator": "",
                "manualinput_default_value": "",
                "parameters": []
            },
            {
                "scriptid": "1",
                "name": "Ping",
```

```
            "command": "/bin/ping -c 3 {HOST.CONN} 2>&1",
            "host_access": "2",
            "usrgrpid": "0",
            "groupid": "0",
            "description": "",
            "confirmation": "",
            "type": "0",
            "execute_on": "1",
            "timeout": "30s",
            "scope": "4",
            "port": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "menu_path": "",
            "url": "",
            "new_window": "1",
            "manualinput": "0",
            "manualinput_prompt": "",
            "manualinput_validator_type": "0",
            "manualinput_validator": "",
            "manualinput_default_value": "",
            "parameters": []
        },
        {
            "scriptid": "4",
            "name": "Open Zabbix page",
            "command": "",
            "host_access": "2",
            "usrgrpid": "0",
            "groupid": "0",
            "description": "",
            "confirmation": "Are you sure you want to open page *UNKNOWN*?",
            "type": "6",
            "execute_on": "2",
            "timeout": "30s",
            "scope": "4",
            "port": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "menu_path": "",
            "url": "http://localhost/ui/zabbix.php?action=*UNKNOWN*",
            "new_window": "1",
            "manualinput": "1",
            "manualinput_prompt": "Zabbix page to open:",
            "manualinput_validator_type": "1",
            "manualinput_validator": "dashboard.view,discovery.view",
            "manualinput_default_value": "",
            "parameters": []
        },
        {
            "scriptid": "2",
            "name": "Traceroute",
            "command": "/usr/bin/traceroute {HOST.CONN} 2>&1",
            "host_access": "2",
            "usrgrpid": "0",
            "groupid": "0",
```

```
            "description": "",
            "confirmation": "",
            "type": "0",
            "execute_on": "1",
            "timeout": "30s",
            "scope": "4",
            "port": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "menu_path": "",
            "url": "",
            "new_window": "1",
            "manualinput": "0",
            "manualinput_prompt": "",
            "manualinput_validator_type": "0",
            "manualinput_validator": "",
            "manualinput_default_value": "",
            "parameters": []
        }
    ],
    "614": [
        {
            "scriptid": "3",
            "name": "Detect operating system",
            "command": "sudo /usr/bin/nmap -O {HOST.CONN} 2>&1",
            "host_access": "2",
            "usrgrpid": "7",
            "groupid": "0",
            "description": "",
            "confirmation": "",
            "type": "0",
            "execute_on": "1",
            "timeout": "30s",
            "scope": "4",
            "port": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "menu_path": "",
            "url": "",
            "new_window": "1",
            "manualinput": "0",
            "manualinput_prompt": "",
            "manualinput_validator_type": "1",
            "manualinput_validator": "",
            "manualinput_default_value": "",
            "parameters": []
        },
        {
            "scriptid": "1",
            "name": "Ping",
            "command": "/bin/ping -c 3 {HOST.CONN} 2>&1",
            "host_access": "2",
            "usrgrpid": "0",
            "groupid": "0",
            "description": "",
            "confirmation": "",
```

```
            "type": "0",
            "execute_on": "1",
            "timeout": "30s",
            "scope": "4",
            "port": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "menu_path": "",
            "url": "",
            "new_window": "1",
            "manualinput": "0",
            "manualinput_prompt": "",
            "manualinput_validator_type": "0",
            "manualinput_validator": "",
            "manualinput_default_value": "",
            "parameters": []
        },
        {
            "scriptid": "4",
            "name": "Open Zabbix page",
            "command": "",
            "host_access": "2",
            "usrgrpid": "0",
            "groupid": "0",
            "description": "",
            "confirmation": "Are you sure you want to open page *UNKNOWN*?",
            "type": "6",
            "execute_on": "2",
            "timeout": "30s",
            "scope": "4",
            "port": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "menu_path": "",
            "url": "http://localhost/ui/zabbix.php?action=*UNKNOWN*",
            "new_window": "1",
            "manualinput": "1",
            "manualinput_prompt": "Zabbix page to open:",
            "manualinput_validator_type": "1",
            "manualinput_validator": "dashboard.view,discovery.view",
            "manualinput_default_value": "",
            "parameters": []
        },
        {
            "scriptid": "2",
            "name": "Traceroute",
            "command": "/usr/bin/traceroute {HOST.CONN} 2>&1",
            "host_access": "2",
            "usrgrpid": "0",
            "groupid": "0",
            "description": "",
            "confirmation": "",
            "type": "0",
            "execute_on": "1",
            "timeout": "30s",
            "scope": "4",
```

```
                "port": "",
                "authtype": "0",
                "username": "",
                "password": "",
                "publickey": "",
                "privatekey": "",
                "menu_path": "",
                "url": "",
                "new_window": "1",
                "manualinput": "0",
                "manualinput_prompt": "",
                "manualinput_validator_type": "0",
                "manualinput_validator": "",
                "manualinput_default_value": "",
                "parameters": []
            }
        ]
    },
    "id": 1
}
```

Recuperar un script específico con valor manualinput.

Recuperar el script con ID ”4” en el evento ”632” con el valor manualinput ”dashboard.view”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "script.getscriptsbyevents",
    "params": [
      {
        "eventid":  "632",
        "scriptid": "4",
        "manualinput": "dashboard.view"
      }
  ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
      "632": [
        {
          "scriptid": "4",
          "name": "Abrir página de Zabbix",
          "command": "",
          "host_access": "2",
          "usrgrpid": "0",
          "groupid": "0",
          "description": "",
          "confirmation": "¿Está seguro de que desea abrir la página dashboard.view?",
          "type": "6",
          "execute_on": "2",
          "timeout": "30s",
          "scope": "4",
          "port": "",
          "authtype": "0",
          "username": "",
          "password": "",
          "publickey": "",
          "privatekey": "",
```

```
            "menu_path": "",
            "url": "http://localhost/ui/zabbix.php?action=dashboard.view",
            "new_window": "1",
            "manualinput": "1",
            "manualinput_prompt": "Página de Zabbix a abrir:",
            "manualinput_validator_type": "1",
            "manualinput_validator": "dashboard.view,discovery.view",
            "manualinput_default_value": "",
            "parameters": []
          }
        ]
      },
      "id": 1
}
```


Fuente

CScript::getScriptsByEvents() en *ui/include/classes/api/services/CScript.php*.

script.getscriptsbyhosts

Descripción

`object script.getscriptsbyhosts(object parameters)`

Este método permite recuperar todos los scripts disponibles en el host especificado o un script concreto si se proporciona el ID del script. Cuando se proporciona manualinput, sustituye la macro `{MANUALINPUT}` por el valor especificado.

Note:
Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(objeto/array)` El método acepta un objeto o un array de objetos con los siguientes parámetros.

Parámetro	Tipo	Descripción
hostid	ID	ID del host para devolver los scripts. Debe ser único. Comportamiento del parámetro: - *requerido*
scriptid	ID	ID del script a devolver.
manualinput	string	Valor del valor de la macro {MANUALINPUT} proporcionado por el usuario.

Valores de retorno

`(object)` Devuelve un objeto con los IDs de equipo como propiedades y arreglos de scripts disponibles como valores. Si se proporciona el ID de script, el valor asociado es un arreglo que contiene el script específico.

Note:
El método expandirá automáticamente las macros en el texto de `confirmation`, el texto de `manualinput prompt` y la `url`.
Si se proporciona el parámetro manualinput, la macro {MANUALINPUT} se resolverá al valor especificado.

Ejemplos

Recuperar scripts por IDs de equipos

Recupera todos los scripts disponibles en los equipos ”30079” y ”30073”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "script.getscriptsbyhosts",
    "params": [
       {
          "hostid":  "30079"
       },
       {
          "hostid":  "30073"
       }
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "30079": [
            {
                "scriptid": "3",
                "name": "Detect operating system",
                "command": "sudo /usr/bin/nmap -O {HOST.CONN} 2>&1",
                "host_access": "2",
                "usrgrpid": "7",
                "groupid": "0",
                "description": "",
                "confirmation": "",
                "type": "0",
                "execute_on": "1",
                "timeout": "30s",
                "scope": "2",
                "port": "",
                "authtype": "0",
                "username": "",
                "password": "",
                "publickey": "",
                "privatekey": "",
                "menu_path": "",
                "url": "",
                "new_window": "1",
                "manualinput": "0",
                "manualinput_prompt": "",
                "manualinput_validator_type": "0",
                "manualinput_validator": "",
                "manualinput_default_value": "",
                "parameters": []
            },
            {
                "scriptid": "1",
                "name": "Ping",
                "command": "/bin/ping -c 3 {HOST.CONN} 2>&1",
                "host_access": "2",
                "usrgrpid": "0",
                "groupid": "0",
                "description": "",
                "confirmation": "",
                "type": "0",
                "execute_on": "1",
                "timeout": "30s",
                "scope": "2",
                "port": "",
```

```
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "menu_path": "",
            "url": "",
            "new_window": "1",
            "manualinput": "0",
            "manualinput_prompt": "",
            "manualinput_validator_type": "0",
            "manualinput_validator": "",
            "manualinput_default_value": "",
            "parameters": []
        },
        {
            "scriptid": "4",
            "name": "Open Zabbix page",
            "command": "",
            "host_access": "2",
            "usrgrpid": "0",
            "groupid": "0",
            "description": "",
            "confirmation": "Are you sure you want to open page *UNKNOWN*?",
            "type": "6",
            "execute_on": "2",
            "timeout": "30s",
            "scope": "2",
            "port": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "menu_path": "",
            "url": "http://localhost/ui/zabbix.php?action=*UNKNOWN*",
            "new_window": "1",
            "manualinput": "0",
            "manualinput_prompt": "Zabbix page to open:",
            "manualinput_validator_type": "0",
            "manualinput_validator": "dashboard.view,discovery.view",
            "manualinput_default_value": "",
            "parameters": []
        },
        {
            "scriptid": "2",
            "name": "Traceroute",
            "command": "/usr/bin/traceroute {HOST.CONN} 2>&1",
            "host_access": "2",
            "usrgrpid": "0",
            "groupid": "0",
            "description": "",
            "confirmation": "",
            "type": "0",
            "execute_on": "1",
            "timeout": "30s",
            "scope": "2",
            "port": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
```

```
            "privatekey": "",
            "menu_path": "",
            "url": "",
            "new_window": "1",
            "manualinput": "0",
            "manualinput_prompt": "",
            "manualinput_validator_type": "0",
            "manualinput_validator": "",
            "manualinput_default_value": "",
            "parameters": []
        }
    ],
    "30073": [
        {
            "scriptid": "3",
            "name": "Detect operating system",
            "command": "sudo /usr/bin/nmap -O {HOST.CONN} 2>&1",
            "host_access": "2",
            "usrgrpid": "7",
            "groupid": "0",
            "description": "",
            "confirmation": "",
            "type": "0",
            "execute_on": "1",
            "timeout": "30s",
            "scope": "2",
            "port": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "menu_path": "",
            "url": "",
            "new_window": "1",
            "manualinput": "0",
            "manualinput_prompt": "",
            "manualinput_validator_type": "0",
            "manualinput_validator": "",
            "manualinput_default_value": "",
            "parameters": []
        },
        {
            "scriptid": "1",
            "name": "Ping",
            "command": "/bin/ping -c 3 {HOST.CONN} 2>&1",
            "host_access": "2",
            "usrgrpid": "0",
            "groupid": "0",
            "description": "",
            "confirmation": "",
            "type": "0",
            "execute_on": "1",
            "timeout": "30s",
            "scope": "2",
            "port": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "menu_path": "",
```

```
            "url": "",
            "new_window": "1",
            "manualinput": "0",
            "manualinput_prompt": "",
            "manualinput_validator_type": "0",
            "manualinput_validator": "",
            "manualinput_default_value": "",
            "parameters": []
        },
        {
            "scriptid": "4",
            "name": "Open Zabbix page",
            "command": "",
            "host_access": "2",
            "usrgrpid": "0",
            "groupid": "0",
            "description": "",
            "confirmation": "Are you sure you want to open page *UNKNOWN*?",
            "type": "6",
            "execute_on": "2",
            "timeout": "30s",
            "scope": "2",
            "port": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "menu_path": "",
            "url": "http://localhost/ui/zabbix.php?action=*UNKNOWN*",
            "new_window": "1",
            "manualinput": "1",
            "manualinput_prompt": "Zabbix page to open:",
            "manualinput_validator_type": "1",
            "manualinput_validator": "dashboard.view,discovery.view",
            "manualinput_default_value": "",
            "parameters": []
        },
        {
            "scriptid": "2",
            "name": "Traceroute",
            "command": "/usr/bin/traceroute {HOST.CONN} 2>&1",
            "host_access": "2",
            "usrgrpid": "0",
            "groupid": "0",
            "description": "",
            "confirmation": "",
            "type": "0",
            "execute_on": "1",
            "timeout": "30s",
            "scope": "2",
            "port": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "menu_path": "",
            "url": "",
            "new_window": "1",
            "manualinput": "0",
            "manualinput_prompt": "",
```

```
                "manualinput_validator_type": "0",
                "manualinput_validator": "",
                "manualinput_default_value": "",
                "parameters": []
            }
        ]
    },
    "id": 1
}
```

Recuperar un script específico con valor manualinput.

Recuperar el script con ID "4" en el equipo "30079" con el valor manualinput "dashboard.view".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "script.getscriptsbyhosts",
    "params": [
      {
        "hostid":  "30079",
        "scriptid": "4",
        "manualinput": "dashboard.view"
      }
  ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
      "30079": [
        {
          "scriptid": "4",
          "name": "Abrir página de Zabbix",
          "command": "",
          "host_access": "2",
          "usrgrpid": "0",
          "groupid": "0",
          "description": "",
          "confirmation": "¿Está seguro de que desea abrir la página dashboard.view?",
          "type": "6",
          "execute_on": "2",
          "timeout": "30s",
          "scope": "2",
          "port": "",
          "authtype": "0",
          "username": "",
          "password": "",
          "publickey": "",
          "privatekey": "",
          "menu_path": "",
          "url": "http://localhost/ui/zabbix.php?action=dashboard.view",
          "new_window": "1",
          "manualinput": "1",
          "manualinput_prompt": "Página de Zabbix a abrir:",
          "manualinput_validator_type": "1",
          "manualinput_validator": "dashboard.view,discovery.view",
          "manualinput_default_value": "",
          "parameters": []
        }
      ]
```

```
    },
    "id": 1
}
```

Origen

CScript::getScriptsByHosts() en *ui/include/classes/api/services/CScript.php*.

script.update

Descripción

`object script.update(object/array scripts)`

Este método permite actualizar scripts existentes.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(objeto/array)` Propiedades de script que se actualizarán.

La propiedad `scriptid` debe definirse para cada script, todas las demás propiedades son opcionales. Sólo se actualizarán las propiedades pasadas, todas las demás permanecerán sin cambios. Una excepción es el cambio de la propiedad `type` de 5 (Webhook) a otra: la propiedad `parameters` se limpiará.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los scripts actualizados bajo la propiedad `scriptids`.

Ejemplos

Cambiar el comando del script

Cambie el comando del script a "/bin/ping -c 10 {HOST.CONN} 2>&1".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "script.update",
    "params": {
        "scriptid": "1",
        "command": "/bin/ping -c 10 {HOST.CONN} 2>&1"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "scriptids": [
            "1"
        ]
    },
    "id": 1
}
```

Cambiar el comando del script y añadir entrada manual

Cambie el comando del script a "/bin/ping -c {MANUALINPUT} {HOST.CONN} 2>&1".

Petición:

```
{
    "jsonrpc": "2.0",
```

```
    "method": "script.update",
    "params": {
      "scriptid": "1",
      "command": "/bin/ping -c {MANUALINPUT} {HOST.CONN} 2>&1",
      "manualinput": "1",
      "manualinput_prompt": "Especifique el número de paquetes ICMP a enviar con el comando ping",
      "manualinput_validator": "^(?:[1-9]|10)$",
      "manualinput_validator_type": "0",
      "manualinput_default_value": "10"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "scriptids": [
            "1"
        ]
    },
    "id": 1
}
```

Fuente

CScript::update() en *ui/include/classes/api/services/CScript.php*.

Servicio

Esta clase está diseñada para trabajar con servicios de infraestructura IT/negocio.

Referencias de objetos:

- Servicio
- Regla de estado
- Etiqueta de servicio
- Alarma de servicio
- Etiqueta de problema

Métodos disponibles:

- service.create - crear nuevos servicios
- service.delete - eliminar servicios
- service.get - obtener servicios
- service.update - actualizar servicios

Objeto de servicio

Los siguientes objetos están directamente relacionados con la API de `service`.

Servicio

El objeto service tiene las siguientes propiedades.

Property	Type	Description
serviceid	ID	ID del service. Property behavior: - *read-only* - *required* for update operations

Property	Type	Description
algorithm	integer	Regla de cálculo del estado. Solo aplica si existen servicios hijos. Valores posibles: 0 - establecer el estado en OK; 1 - el más crítico si todos los hijos tienen problemas; 2 - el más crítico de los servicios hijos. Property behavior: - *required* for create operations
name	string	Nombre del service. Property behavior: - *required* for create operations
sortorder	integer	Posición del service utilizada para la ordenación. Valores posibles: 0-999. Property behavior: - *required* for create operations
weight	integer	Peso del service. Valores posibles: 0-1000000. Predeterminado: 0.
propagation_rule	integer	Regla de propagación del estado. Valores posibles: 0 - *(predeterminado)* propagar el estado del service tal cual, sin cambios; 1 - aumentar el estado propagado en un valor dado de `propagation_value` (de 1 a 5 severidades); 2 - disminuir el estado propagado en un valor dado de `propagation_value` (de 1 a 5 severidades); 3 - ignorar este service: el estado no se propaga al service padre en absoluto; 4 - establecer un estado fijo del service usando un valor dado de `propagation_value`. Property behavior: - *required* if `propagation_value` is set

Property	Type	Description
propagation_value	integer	Valor de propagación del estado. Valores posibles si `propagation_rule` está establecido en ”0” o ”3”: 0 - No clasificado. Valores posibles si `propagation_rule` está establecido en ”1” o ”2”: 1 - Información; 2 - Advertencia; 3 - Promedio; 4 - Alto; 5 - Desastre. Valores posibles si `propagation_rule` está establecido en ”4”: -1 - OK; 0 - No clasificado; 1 - Información; 2 - Advertencia; 3 - Promedio; 4 - Alto; 5 - Desastre. Property behavior: - *required* if `propagation_rule` is set
status	integer	Indica si el service está en estado OK o en estado de problema. Si el service está en estado de problema, `status` es igual a la severidad del problema más crítico o al estado más alto de un service hijo en estado de problema. Si el service está en estado OK, `status` es igual a: -1. Property behavior: - *read-only*
description	string	Descripción del service.
uuid	string	Identificador único universal, usado para vincular los services importados con los ya existentes. Se genera automáticamente si no se proporciona.
created_at	integer	Marca de tiempo Unix cuando se creó el service.
readonly	integer	Acceso al service. Valores posibles: 0 - Lectura y escritura; 1 - Solo lectura. Property behavior: - *read-only*

Regla de estado

El objeto de regla de estado tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
type	integer	Condición para establecer el estado (Nuevo estado). Valores posibles: 0 - si al menos (N) servicios hijos tienen el estado (Estado) o superior; 1 - si al menos (N%) de los servicios hijos tienen el estado (Estado) o superior; 2 - si menos de (N) servicios hijos tienen el estado (Estado) o inferior; 3 - si menos de (N%) de los servicios hijos tienen el estado (Estado) o inferior; 4 - si el peso de los servicios hijos con el estado (Estado) o superior es al menos (W); 5 - si el peso de los servicios hijos con el estado (Estado) o superior es al menos (N%); 6 - si el peso de los servicios hijos con el estado (Estado) o inferior es menor que (W); 7 - si el peso de los servicios hijos con el estado (Estado) o inferior es menor que (N%). Donde: - N (W) es `limit_value`; - (Estado) es `limit_status`; - (Nuevo estado) es `new_status`. Comportamiento de la propiedad: - *requerido*
limit_value	integer	Valor límite. Valores posibles: - para N y W: 1-100000; - para N%: 1-100. Comportamiento de la propiedad: - *requerido*
limit_status	integer	Estado límite. Valores posibles: -1 - OK; 0 - No clasificado; 1 - Información; 2 - Advertencia; 3 - Media; 4 - Alta; 5 - Desastre. Comportamiento de la propiedad: - *requerido*
new_status	integer	Valor del nuevo estado. Valores posibles: 0 - No clasificado; 1 - Información; 2 - Advertencia; 3 - Media; 4 - Alta; 5 - Desastre. Comportamiento de la propiedad: - *requerido*

Etiqueta de servicio

El objeto de etiqueta de servicio tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
tag	string	Nombre de la etiqueta de servicio. Comportamiento de la propiedad: - *requerido*
value	string	Valor de la etiqueta de servicio.

Alarma de servicio

> **Note:**
> Las alarmas de servicio no se pueden crear, actualizar ni eliminar directamente a través de la API de Zabbix.

Los objetos de alarma de servicio representan un cambio de estado de un servicio. Tiene las siguientes propiedades.

Property	Type	Description
clock	timestamp	Momento en que se produjo el cambio de estado del servicio.
value	integer	Estado del servicio. Consulte la propiedad `status` del servicio para obtener una lista de los valores posibles.

Etiqueta de problema

Las etiquetas de problema permiten vincular servicios con eventos de problema. El objeto de etiqueta de problema tiene las siguientes propiedades.

Property	Type	Description
tag	string	Nombre de la etiqueta de problema. Property behavior: - *required*
operator	integer	Operador de la condición de mapeo operator. Valores posibles: 0 - *(default)* Igual a; 2 - Contiene.
value	string	Valor de la etiqueta de problema.

service.create

Descripción

`object service.create(object/array services)`

Este método permite crear nuevos servicios.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(objeto/array)` servicios a crear.

Además de las propiedades estándar del servicio, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
children	array	Servicios hijos que se vincularán al servicio. Los servicios hijos deben tener definida únicamente la propiedad `serviceid`.
parents	array	Servicios padres que se vincularán al servicio. Los servicios padres deben tener definida únicamente la propiedad `serviceid`.
tags	array	Etiquetas de servicio que se crearán para el servicio.
problem_tags	array	Etiquetas de problema que se crearán para el servicio.
status_rules	array	Reglas de estado que se crearán para el servicio.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los servicios creados bajo la propiedad `serviceids`. El orden de los IDs devueltos coincide con el orden de los servicios enviados.

Ejemplos

Creando un servicio

Cree un servicio que cambiará al estado de problema si al menos uno de los hijos tiene un problema.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "service.create",
    "params": {
        "name": "Server 1",
        "algorithm": 1,
        "sortorder": 1
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "serviceids": [
            "5"
        ]
    },
    "id": 1
}
```

Fuente

CService::create() en *ui/include/classes/api/services/CService.php*.

service.delete

Descripción

`object service.delete(array serviceIds)`

Este método le permite eliminar servicios.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(array)` IDs de los servicios a eliminar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los servicios eliminados bajo la propiedad `serviceids`.

Ejemplos

Eliminando múltiples servicios

Eliminar dos servicios.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "service.delete",
    "params": [
        "4",
        "5"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "serviceids": [
            "4",
            "5"
        ]
    },
    "id": 1
}
```

Fuente

CService::delete() en *ui/include/classes/api/services/CService.php*.

service.get

Descripción

`integer/array service.get(object parameters)`

El método permite recuperar servicios según los parámetros proporcionados.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
serviceids	ID/array	Devolver solo los servicios con los IDs dados.
parentids	ID/array	Devolver solo los servicios que están vinculados a los servicios padre dados.
deep_parentids	flag	Devolver todos los servicios hijos directos e indirectos. Se utiliza junto con `parentids`.
childids	ID/array	Devolver solo los servicios que están vinculados a los servicios hijos dados.

Parámetro	Tipo	Descripción
evaltype	integer	Método de evaluación de la etiqueta. Valores posibles: 0 - *(por defecto)* And/Or; 2 - Or.
tags	object/array	Devolver solo los servicios con las etiquetas dadas. Formato: `[{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]`. Un array vacío devuelve todos los servicios. Valores posibles de operator: 0 - *(por defecto)* Contiene; 1 - Igual; 2 - No contiene; 3 - No es igual; 4 - Existe; 5 - No existe.
problem_tags	object/array	Devolver solo los servicios con las etiquetas de problema dadas. Formato: `[{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]`. Un array vacío devuelve todos los servicios. Valores posibles de operator: 0 - *(por defecto)* Contiene; 1 - Igual; 2 - No contiene; 3 - No es igual; 4 - Existe; 5 - No existe.
without_problem_tags	flag	Devolver solo los servicios sin etiquetas de problema.
slaids	ID/array	Devolver solo los servicios que están vinculados a la(s) SLA(s) específica(s).
selectChildren	query	Devolver una propiedad `children` con los servicios hijos. Admite `count`.
selectParents	query	Devolver una propiedad `parents` con los servicios padre. Admite `count`.
selectTags	query	Devolver una propiedad `tags` con las etiquetas del servicio. Admite `count`.
selectProblemEvents	query	Devolver una propiedad `problem_events` con un array de objetos de eventos de problema. El objeto de evento de problema tiene las siguientes propiedades: `eventid` - `(ID)` ID del evento; `severity` - `(string)` Severidad actual del evento; `name` - `(string)` Nombre resuelto del evento. Admite `count`.
selectProblemTags	query	Devolver una propiedad `problem_tags` con las etiquetas de problema. Admite `count`.
selectStatusRules	query	Devolver una propiedad `status_rules` con las reglas de estado. Admite `count`.

Parámetro	Tipo	Descripción
selectStatusTimeline	object/array	Devolver una propiedad `status_timeline` que contiene los cambios de estado del servicio para los periodos dados. Formato `[{"period_from": "<period_from>", "period_to": "<period_to>"}, ...]` - `period_from` es la fecha de inicio (inclusive; timestamp entero) y `period_to` es la fecha de fin (exclusive; timestamp entero) para el periodo de interés. Devuelve un array de entradas que contienen una propiedad `start_value` y un array `alarms` para los cambios de estado dentro de los periodos especificados.
sortfield	string/array	Ordenar el resultado por las propiedades dadas. Valores posibles: `serviceid`, `name`, `status`, `sortorder`, `created_at`.
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperar todos los servicios

Recupera todos los datos sobre todos los servicios y sus dependencias.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "service.get",
    "params": {
        "output": "extend",
        "selectChildren": "extend",
        "selectParents": ["serviceid", "name"]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "serviceid": "1",
            "name": "Zabbix cluster",
            "status": "-1",
            "algorithm": "2",
            "sortorder": "0",
```

```
            "weight": "0",
            "propagation_rule": "0",
            "propagation_value": "0",
            "description": "",
            "uuid": "8d77bd91b62347e4b79382912eb5df95",
            "created_at": "1761742392",
            "readonly": false,
            "parents": [],
            "children": [
                {
                    "serviceid": "2",
                    "name": "Zabbix server node 1",
                    "status": "-1",
                    "algorithm": "2",
                    "sortorder": "0",
                    "weight": "0",
                    "propagation_rule": "0",
                    "propagation_value": "0",
                    "description": "",
                    "uuid": "195911d26d7f4e218d6217079bcd5929",
                    "created_at": "1761742623",
                    "readonly": false
                },
                {
                    "serviceid": "3",
                    "name": "Zabbix server node 2",
                    "status": "-1",
                    "algorithm": "2",
                    "sortorder": "0",
                    "weight": "0",
                    "propagation_rule": "0",
                    "propagation_value": "0",
                    "description": "",
                    "uuid": "9fc659a30fe244f690dff25fc2a9db5c",
                    "created_at": "1761742654",
                    "readonly": false
                }
            ]
        },
        {
            "serviceid": "2",
            "name": "Zabbix server node 1",
            "status": "-1",
            "algorithm": "2",
            "sortorder": "0",
            "weight": "0",
            "propagation_rule": "0",
            "propagation_value": "0",
            "description": "",
            "uuid": "195911d26d7f4e218d6217079bcd5929",
            "created_at": "1761742623",
            "readonly": false,
            "parents": [
                {
                    "serviceid": "1",
                    "name": "Zabbix cluster"
                }
            ],
            "children": []
        },
        {
            "serviceid": "3",
```

```
            "name": "Zabbix server node 2",
            "status": "-1",
            "algorithm": "2",
            "sortorder": "0",
            "weight": "0",
            "propagation_rule": "0",
            "propagation_value": "0",
            "description": "",
            "uuid": "9fc659a30fe244f690dff25fc2a9db5c",
            "created_at": "1761742654",
            "readonly": false,
            "parents": [
                {
                    "serviceid": "1",
                    "name": "Zabbix cluster"
                }
            ],
            "children": []
        }
    ],
    "id": 1
}
```


Fuente

CService::get() en *ui/include/classes/api/services/CService.php*.

service.update

Descripción

`object service.update(object/array services)`

Este método permite actualizar servicios existentes.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object/array)` propiedades del servicio a actualizar.

La propiedad `serviceid` debe estar definida para cada servicio, todas las demás propiedades son opcionales. Solo se actualizarán las propiedades pasadas, todas las demás permanecerán sin cambios.

Además de las propiedades estándar del servicio, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
children	array	Servicios hijos para reemplazar los servicios hijos actuales. Los servicios hijos solo deben tener definida la propiedad `serviceid`.
parents	array	Servicios padres para reemplazar los servicios padres actuales. Los servicios padres solo deben tener definida la propiedad `serviceid`.
tags	array	Etiquetas de servicio para reemplazar las etiquetas de servicio actuales.
problem_tags	array	Etiquetas de problema para reemplazar las etiquetas de problema actuales.
status_rules	array	Reglas de estado para reemplazar las reglas de estado actuales.

Valores de retorno

(`object`) Devuelve un objeto que contiene los IDs de los servicios actualizados bajo la propiedad `serviceids`.

Ejemplos

Establecer el padre para un servicio

Hacer que el servicio con ID ”3” sea el padre del servicio con ID ”5”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "service.update",
    "params": {
        "serviceid": "5",
        "parents": [
            {
                "serviceid": "3"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "serviceids": [
            "5"
        ]
    },
    "id": 1
}
```

Añadir un tiempo de inactividad programado

Añadir un tiempo de inactividad para el servicio con ID ”4” programado semanalmente desde el lunes 22:00 hasta el martes 10:00.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "service.update",
    "params": {
        "serviceid": "4",
        "times": [
            {
                "type": "1",
                "ts_from": "165600",
                "ts_to": "201600"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "serviceids": [
            "4"
        ]
    },
    "id": 1
}
```

Origen

CService::update() en *ui/include/classes/api/services/CService.php*.

Servicio descubierto

Esta clase está diseñada para trabajar con servicios descubiertos.

Referencias de objetos:

- Servicio descubierto

Métodos disponibles:

- dservice.get - recuperar servicios descubiertos

Objeto de servicio descubierto

Los siguientes objetos están directamente relacionados con la API `dservice`.

Servicio descubierto

> **Note:**
> Los servicios descubiertos son creados por el servidor Zabbix y no pueden ser modificados a través de la API.

El objeto de servicio descubierto contiene información sobre un servicio descubierto por una regla de descubrimiento de red en un host. Tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
dserviceid	ID	ID del servicio descubierto.
dcheckid	ID	ID de la comprobación de descubrimiento utilizada para detectar el servicio.
dhostid	ID	ID del host descubierto que ejecuta el servicio.
dns	string	DNS del host que ejecuta el servicio.
ip	string	Dirección IP del host que ejecuta el servicio.
lastdown	timestamp	Hora en que el servicio descubierto se cayó por última vez.
lastup	timestamp	Hora en que el servicio descubierto se levantó por última vez.
port	integer	Número de puerto del servicio.
status	integer	Estado del servicio. Valores posibles: 0 - servicio levantado; 1 - servicio caído.
value	string	Valor devuelto por el servicio al realizar una comprobación de descubrimiento de Zabbix agent, SNMPv1, SNMPv2 o SNMPv3.

dservice.get

Descripción

`integer/array dservice.get(object parameters)`

El método permite recuperar los servicios descubiertos según los parámetros dados.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
dserviceids	ID/array	Devolver solo los servicios descubiertos con los IDs dados.
dhostids	ID/array	Devolver solo los servicios descubiertos que pertenecen a los hosts descubiertos dados.
dcheckids	ID/array	Devolver solo los servicios descubiertos que han sido detectados por las comprobaciones de descubrimiento dadas.
druleids	ID/array	Devolver solo los servicios descubiertos que han sido detectados por las reglas de descubrimiento dadas.
selectDRules	query	Devuelve una propiedad `drules` con un array de las reglas de descubrimiento que detectaron el servicio.
selectDHosts	query	Devuelve una propiedad `dhosts` con un array de los hosts descubiertos a los que pertenece el servicio.
selectHosts	query	Devuelve una propiedad `hosts` con los hosts con la misma dirección IP y proxy que el servicio. Admite `count`.
limitSelects	integer	Limita el número de registros devueltos por las subselecciones. Se aplica a las siguientes subselecciones: `selectHosts` - el resultado se ordenará por `hostid`.
sortfield	string/array	Ordena el resultado por las propiedades dadas. Valores posibles: `dserviceid`, `dhostid`, `ip`.
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperar servicios descubiertos en un equipo

Recuperar todos los servicios descubiertos detectados en el equipo descubierto ”11”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "dservice.get",
    "params": {
        "output": "extend",
        "dhostids": "11"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
```

```
        {
            "dserviceid": "12",
            "dhostid": "11",
            "value": "",
            "port": "80",
            "status": "1",
            "lastup": "0",
            "lastdown": "1348650607",
            "dcheckid": "5",
            "ip": "192.168.1.134",
            "dns": "john.local"
        },
        {
            "dserviceid": "13",
            "dhostid": "11",
            "value": "",
            "port": "21",
            "status": "1",
            "lastup": "0",
            "lastdown": "1348650610",
            "dcheckid": "6",
            "ip": "192.168.1.134",
            "dns": "john.local"
        }
    ],
    "id": 1
}
```


Ver también

- Host descubierto
- Comprobación de descubrimiento
- Host

Fuente

CDService::get() en *ui/include/classes/api/services/CDService.php*.

SLA

Esta clase está diseñada para trabajar con objetos SLA (Acuerdo de Nivel de Servicio) utilizados para estimar el rendimiento de la infraestructura de TI y los servicios empresariales.

Referencias de objetos:

- SLA
- Horario de SLA
- Tiempo de inactividad excluido de SLA
- Etiqueta de servicio de SLA

Métodos disponibles:

- sla.create - crear nuevos SLA
- sla.delete - eliminar SLA
- sla.get - recuperar SLA
- sla.getsli - recuperar datos del Indicador de Nivel de Servicio (SLI) para SLA
- sla.update - actualizar SLA

Objeto SLA

Los siguientes objetos están directamente relacionados con la API de `sla` (Acuerdo de Nivel de Servicio).

SLA

El objeto SLA tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
slaid	ID	ID del SLA. Comportamiento de la propiedad: - *solo lectura* - *requerido* para operaciones de actualización
name	string	Nombre del SLA. Comportamiento de la propiedad: - *requerido* para operaciones de creación
period	integer	Periodo de reporte del SLA. Valores posibles: 0 - diario; 1 - semanal; 2 - mensual; 3 - trimestral; 4 - anual. Comportamiento de la propiedad: - *requerido* para operaciones de creación
slo	float	Objetivo de nivel de servicio mínimo aceptable expresado como porcentaje. Si el Indicador de Nivel de Servicio (SLI) cae por debajo, el SLA se considera en estado de problema/incumplido. Valores posibles: 0-100 (hasta 4 decimales). Comportamiento de la propiedad: - *requerido* para operaciones de creación
effective_date	integer	Fecha de entrada en vigor del SLA. Valores posibles: marca de tiempo de la fecha en UTC.
timezone	string	Zona horaria de reporte, por ejemplo: `Europe/London`, `UTC`. Para la lista completa de zonas horarias soportadas, consulte la documentación de PHP. Comportamiento de la propiedad: - *requerido* para operaciones de creación
status	integer	Estado del SLA. Valores posibles: 0 - *(por defecto)* SLA deshabilitado; 1 - SLA habilitado.
description	string	Descripción del SLA.

Calendario de SLA

El objeto de calendario de SLA define los períodos en los que el/los servicio(s) conectado(s) están programados para estar en funcionamiento. Tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
period_from	integer	Hora de inicio del período de tiempo semanal recurrente (inclusive). Valores posibles: número de segundos (contando desde el domingo). Comportamiento de la propiedad: - *obligatorio*

Propiedad	Tipo	Descripción
period_to	integer	Hora de finalización del período de tiempo semanal recurrente (exclusivo). Valores posibles: número de segundos (contando desde el domingo). Comportamiento de la propiedad: - *obligatorio*

Tiempo de inactividad excluido del SLA

El objeto de tiempo de inactividad excluido define los períodos en los que el/los servicio(s) conectado(s) están programados para estar fuera de servicio, sin afectar al SLI, por ejemplo, durante un mantenimiento planificado. Tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
name	string	Nombre del tiempo de inactividad excluido. Comportamiento de la propiedad: - *requerido*
period_from	integer	Hora de inicio del tiempo de inactividad excluido (inclusive). Valores posibles: timestamp. Comportamiento de la propiedad: - *requerido*
period_to	integer	Hora de finalización del tiempo de inactividad excluido (exclusivo). Valores posibles: timestamp. Comportamiento de la propiedad: - *requerido*

Etiqueta de servicio SLA

El objeto de etiqueta de servicio SLA vincula los servicios a incluir en los cálculos para el SLA. Tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
tag	string	Nombre de la etiqueta del servicio SLA. Comportamiento de la propiedad: - *obligatorio*
operator	integer	Operador de la etiqueta del servicio SLA. Valores posibles: 0 - *(por defecto)* Igual; 2 - Contiene.
value	string	Valor de la etiqueta del servicio SLA.

sla.create

Descripción

`object sla.create(object/array SLAs)`

Este método le permite crear nuevos objetos SLA.

Note:
Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(objeto/array)` Objetos SLA a crear.

Además de las propiedades estándar de SLA, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
service_tags	array	Etiquetas de servicio SLA que se crearán para el SLA. Comportamiento del parámetro: - *obligatorio*
schedule	array	Horario de SLA que se creará para el SLA. Especificar un parámetro vacío se interpretará como un horario 24x7. Por defecto: horario 24x7.
excluded_downtimes	array	Tiempos de inactividad excluidos de SLA que se crearán para el SLA.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los SLAs creados bajo la propiedad `slaids`. El orden de los IDs devueltos coincide con el orden de los SLAs enviados.

Ejemplos

Creación de un SLA

Indique crear una entrada de SLA para: * rastrear el tiempo de actividad de los servicios relacionados con el motor SQL; * horario personalizado de todos los días de la semana excluyendo la última hora del sábado; * una fecha de vigencia del último día del año 2022; * con un tiempo de inactividad planificado de 1 hora y 15 minutos comenzando a la medianoche del 4 de julio; * el cálculo del informe semanal de SLA estará activado; * el SLO mínimo aceptable será del 99.9995%.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "sla.create",
    "params": [
        {
            "name": "Database Uptime",
            "slo": "99.9995",
            "period": "1",
            "timezone": "America/Toronto",
            "description": "Provide excellent uptime for main database engines.",
            "effective_date": 1672444800,
            "status": 1,
            "schedule": [
                {
                    "period_from": 0,
                    "period_to": 601200
                }
            ],
            "service_tags": [
                {
                    "tag": "database",
                    "operator": "0",
                    "value": "mysql"
                },
                {
                    "tag": "database",
                    "operator": "0",
                    "value": "postgresql"
                }
            ],
            "excluded_downtimes": [
                {
                    "name": "Software version upgrade rollout",
                    "period_from": "1648760400",
```

```
                    "period_to": "1648764900"
                }
            ]
        }
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "slaids": [
            "5"
        ]
    },
    "id": 1
}
```

Fuente

CSla::create() en *ui/include/classes/api/services/CSla.php*.

sla.delete

Descripción

`object sla.delete(array slaids)`

Este método permite eliminar entradas de SLA.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración del rol de usuario. Consulte User roles para obtener más información.

Parámetros

`(array)` IDs de los SLA a eliminar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los SLAs eliminados bajo la propiedad `slaids`.

Ejemplos

Eliminando múltiples SLAs

Eliminar dos entradas de SLA.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "sla.delete",
    "params": [
        "4",
        "5"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "slaids": [
            "4",
```

```
            "5"
        ]
    },
    "id": 1
}
```

Fuente

CSla::delete() en *ui/include/classes/api/services/CSla.php*.

sla.get

Descripción

`integer/array sla.get(object parameters)`

El método permite recuperar objetos SLA según los parámetros proporcionados.

Note:
Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
slaids	ID/array	Devolver solo los SLA con los IDs dados.
serviceids	ID/array	Devolver solo los SLA que coincidan con los servicios específicos.
selectSchedule	query	Devolver una propiedad `schedule` con los horarios de SLA. Admite `count`.
selectExcludedDowntimes	query	Devolver una propiedad `excluded_downtimes` con los periodos de exclusión de SLA. Admite `count`.
selectServiceTags	query	Devolver una propiedad `service_tags` con las etiquetas de servicio de SLA. Admite `count`.
sortfield	string/array	Ordenar el resultado por las propiedades dadas. Valores posibles: `slaid`, `name`, `period`, `slo`, `effective_date`, `timezone`, `status`, `description`.
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos;

- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperar todos los SLA

Recupera todos los datos sobre todos los SLA y sus propiedades.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "sla.get",
    "params": {
        "output": "extend",
        "selectSchedule": ["period_from", "period_to"],
        "selectExcludedDowntimes": ["name", "period_from", "period_to"],
        "selectServiceTags": ["tag", "operator", "value"],
        "preservekeys": true
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "1": {
            "slaid": "1",
            "name": "Database Uptime",
            "period": "1",
            "slo": "99.9995",
            "effective_date": "1672444800",
            "timezone": "America/Toronto",
            "status": "1",
            "description": "Proporcionar un excelente tiempo de actividad para los principales motores de
            "service_tags": [
                {
                    "tag": "database",
                    "operator": "0",
                    "value": "mysql"
                },
                {
                    "tag": "database",
                    "operator": "0",
                    "value": "postgresql"
                }
            ],
            "schedule": [
                {
                    "period_from": "0",
                    "period_to": "601200"
                }
            ],
            "excluded_downtimes": [
                {
                    "name": "Despliegue de actualización de versión de software",
                    "period_from": "1648760400",
                    "period_to": "1648764900"
                }
            ]
        }
    },
    "id": 1
}
```

Fuente

CSla:get() en *ui/include/classes/api/services/CSla.php*.

sla.getsli

Descripción

`object sla.getsli(object parameters)`

Este método permite calcular los datos del Service Level Indicator (SLI) para un Service Level Agreement (SLA).

Note:
Este método está disponible para usuarios de cualquier tipo. Los permisos para invocar el método se pueden revocar en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object)` Parámetros que contienen el ID del SLA, los períodos de informe y, opcionalmente, los IDs de los servicios para calcular el SLI.

Parámetro	Tipo	Descripción
slaid	ID	ID del SLA para devolver la información de disponibilidad. Comportamiento del parámetro: - *requerido*
period_from	timestamp	Marca de tiempo de inicio (inclusive) para informar el SLI. Valores posibles: marca de tiempo Unix.
period_to	timestamp	Marca de tiempo de finalización (inclusive) para informar el SLI. Valores posibles: marca de tiempo Unix.
periods	integer	Número de períodos a informar. Valores posibles: 1-100
serviceids	ID/array	IDs de los servicios para devolver el SLI.

División de períodos

La siguiente tabla muestra la disposición de los períodos devueltos en función de las combinaciones de parámetros.

Note:
Los períodos devueltos no preceden al primer período disponible (basado en la fecha efectiva del SLA) y no exceden el período actual.

Parámetros			Períodos devueltos
period_from	**period_to**	**periods**	
-	-	-	Últimos 20 períodos, incluido el actual.
-	-	N	Últimos N períodos.
-	especificado	-	Últimos 20 períodos antes de `period_to`.
-	especificado	N	Últimos N períodos antes de `period_to`.
especificado	-	-	Primeros 20 períodos comenzando con `period_from`.
especificado	-	N	Primeros N períodos comenzando con `period_from`.
especificado	especificado	-	Hasta 100 períodos en el rango especificado.
especificado	especificado	N	N períodos en el rango especificado.

Valores de retorno

`(object)` Devuelve los resultados del cálculo.

Propiedad	Tipo	Descripción
periods	array	Lista de los periodos informados. Cada periodo informado se representa como un objeto que consta de: - `period_from` - `(timestamp)` Marca de tiempo de inicio (inclusive) del periodo informado. - `period_to` - `(timestamp)` Marca de tiempo de finalización (exclusive) del periodo informado. Los periodos se ordenan por `period_from`, apareciendo primero los más antiguos.
serviceids	array	Lista de IDs de servicios en los periodos informados. El orden de clasificación de la lista no está definido. Incluso si se pasó el parámetro `serviceids` al método `sla.getsli`.
sli	array	Datos SLI (como **matriz bidimensional**) para cada periodo y servicio informado. El índice de la propiedad `periods` se utiliza como la **primera** dimensión de la propiedad `sli`. El índice de la propiedad `serviceids` se utiliza como la **segunda** dimensión de la propiedad `sli`.

Datos SLI

Los datos SLI devueltos para cada período informado y servicio consisten en:

Propiedad	Tipo	Descripción
uptime	integer	Cantidad de tiempo que el servicio estuvo en estado *OK* durante el tiempo de actividad programado, menos los tiempos de inactividad excluidos.
downtime	integer	Cantidad de tiempo que el servicio estuvo en estado *no OK* durante el tiempo de actividad programado, menos los tiempos de inactividad excluidos.
sli	float	SLI (porcentaje del tiempo total de actividad), basado en uptime y downtime.
error_budget	integer	Error budget (en segundos), basado en el SLI y el SLO.
excluded_downtimes	array	Array de tiempos de inactividad excluidos en este período de informe. Cada objeto contendrá los siguientes parámetros: - `name` - `(string)` Nombre del tiempo de inactividad excluido. - `period_from` - `(timestamp)` Marca de tiempo de inicio (inclusive) del tiempo de inactividad excluido. - `period_to` - `(timestamp)` Marca de tiempo de finalización (exclusiva) del tiempo de inactividad excluido. Los tiempos de inactividad excluidos se ordenan por `period_from`, apareciendo primero los períodos más tempranos.

Ejemplos

Calcular SLI para un SLA diario

Recuperar datos de SLI para los servicios con ID ”1” y ”4” que están vinculados al SLA con ID ”1”. Recuperar datos para un solo periodo hasta ”1761861599” (30 de octubre de 2025 23:59:59 GMT+0200). Dado que el periodo de informe del SLA es diario, los datos de SLI se recuperan desde ”1761775200” (30 de octubre de 2025 00:00:00 GMT+0200) hasta ”1761861600” (31 de octubre de 2025 00:00:00 GMT+0200).

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "sla.getsli",
    "params": {
        "slaid": "1",
        "serviceids": [
            1,
            4
        ],
        "periods": 1,
        "period_to": 1761861599,
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "periods": [
            {
                "period_from": 1761775200,
                "period_to": 1761861600
            }
        ],
        "serviceids": [
            1,
            4
        ],
        "sli": [
            [
                {
                    "uptime": 43843,
                    "downtime": 0,
                    "sli": 100,
                    "error_budget": 0,
                    "excluded_downtimes": [
                        {
                            "name": "Maintenance OCT",
                            "period_from": 1761825600,
                            "period_to": 1761829200
                        }
                    ]
                },
                {
                    "uptime": 32225,
                    "downtime": 0,
                    "sli": 100,
                    "error_budget": 0,
                    "excluded_downtimes": []
                }
            ]
        ]
    },
    "id": 1
}
```

Calcular el SLI para un SLA mensual

Recuperar los datos de SLI de los servicios con IDs ”50”, ”60” y ”70” que están vinculados al SLA con ID ”5”. Recuperar datos para tres períodos a partir de ”1635724800” (01 Nov 2021 00:00:00 UTC). Dado que el período de informe del SLA es mensual, los datos de SLI se recuperan para los siguientes tres meses:

- Desde ”1635724800” (01 Nov 2021 00:00:00 UTC) hasta ”1638316800” (01 Dic 2021 00:00:00 UTC)

- Desde ”1638316800” (01 Dic 2021 00:00:00 UTC) hasta ”1640995200” (01 Ene 2022 00:00:00 UTC)
- Desde ”1640995200” (01 Ene 2022 00:00:00 UTC) hasta ”1643673600” (01 Feb 2022 00:00:00 UTC)

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "sla.getsli",
    "params": {
        "slaid": "5",
        "serviceids": [
            50,
            60,
            70
        ],
        "periods": 3,
        "period_from": 1635724800
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "periods": [
            {
                "period_from": 1635724800,
                "period_to": 1638316800
            },
            {
                "period_from": 1638316800,
                "period_to": 1640995200
            },
            {
                "period_from": 1640995200,
                "period_to": 1643673600
            }
        ],
        "serviceids": [
            50,
            60,
            70
        ],
        "sli": [
            [
                {
                    "uptime": 1186212,
                    "downtime": 0,
                    "sli": 100,
                    "error_budget": 0,
                    "excluded_downtimes": [
                        {
                            "name": "Maintenance Nov 25 - Dec 01",
                            "period_from": 1637836212,
                            "period_to": 1638316800
                        }
                    ]
                },
                {
                    "uptime": 1186212,
                    "downtime": 0,
                    "sli": 100,
                    "error_budget": 0,
```

```
                "excluded_downtimes": [
                    {
                        "name": "Maintenance Nov 25 - Dec 01",
                        "period_from": 1637836212,
                        "period_to": 1638316800
                    }
                ]
            },
            {
                "uptime": 1186212,
                "downtime": 0,
                "sli": 100,
                "error_budget": 0,
                "excluded_downtimes": [
                    {
                        "name": "Maintenance Nov 25 - Dec 01",
                        "period_from": 1637836212,
                        "period_to": 1638316800
                    }
                ]
            }
        ],
        [
            {
                "uptime": 1147548,
                "downtime": 0,
                "sli": 100,
                "error_budget": 0,
                "excluded_downtimes": [
                    {
                        "name": "Maintenance Dec 02 - Dec 10",
                        "period_from": 1638439200,
                        "period_to": 1639109652
                    }
                ]
            },
            {
                "uptime": 1147548,
                "downtime": 0,
                "sli": 100,
                "error_budget": 0,
                "excluded_downtimes": [
                    {
                        "name": "Maintenance Dec 02 - Dec 10",
                        "period_from": 1638439200,
                        "period_to": 1639109652
                    }
                ]
            },
            {
                "uptime": 1147548,
                "downtime": 0,
                "sli": 100,
                "error_budget": 0,
                "excluded_downtimes": [
                    {
                        "name": "Maintenance Dec 02 - Dec 10",
                        "period_from": 1638439200,
                        "period_to": 1639109652
                    }
                ]
            }
```

```
            ],
            [
                {
                    "uptime": 1674000,
                    "downtime": 0,
                    "sli": 100,
                    "error_budget": 0,
                    "excluded_downtimes": []
                },
                {
                    "uptime": 1674000,
                    "downtime": 0,
                    "sli": 100,
                    "error_budget": 0,
                    "excluded_downtimes": []
                },
                {
                    "uptime": 1674000,
                    "downtime": 0,
                    "sli": 100,
                    "error_budget": 0,
                    "excluded_downtimes": []
                }
            ]
        ]
    },
    "id": 1
}
```


Fuente

CSla::getSli() en *ui/include/classes/api/services/CSla.php*

sla.update

Descripción

`object sla.update(object/array slaids)`

Este método le permite actualizar entradas de SLA existentes.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object/array)` Propiedades del SLA que se van a actualizar.

La propiedad `slaid` debe estar definida para cada SLA, todas las demás propiedades son opcionales. Sólo se actualizarán las propiedades pasadas, todas las demás permanecerán sin cambios.

Además de las propiedades estándar del SLA, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
service_tags	array	Etiquetas de servicio SLA para reemplazar las etiquetas de servicio SLA actuales.
schedule	array	Horario del SLA para reemplazar el actual. Si se especifica el parámetro como vacío, se interpretará como un horario 24x7.
excluded_downtimes	array	Tiempos de inactividad excluidos del SLA para reemplazar los actuales.

Valores de retorno

(`object`) Devuelve un objeto que contiene los IDs de los SLAs actualizados bajo la propiedad `slaids`.

Ejemplos

Actualización de etiquetas de servicio

Hacer que el SLA con ID "5" se calcule en intervalos mensuales para servicios relacionados con NoSQL, sin cambiar su programación ni los tiempos de inactividad excluidos; establecer el SLO en 95%.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "sla.update",
    "params": [
        {
            "slaid": "5",
            "name": "NoSQL Database engines",
            "slo": "95",
            "period": 2,
            "service_tags": [
                {
                    "tag": "database",
                    "operator": "0",
                    "value": "redis"
                },
                {
                    "tag": "database",
                    "operator": "0",
                    "value": "mongodb"
                }
            ]
        }
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "slaids": [
            "5"
        ]
    },
    "id": 1
}
```

Cambiar el calendario de un SLA

Cambie el SLA con ID "5" a un calendario 24x7.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "sla.update",
    "params": {
        "slaid": "5",
        "schedule": []
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
```

```
    "result": {
        "slaids": [
            "5"
        ]
    },
    "id": 1
}
```

Cambiar los tiempos de inactividad excluidos para un SLA

Agregue un tiempo de inactividad planificado de 4 horas para una actualización de RAM el 6 de abril de 2022, mientras mantiene (debe definirse de nuevo) una actualización de software previamente existente planificada para el 4 de julio para el SLA con ID ”5”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "sla.update",
    "params": {
        "slaid": "5",
        "excluded_downtimes": [
            {
                "name": "Software version upgrade rollout",
                "period_from": "1648760400",
                "period_to": "1648764900"
            },
            {
                "name": "RAM upgrade",
                "period_from": "1649192400",
                "period_to": "1649206800"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "slaids": [
            "5"
        ]
    },
    "id": 1
}
```

Fuente

CSla::update() en *ui/include/classes/api/services/CSla.php*.

Span de APM

Esta clase está diseñada para trabajar con spans recopilados para la monitorización del rendimiento de las aplicaciones (APM).

Referencias de objetos:

- Span de APM

Métodos disponibles:

- apmspan.get: recuperar spans

Objeto span de APM

Los siguientes objetos están directamente relacionados con la API `apmspan`.

Note:
Cada propiedad corresponde a una columna de la tabla `otel_traces` del origen de datos de telemetría. Para consultar la lista de columnas, consulte Columnas de datos de telemetría compatibles.

Span de APM

El objeto de span de APM representa un span (una única operación dentro de un trace, como una solicitud HTTP, una consulta a una base de datos o una llamada a una función).

Los spans forman una jerarquía que comienza con el span raíz, que representa la primera operación de la solicitud. Cada uno de los demás spans es un span secundario del span durante el cual se inició su operación.

El objeto de span de APM tiene las siguientes propiedades:

Property	Type	Description
timestamp	integer	Hora Unix (en nanosegundos) en la que se inició el span. Comportamiento de la propiedad: - *solo lectura*
traceid	string	ID del trace al que pertenece el span. Comportamiento de la propiedad: - *solo lectura*
spanid	string	ID del span. Comportamiento de la propiedad: - *solo lectura*
parent_spanid	string	ID del span principal. Vacío para el span raíz. Comportamiento de la propiedad: - *solo lectura*
trace_state	string	Datos adicionales, si los hay, que las herramientas de tracing adjuntan al trace (consulte W3C Trace Context). Comportamiento de la propiedad: - *solo lectura*
span_name	string	Nombre de la operación que representa el span (por ejemplo, `mysqli_query`). Comportamiento de la propiedad: - *solo lectura*
span_kind	string	Rol del span. Valores posibles: `Unspecified`: el tipo de span no está especificado; `Internal`: operación realizada dentro de una aplicación; `Server`: gestión de una solicitud entrante; `Client`: solicitud saliente; `Producer`: creación de una operación asíncrona (por ejemplo, el envío de un mensaje); `Consumer`: procesamiento de una operación asíncrona. Comportamiento de la propiedad: - *solo lectura*
service_name	string	Nombre del servicio que produjo el span (por ejemplo, `zabbix-frontend`), obtenido del atributo de recurso `service.name` de OpenTelemetry. Comportamiento de la propiedad: - *solo lectura*

Property	Type	Description
resource_attributes	object	Atributos de recurso clave-valor del span (por ejemplo, `host.name: eabf8c2a5ddc`). Comportamiento de la propiedad: - *solo lectura*
scope_name	string	Nombre del ámbito de instrumentación que produjo el span. Comportamiento de la propiedad: - *solo lectura*
scope_version	string	Versión del ámbito de instrumentación. Comportamiento de la propiedad: - *solo lectura*
span_attributes	object	Atributos del span clave-valor del span (por ejemplo, `db.system: mysql`). Comportamiento de la propiedad: - *solo lectura*
duration	integer	Duración del span en nanosegundos. Comportamiento de la propiedad: - *solo lectura*
status_code	string	Estado de la operación que representa el span. Valores posibles: `Unset`: no se ha asignado ningún estado explícito al span; `Ok`: la operación se completó correctamente; `Error`: la operación se completó con un error. Comportamiento de la propiedad: - *solo lectura*
status_message	string	Información adicional, si la hay, sobre el estado del span. Comportamiento de la propiedad: - *solo lectura*
events	array	Matriz de objetos event del span, es decir, puntos relevantes en el tiempo durante la operación (por ejemplo, excepciones). Comportamiento de la propiedad: - *solo lectura*
links	array	Matriz de objetos link del span, que apuntan a spans relacionados en el mismo trace o en otro. Comportamiento de la propiedad: - *solo lectura*

apmspan.get

Descripción

`integer/array apmspan.get(object parameters)`

El método permite recuperar spans según los parámetros indicados.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración del rol de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object)` Parámetros que definen el resultado deseado.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
time_from	timestamp	Devuelve solo los spans que comenzaron en el momento indicado o después. Comportamiento del parámetro: - *obligatorio*
time_till	timestamp	Devuelve solo los spans que comenzaron antes del momento indicado. Comportamiento del parámetro: - *obligatorio*
traceids	string/array	Devuelve solo los spans que pertenecen a las trazas indicadas.
spanids	string/array	Devuelve solo los spans con los ID indicados.
parent_spanids	string/array	Devuelve solo los spans con los spans principales indicados.
resource_attributes	object/array	Devuelve solo los spans con los atributos de recurso indicados. Formato: `[{"key": "<key>", "value": "<value>", "operator": "<operator>"}, ...]`. Una matriz vacía devuelve todos los spans. Valores posibles de operator: 0 - *(predeterminado)* Contiene; 1 - Es igual a; 2 - No contiene; 3 - No es igual a; 4 - Existe; 5 - No existe.
resource_attributes_evaltype	integer	Método de evaluación de las condiciones en `resource_attributes`. Valores posibles: 0 - *(predeterminado)* Y/O; 2 - O.
span_attributes	object/array	Devuelve solo los spans con los atributos de span indicados. Formato: `[{"key": "<key>", "value": "<value>", "operator": "<operator>"}, ...]`. Una matriz vacía devuelve todos los spans. Valores posibles de operator: 0 - *(predeterminado)* Contiene; 1 - Es igual a; 2 - No contiene; 3 - No es igual a; 4 - Existe; 5 - No existe.
span_attributes_evaltype	integer	Método de evaluación de las condiciones en `span_attributes`. Valores posibles: 0 - *(predeterminado)* Y/O; 2 - O.
min_duration	integer	Devuelve solo los spans cuya duración sea mayor o igual al valor indicado, en nanosegundos.
max_duration	integer	Devuelve solo los spans cuya duración sea menor que el valor indicado, en nanosegundos.
filter	object	Devuelve solo los resultados que coincidan exactamente con el filtro indicado. Acepta un objeto cuyas claves son nombres de propiedades y cuyos valores son un único valor o una matriz de valores con los que se debe comparar. Propiedades compatibles: `traceid`, `spanid`, `parent_spanid`, `trace_state`, `span_name`, `span_kind`, `service_name`, `scope_name`, `scope_version`, `status_code`, `status_message`.

Parámetro	Tipo	Descripción
search	object	Devuelve los resultados que coincidan con el patrón indicado (sin distinguir entre mayúsculas y minúsculas). Acepta un objeto cuyas claves son nombres de propiedades y cuyos valores son cadenas que se deben buscar. Si no se proporcionan opciones adicionales, se realizará una búsqueda `LIKE "%...%"`. Propiedades compatibles: `trace_state`, `span_name`, `service_name`, `scope_name`, `scope_version`, `status_message`.
sortfield	string/array	Ordena el resultado según las propiedades indicadas. Valores posibles: `timestamp`, `traceid`, `spanid`, `parent_spanid`, `trace_state`, `span_name`, `span_kind`, `service_name`, `scope_name`, `scope_version`, `duration`, `status_code`, `status_message`.
offset	integer	Número de registros que se deben omitir antes de devolver el resultado.
countOutput	boolean	Estos parámetros se describen en los comentarios de referencia.
excludeSearch	boolean	
limit	integer	
output	query	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores devueltos

`(integer/array)` Devuelve uno de los siguientes valores:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperación de la jerarquía de spans de un trace

Recupere los primeros cuatro spans del trace "dee625f33cb0994e86c5cd4640e3b09c", ordenados por hora de inicio. La propiedad `parent_spanid` muestra cómo están relacionados los spans: el span raíz no tiene padre y cada uno de los demás spans hace referencia al span que lo inició.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "apmspan.get",
    "params": {
        "output": ["spanid", "parent_spanid", "service_name", "span_name", "duration", "status_code"],
        "time_from": 1790848800,
        "time_till": 1790852400,
        "traceids": "dee625f33cb0994e86c5cd4640e3b09c",
        "sortfield": "timestamp",
        "sortorder": "ASC",
        "limit": 4
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "spanid": "7c1a9e2f4b6d8a03",
```

```
            "parent_spanid": "",
            "service_name": "zabbix-nginx",
            "span_name": "POST /zabbix.php",
            "duration": "1104826531",
            "status_code": "Unset"
        },
        {
            "spanid": "3f8b2c6d1e9a7054",
            "parent_spanid": "7c1a9e2f4b6d8a03",
            "service_name": "zabbix-frontend",
            "span_name": "POST /zabbix.php",
            "duration": "1098230418",
            "status_code": "Unset"
        },
        {
            "spanid": "b3390ad41931ed08",
            "parent_spanid": "3f8b2c6d1e9a7054",
            "service_name": "zabbix-frontend",
            "span_name": "mysqli::real_connect",
            "duration": "36671360",
            "status_code": "Ok"
        },
        {
            "spanid": "e41f7a2c9d5b3086",
            "parent_spanid": "3f8b2c6d1e9a7054",
            "service_name": "zabbix-frontend",
            "span_name": "mysqli_query",
            "duration": "19704118",
            "status_code": "Ok"
        }
    ],
    "id": 1
}
```


Véase también

- Traza de APM
- Registro de APM
- Métrica de APM

Fuente

CApmSpan::get() en *ui/include/classes/api/services/CApmSpan.php*.

Task

Esta clase está diseñada para trabajar con tareas (como comprobar items o reglas de descubrimiento de bajo nivel sin recargar la configuración).

Referencias de objetos:

- Task
  - Objeto de petición 'Ejecutar ahora'
  - Objeto de petición 'Actualizar configuración del proxy'
  - Objeto de petición 'Información de diagnóstico'
    * Objeto de petición de estadísticas
    * Objeto de resultado de estadísticas

Métodos disponibles:

- task.create - crear nuevas tareas
- task.get - recuperar tareas

Objeto Task

Los siguientes objetos están directamente relacionados con la API `task`.

Task

El objeto task tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
taskid	ID	ID de la tarea. Comportamiento de la propiedad: - *solo lectura*
type	integer	Tipo de tarea. Valores posibles: 1 - Información de diagnóstico; 2 - Actualizar configuración del proxy; 6 - Ejecutar ahora. Los usuarios de tipo *Admin* y *User* solo pueden crear tareas 'Ejecutar ahora'. Tenga en cuenta que `task.get` siempre devuelve "7" (Resumen de ejecución de la tarea). Comportamiento de la propiedad: - *obligatorio*
status	integer	Estado de la tarea. Valores posibles: 1 - nueva tarea; 2 - tarea en progreso; 3 - tarea completada; 4 - tarea caducada. Comportamiento de la propiedad: - *solo lectura*
clock	timestamp	Hora en que se creó la tarea. Comportamiento de la propiedad: - *solo lectura*
ttl	integer	El tiempo en segundos después del cual la tarea caduca. Comportamiento de la propiedad: - *solo lectura*
proxyid	ID	ID del proxy sobre el que se recopila la estadística de información de diagnóstico. Comportamiento de la propiedad: - *soportado* si `type` está establecido en "Información de diagnóstico" o "Actualizar configuración del proxy"
request	object	Objeto de solicitud de tarea según el tipo de tarea: El objeto de la tarea 'Ejecutar ahora' se describe en detalle a continuación; El objeto de la tarea 'Actualizar configuración del proxy' se describe en detalle a continuación; El objeto de la tarea 'Información de diagnóstico' se describe en detalle a continuación. Comportamiento de la propiedad: - *obligatorio*

Propiedad	Tipo	Descripción
result	object	Objeto de resultado de la tarea de información de diagnóstico. Puede contener NULL si el resultado aún no está listo. El objeto de resultado se describe en detalle a continuación. Comportamiento de la propiedad: - *solo lectura*

Objeto de solicitud 'Ejecutar ahora'

El objeto de solicitud de tarea 'Ejecutar ahora' tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
itemid	ID	ID del item y reglas de bajo nivel de descubrimiento. Los usuarios de tipo *Admin* y *User* pueden 'Ejecutar ahora' items en hosts para los que tengan permiso de *lectura-escritura*, o permiso de *lectura* y la acción `invoke_execute_now` habilitada para su rol. Lo mismo se aplica a los usuarios de tipo *Admin* para las reglas de bajo nivel de descubrimiento (LLD).

Objeto de solicitud 'Actualizar configuración del proxy'

El objeto de solicitud de la tarea 'Actualizar configuración del proxy' tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
proxyids	array	IDs de proxy.

Objeto de solicitud de 'Información de diagnóstico'

El objeto de solicitud de tarea de información de diagnóstico tiene las siguientes propiedades. El objeto de solicitud de estadísticas para todos los tipos de propiedades se describe en detalle más abajo.

Propiedad	Tipo	Descripción
historycache	object	Solicitud de estadísticas de la caché de histórico. Disponible en server y proxy.
valuecache	object	Solicitud de estadísticas de la caché de items. Disponible en server.
preprocessing	object	Solicitud de estadísticas del gestor de preprocesamiento. Disponible en server y proxy.
alerting	object	Solicitud de estadísticas del gestor de alertas. Disponible en server.
lld	object	Solicitud de estadísticas del gestor de LLD. Disponible en server.

Objeto de solicitud de estadísticas

El objeto de solicitud de estadísticas se utiliza para definir qué tipo de información debe recopilarse sobre los procesos internos del server/proxy. Tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
stats	query	Propiedades del objeto de estadísticas que se devolverán. La lista de campos disponibles para cada tipo de información estadística de diagnóstico se describe en detalle a continuación. Por defecto: `extend` devolverá todos los campos estadísticos disponibles.

Propiedad	Tipo	Descripción
top	object	Objeto para ordenar y limitar los valores estadísticos devueltos. La lista de campos disponibles para cada tipo de información estadística de diagnóstico se describe en detalle a continuación. Ejemplo: { “source.alerts”: 10 }

Lista de campos estadísticos disponibles para cada tipo de solicitud de información de diagnóstico

Se pueden solicitar los siguientes campos estadísticos para cada tipo de propiedad de solicitud de información de diagnóstico.

Tipo de diagnóstico	Campos disponibles	Descripción
historycache	items	Número de métricas en caché.
	values	Número de valores en caché.
	memory	Estadísticas de memoria compartida (espacio libre, número de fragmentos usados, número de fragmentos libres, tamaño máximo del fragmento libre).
	memory.data	Estadísticas de memoria compartida de la caché de datos históricos.
	memory.index	Estadísticas de memoria compartida de la caché de índices históricos.
valuecache	items	Número de métricas en caché.
	values	Número de valores en caché.
	memory	Estadísticas de memoria compartida (espacio libre, número de fragmentos usados, número de fragmentos libres, tamaño máximo del fragmento libre).
	mode	Modo de caché de valores.
preprocessing	values	Número de valores en cola.
	preproc.values	Número de valores en cola con pasos de preprocesamiento.
alerting	alerts	Número de alertas en cola.
lld	rules	Número de reglas en cola.
	values	Número de valores en cola.

Lista de campos de ordenación disponibles para cada tipo de solicitud de información de diagnóstico

Se pueden utilizar los siguientes campos estadísticos para ordenar y limitar la información solicitada.

Tipo de diagnóstico	Campos disponibles	Tipo
historycache	values	integer
valuecache	values	integer
	request.values	integer
preprocessing	values	integer
alerting	media.alerts	integer
	source.alerts	integer
lld	values	integer

Objeto de resultado estadístico

El objeto de resultado estadístico se recupera en el campo `result` del objeto de tarea.

Propiedad	Tipo	Descripción
status	integer	Estado del resultado de la tarea. Valores posibles: -1 - se produjo un error durante la ejecución de la tarea; 0 - el resultado de la tarea se ha creado. Comportamiento de la propiedad: - *sólo lectura*

Propiedad	Tipo	Descripción
data	string/object	Resultados según el objeto de solicitud estadística de una tarea de información de diagnóstico concreta. Contiene el mensaje de error si se produjo un error durante la ejecución de la tarea.

task.create

Descripción

`object task.create(object/array tasks)`

Este método permite crear tareas.

Note:
Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(objeto/array)` Una tarea a crear.

El método acepta tareas con las propiedades estándar de la tarea.

Tenga en cuenta que las tareas de 'Ejecutar ahora' solo se pueden crear para los siguientes tipos de elementos/reglas de descubrimiento:

- Agente Zabbix (pasivo)
- Comprobación simple
- Agente SNMP (v1/v2/v3)
- Zabbix interno
- Comprobación externa
- Monitor de base de datos
- Agente HTTP
- Agente IPMI
- Agente SSH
- Agente TELNET
- Agente JMX
- Calculado
- Elemento dependiente
- Script
- Navegador

Si el elemento/regla de descubrimiento es de tipo "Elemento dependiente", entonces su elemento principal también debe ser uno de los tipos anteriores.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las tareas creadas bajo la propiedad `taskids`. Se crea una tarea por cada métrica y regla de bajo nivel de descubrimiento. El orden de los IDs devueltos coincide con el orden de los `itemids` proporcionados.

Ejemplos

Creación de una tarea

Cree una tarea de 'Ejecutar ahora' para un elemento y una regla de bajo nivel de descubrimiento.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "task.create",
    "params": [
        {
            "type": 6,
            "request": {
                "itemid": "10092"
            }
```

```
        },
        {
            "type": 6,
            "request": {
                "itemid": "10093"
            }
        }
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "taskids": [
            "1",
            "2"
        ]
    },
    "id": 1
}
```

Cree una tarea de 'Actualizar configuración del proxy' para dos proxies.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "task.create",
    "params": [
        {
            "type": 2,
            "request": {
                "proxyids": ["10459", "10460"]
            }
        }
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "taskids": [
            "1"
        ]
    },
    "id": 1
}
```

Cree una tarea de 'Información de diagnóstico'.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "task.create",
    "params": [
        {
            "type": 1,
            "request": {
                "alerting": {
                    "stats": [
```

```
                        "alerts"
                    ],
                    "top": {
                        "media.alerts": 10
                    }
                },
                "lld": {
                    "stats": "extend",
                    "top": {
                        "values": 5
                    }
                }
            },
            "proxyid": 0
        }
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "taskids": [
            "3"
        ]
    },
    "id": 1
}
```

Véase también

- Tarea
- Objeto de solicitud 'Ejecutar ahora'
- Objeto de solicitud 'Actualizar configuración de proxy'
- Objeto de solicitud 'Información de diagnóstico'

Fuente

CTask::create() en *ui/include/classes/api/services/CTask.php*.

task.get

Descripción

`integer/array task.get(object parameters)`

El método permite recuperar tareas según los parámetros proporcionados.

Note:
Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método soporta los siguientes parámetros.

Parámetro	Tipo	Descripción
taskids	ID/array	Devolver solo las tareas con los IDs dados.
output	query	Estos parámetros se describen en el comentario de referencia.
preservekeys	boolean	

Valores de retorno

`(entero/array)` Devuelve un array de objetos.

Ejemplos

Recuperar tarea por ID

Recuperar todos los datos sobre la tarea con el ID "1".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "task.get",
    "params": {
        "output": "extend",
        "taskids": "1"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "taskid": "1",
            "type": "7",
            "status": "3",
            "clock": "1601039076",
            "ttl": "3600",
            "proxyid": null,
            "request": {
                "alerting": {
                    "stats": [
                        "alerts"
                    ],
                    "top": {
                        "media.alerts": 10
                    }
                },
                "lld": {
                    "stats": "extend",
                    "top": {
                        "values": 5
                    }
                }
            },
            "result": {
                "data": {
                    "alerting": {
                        "alerts": 0,
                        "top": {
                            "media.alerts": []
                        },
                        "time": 0.000663
                    },
                    "lld": {
                        "rules": 0,
                        "values": 0,
                        "top": {
                            "values": []
                        },
                        "time": 0.000442
                    }
```

```
                },
                "status": "0"
            }
        }
    ],
    "id": 1
}
```

Véase también

- Tarea
- Objeto de solicitud 'Ejecutar ahora'
- Objeto de solicitud 'Actualizar configuración de proxy'
- Objeto de solicitud 'Información de diagnóstico'

Fuente

CTask::get() en *ui/include/classes/api/services/CTask.php*.

Template

Esta clase está diseñada para trabajar con templates.

Referencias de objetos:

- Template
- Etiqueta de template

Métodos disponibles:

- template.create - crear nuevos templates
- template.delete - eliminar templates
- template.get - recuperar templates
- template.massadd - añadir objetos relacionados a los templates
- template.massremove - eliminar objetos relacionados de los templates
- template.update - actualizar templates

Objeto de plantilla

Los siguientes objetos están directamente relacionados con la API de `plantilla`.

Template

El objeto template tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
templateid	ID	ID de la template. Comportamiento de la propiedad: - *solo lectura* - *requerido* para operaciones de actualización
description	text	Descripción de la template.
host	string	Nombre técnico de la template. Comportamiento de la propiedad: - *requerido* para operaciones de creación
name	string	Nombre visible de la template. Por defecto: valor de la propiedad `host`.
readme	text	Instrucciones de configuración específicas de la template que se mostrarán en el Asistente de host. Soporta formato Markdown.
uuid	string	Identificador único universal, utilizado para vincular templates importadas con las ya existentes. Se genera automáticamente si no se proporciona.

Propiedad	Tipo	Descripción
vendor_name	string	Nombre del proveedor de la template. Para operaciones de creación, tanto `vendor_name` como `vendor_version` deben establecerse o dejarse vacíos. Para operaciones de actualización, `vendor_version` puede dejarse vacío si tiene un valor en la base de datos.
vendor_version	string	Versión del proveedor de la template. Para operaciones de creación, tanto `vendor_name` como `vendor_version` deben establecerse o dejarse vacíos. Para operaciones de actualización, `vendor_name` puede dejarse vacío si tiene un valor en la base de datos.
wizard_ready	integer	Indica si la template está disponible para su selección en el Asistente de host. Valores posibles: 0 - *(por defecto)* No disponible. 1 - Disponible.

Etiqueta de template

El objeto de etiqueta de template tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
tag	string	Nombre de la etiqueta de template. Comportamiento de la propiedad: - *obligatorio*
value	string	Valor de la etiqueta de template. Comportamiento de la propiedad: - *solo lectura*.
object	integer	Tipo de objeto del que se heredó la etiqueta. Valores posibles: 0 - template. Comportamiento de la propiedad: - *solo lectura*.
objectid	ID	ID del objeto del que se heredó la etiqueta. Comportamiento de la propiedad: - *solo lectura*.

template.create

Descripción

`object template.create(object/array templates)`

Este método permite crear nuevos templates.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object/array)` Templates a crear.

Además de las propiedades estándar de template, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
groups	object/array	Grupos de template a los que añadir el template. Los grupos de template deben tener sólo la propiedad `groupid` definida. Comportamiento del parámetro: - *obligatorio*
tags	object/array	Etiquetas de template.
templates	object/array	Templates a enlazar con el template. Los templates deben tener sólo la propiedad `templateid` definida.
macros	object/array	Macros de usuario a crear para el template.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las plantillas creadas bajo la propiedad `templateids`. El orden de los IDs devueltos coincide con el orden de las plantillas enviadas.

Ejemplos

Creando una plantilla

Cree una plantilla con etiquetas y vincule dos plantillas a esta plantilla.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "template.create",
    "params": {
        "host": "Linux template",
        "groups": {
            "groupid": 1
        },
        "templates": [
            {
                "templateid": "11115"
            },
            {
                "templateid": "11116"
            }
        ],
        "tags": [
            {
                "tag": "host-name",
                "value": "{HOST.NAME}"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "11117"
        ]
    },
    "id": 1
}
```

Fuente

CTemplate::create() en *ui/include/classes/api/services/CTemplate.php*.

template.delete

Descripción

`object template.delete(array templateIds)`

Este método le permite eliminar templates.

Eliminar un template también elimina todas sus entidades (items, triggers, gráficos, etc.) de cualquier host o template al que esté vinculado. Para conservar estas entidades, primero desvincule el template de los hosts o templates relevantes usando los métodos `host.update`, `host.massremove`, `template.update` o `template.massremove`.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(array)` IDs de las plantillas a eliminar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las plantillas eliminadas bajo la propiedad `templateids`.

Ejemplos

Eliminando varias plantillas

Eliminar dos plantillas.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "template.delete",
    "params": [
        "13",
        "32"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "13",
            "32"
        ]
    },
    "id": 1
}
```

Fuente

CTemplate::delete() en *ui/include/classes/api/services/CTemplate.php*.

template.get

Descripción

`integer/array template.get(object parameters)`

El método permite recuperar templates según los parámetros proporcionados.

Note:
Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
templateids	ID/array	Devolver solo las templates con los IDs de template dados.
groupids	ID/array	Devolver solo las templates que pertenecen a los grupos de templates dados.
parentTemplateids	ID/array	Devolver solo las templates a las que está vinculada la template dada.
hostids	ID/array	Devolver solo las templates que están vinculadas a los hosts/templates dados.
graphids	ID/array	Devolver solo las templates que contienen los gráficos dados.
itemids	ID/array	Devolver solo las templates que contienen los items dados.
triggerids	ID/array	Devolver solo las templates que contienen los triggers dados.
with_items	flag	Devolver solo las templates que tienen items.
with_triggers	flag	Devolver solo las templates que tienen triggers.
with_graphs	flag	Devolver solo las templates que tienen gráficos.
with_httptests	flag	Devolver solo las templates que tienen escenarios web.
evaltype	integer	Método de evaluación de la etiqueta. Valores posibles: 0 - *(por defecto)* And/Or; 2 - Or.
tags	object/array	Devolver solo las templates con las etiquetas dadas. Formato: `[{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]`. Un array vacío devuelve todas las templates. Valores posibles de operator: 0 - *(por defecto)* Contiene; 1 - Es igual a; 2 - No contiene; 3 - No es igual a; 4 - Existe; 5 - No existe.
inheritedTags	boolean	Devolver templates que tengan las `tags` dadas también en las templates vinculadas. Valores posibles: `true` - las templates vinculadas también deben tener las etiquetas dadas; `false` - *(por defecto)* se ignoran las etiquetas de las templates vinculadas.
selectInheritedTags	query	Devolver una propiedad `inheritedTags` con las etiquetas que están en las templates vinculadas.
selectTags	query	Devolver las etiquetas de la template en la propiedad `tags`.
selectDiscoveryRules	query	Devolver una propiedad `discoveryRules` con las reglas LLD de la template. Soporta `count`.
selectHosts	query	Devolver los hosts que están vinculados a la template en la propiedad `hosts`. Soporta `count`.
selectTemplateGroups	query	Devolver los grupos de templates a los que pertenece la template en la propiedad `templategroups`.

Parámetro	Tipo	Descripción
selectTemplates	query	Devolver las templates a las que está vinculada la template dada en la propiedad `templates`. Soporta `count`.
selectParentTemplates	query	Devolver las templates que están vinculadas a la template dada en la propiedad `parentTemplates`. Soporta `count`.
selectHttpTests	query	Devolver los escenarios web de la template en la propiedad `httpTests`. Soporta `count`.
selectItems	query	Devolver los items de la template en la propiedad `items`. Soporta `count`.
selectTriggers	query	Devolver los triggers de la template en la propiedad `triggers`. Soporta `count`.
selectGraphs	query	Devolver los gráficos de la template en la propiedad `graphs`. Soporta `count`.
selectMacros	query	Devolver las macros de la template en la propiedad `macros`.
selectDashboards	query	Devolver los dashboards de la template en la propiedad `dashboards`. Soporta `count`.
selectValueMaps	query	Devolver una propiedad `valuemaps` con los mapas de valores de la template.
limitSelects	integer	Limita el número de registros devueltos por subconsultas. Se aplica a las siguientes subconsultas: `selectTemplates` - los resultados se ordenarán por `name`; `selectHosts` - ordenados por `host`; `selectParentTemplates` - ordenados por `host`; `selectItems` - ordenados por `name`; `selectDiscoveryRules` - ordenados por `name`; `selectTriggers` - ordenados por `description`; `selectGraphs` - ordenados por `name`; `selectDashboards` - ordenados por `name`.
sortfield	string/array	Ordenar el resultado por las propiedades dadas. Valores posibles: `hostid`, `host`, `name`, `status`.
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	
selectDiscoveries	query	Devolver los descubrimientos de bajo nivel de la template en la propiedad `discoveries`. Soporta `count`. Esta consulta está **obsoleta**, por favor use `selectDiscoveryRules` en su lugar.

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperar templates por nombre

Recuperar todos los datos sobre dos templates llamados "Linux by Zabbix agent" y "Windows by Zabbix agent".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "template.get",
    "params": {
        "output": "extend",
        "filter": {
            "host": [
                "Linux by Zabbix agent",
                "Windows by Zabbix agent"
            ]
        }
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "proxyid": "0",
            "host": "Linux by Zabbix agent",
            "status": "3",
            "ipmi_authtype": "-1",
            "ipmi_privilege": "2",
            "ipmi_username": "",
            "ipmi_password": "",
            "maintenanceid": "0",
            "maintenance_status": "0",
            "maintenance_type": "0",
            "maintenance_from": "0",
            "name": "Linux by Zabbix agent",
            "flags": "0",
            "templateid": "10001",
            "description": "This is an official Linux template. It requires Zabbix agent 8.0 or newer.\r\n
            "tls_connect": "1",
            "tls_accept": "1",
            "tls_issuer": "",
            "tls_subject": "",
            "tls_psk_identity": "",
            "tls_psk": "",
            "custom_interfaces": "0",
            "uuid": "f8f7908280354f2abeed07dc788c3747",
            "vendor_name": "Zabbix",
            "vendor_version": "8.0-2",
            "proxy_groupid": "0",
            "monitored_by": "0",
            "wizard_ready": "1",
            "readme": "## Overview\r\n\r\nThis is an official Linux template. It requires Zabbix agent 8.0
        },
        {
```

```
            "proxyid": "0",
            "host": "Windows by Zabbix agent",
            "status": "3",
            "ipmi_authtype": "-1",
            "ipmi_privilege": "2",
            "ipmi_username": "",
            "ipmi_password": "",
            "maintenanceid": "0",
            "maintenance_status": "0",
            "maintenance_type": "0",
            "maintenance_from": "0",
            "name": "Windows by Zabbix agent",
            "flags": "0",
            "templateid": "10081",
            "description": "This is an official Windows template. It requires Zabbix agent 8.0 or newer.\r
            "tls_connect": "1",
            "tls_accept": "1",
            "tls_issuer": "",
            "tls_subject": "",
            "tls_psk_identity": "",
            "tls_psk": "",
            "custom_interfaces": "0",
            "uuid": "13b06904a6bf41cbb795e3193d896340",
            "vendor_name": "Zabbix",
            "vendor_version": "8.0-2",
            "proxy_groupid": "0",
            "monitored_by": "0",
            "wizard_ready": "1",
            "readme": "## Overview\r\n\r\nThis is an official Windows template. It requires Zabbix agent 8
        }
    ],
    "id": 1
}
```


Recuperar grupos de plantillas

Recuperar los grupos de plantillas a los que pertenece la plantilla "Linux by Zabbix agent".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "template.get",
    "params": {
        "output": ["hostid"],
        "selectTemplateGroups": "extend",
        "filter": {
            "host": [
                "Linux by Zabbix agent"
            ]
        }
    },
    "id": 1
}
```

Respuesta:


```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "templateid": "10001",
            "templategroups": [
                {
                    "groupid": "10",
```

```
                    "name": "Templates/Operating systems",
                    "uuid": "846977d1dfed4968bc5f8bdb363285bc"
                }
            ]
        }
    ],
    "id": 1
}
```

Recuperar equipos por plantilla

Recuperar los equipos que tienen la plantilla "10001" (*Linux por Zabbix agent*) vinculada a ellos.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "template.get",
    "params": {
        "output": "templateid",
        "templateids": "10001",
        "selectHosts": ["hostid", "name"]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "templateid": "10001",
            "hosts": [
                {
                    "hostid": "10084",
                    "name": "Zabbix server"
                },
                {
                    "hostid": "10603",
                    "name": "Host 1"
                },
                {
                    "hostid": "10604",
                    "name": "Host 2"
                }
            ]
        }
    ],
    "id": 1
}
```

Búsqueda por etiquetas de template

Recuperar templates que tengan la etiqueta "host-name" igual a "{HOST.NAME}".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "template.get",
    "params": {
        "output": ["hostid"],
        "selectTags": "extend",
        "evaltype": 0,
        "tags": [
            {
```

```
                "tag": "host-name",
                "value": "{HOST.NAME}",
                "operator": 1
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "templateid": "10402",
            "tags": [
                {
                    "tag": "host-name",
                    "value": "{HOST.NAME}"
                }
            ]
        }
    ],
    "id": 1
}
```

Ver también

- Grupo de templates
- Template
- Macro de usuario
- Interfaz de host

Fuente

CTemplate::get() en *ui/include/classes/api/services/CTemplate.php*.

template.massadd

Descripción

`object template.massadd(object parameters)`

Este método permite agregar simultáneamente varios objetos relacionados a los templates dados.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object)` Parámetros que contienen los IDs de las templates a actualizar y los objetos a añadir a las templates.

El método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
templates	object/array	Templates que se actualizarán. Las templates deben tener definida únicamente la propiedad `templateid`. Comportamiento del parámetro: - *obligatorio*

Parámetro	Tipo	Descripción
groups	object/array	Grupos de templates a los que añadir las templates indicadas. Los grupos de templates deben tener definida únicamente la propiedad `groupid`.
macros	object/array	Macros de usuario que se crearán para las templates indicadas.
templates_link	object/array	Templates a enlazar con las templates indicadas. Las templates deben tener definida únicamente la propiedad `templateid`.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las plantillas actualizadas bajo la propiedad `templateids`.

Ejemplos

Vincular un grupo a plantillas

Añadir el grupo de plantillas ”2” a dos plantillas.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "template.massadd",
    "params": {
        "templates": [
            {
                "templateid": "10085"
            },
            {
                "templateid": "10086"
            }
        ],
        "groups": [
            {
                "groupid": "2"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10085",
            "10086"
        ]
    },
    "id": 1
}
```

Vincular dos plantillas a una plantilla

Vincular las plantillas ”10106” y ”10104” a la plantilla ”10073”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "template.massadd",
    "params": {
```

```
        "templates": [
            {
                "templateid": "10073"
            }
        ],
        "templates_link": [
            {
                "templateid": "10106"
            },
            {
                "templateid": "10104"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10073"
        ]
    },
    "id": 1
}
```

Ver también

- template.update
- Host
- Grupo de template
- Macro de usuario

Fuente

CTemplate::massAdd() en *ui/include/classes/api/services/CTemplate.php*.

template.massremove

Descripción

`object template.massremove(object parameters)`

Este método permite eliminar objetos relacionados de múltiples templates.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object)` Parámetros que contienen los IDs de las templates a actualizar y los objetos que deben eliminarse.

Parámetro	Tipo	Descripción
templateids	ID/array	IDs de las templates que se actualizarán. Comportamiento del parámetro: - *obligatorio*
groupids	ID/array	IDs de los grupos de templates de los que se eliminarán las templates indicadas.
macros	string/array	IDs de las macros de usuario que se eliminarán de las templates indicadas.

Parámetro	Tipo	Descripción
templateids_clear	ID/array	IDs de las templates que se desvincularán y borrarán de las templates indicadas (ascendentes).
templateids_link	ID/array	IDs de las templates que se desvincularán de las templates indicadas (ascendentes).

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las plantillas actualizadas bajo la propiedad `templateids`.

Ejemplos

Eliminando plantillas de un grupo

Eliminar dos plantillas del grupo ”2”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "template.massremove",
    "params": {
        "templateids": [
            "10085",
            "10086"
        ],
        "groupids": "2"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10085",
            "10086"
        ]
    },
    "id": 1
}
```

Desvincular plantillas de un equipo

Desvincular las plantillas ”10106” y ”10104” de la plantilla ”10085”.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "template.massremove",
    "params": {
        "templateids": "10085",
        "templateids_link": [
            "10106",
            "10104"
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
```

```
        "templateids": [
            "10085"
        ]
    },
    "id": 1
}
```

Ver también

- template.update
- Macro de usuario

Fuente

CTemplate::massRemove() en *ui/include/classes/api/services/CTemplate.php*.

template.update

Descripción

`object template.update(object/array templates)`

Este método permite actualizar templates existentes.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object/array)` Propiedades de la template que se actualizarán.

La propiedad `templateid` debe estar definida para cada template, todas las demás propiedades son opcionales. Sólo se actualizarán las propiedades indicadas, todas las demás permanecerán sin cambios.

Además de las propiedades estándar de la template, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
groups	object/array	Grupos de templates para reemplazar los grupos de templates actuales a los que pertenecen las templates. Los grupos de templates sólo deben tener definida la propiedad `groupid`.
tags	object/array	Etiquetas de template para reemplazar las etiquetas de template actuales.
macros	object/array	Macros de usuario para reemplazar las macros de usuario actuales en las templates indicadas.
templates	object/array	Templates para reemplazar las templates actualmente enlazadas. Las templates que no se pasen sólo se desvinculan. Las templates sólo deben tener definida la propiedad `templateid`.
templates_clear	object/array	Templates para desvincular y limpiar de las templates indicadas. Las templates sólo deben tener definida la propiedad `templateid`.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de las plantillas actualizadas bajo la propiedad `templateids`.

Ejemplos

Cambiando las propiedades estándar de la plantilla

Cambie el nombre técnico de la plantilla a ”Linux by Zabbix agent Custom”, el nombre visible a ”Mi plantilla” y actualice la descripción de la plantilla.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "template.update",
    "params": {
        "templateid": "10086",
        "host": "Linux by Zabbix agent Custom",
        "name": "Mi plantilla",
        "description": "Esta es una plantilla personalizada de Linux."
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10086"
        ]
    },
    "id": 1
}
```

Actualización de grupos de plantillas

Reemplace todos los grupos de plantillas para la plantilla dada por uno diferente.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "template.update",
    "params": {
        "templateid": "10086",
        "groups": [
            {
                "groupid": "24"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10086"
        ]
    },
    "id": 1
}
```

Actualización de varios grupos de templates

Reemplazar todos los grupos de templates para varios templates con diferentes grupos.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "template.update",
    "params": [
        {
            "templateid": "10086",
```

```
            "groups": [
                {
                    "groupid": "24"
                }
            ]
        },
        {
            "templateid": "10087",
            "groups": [
                {
                    "groupid": "1"
                },
                {
                    "groupid": "12"
                }
            ]
        },
        {
            "templateid": "10088",
            "groups": [
                {
                    "groupid": "1"
                },
                {
                    "groupid": "12"
                }
            ]
        }
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10086",
            "10087",
            "10088"
        ]
    },
    "id": 1
}
```

Actualización de etiquetas de plantilla

Reemplace todas las etiquetas de la plantilla por una diferente.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "template.update",
    "params": {
        "templateid": "10086",
        "tags": [
            {
                "tag": "host-name",
                "value": "{HOST.NAME}"
            }
        ]
    },
    "id": 1
```

```
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10086"
        ]
    },
    "id": 1
}
```

Actualización de macros de plantilla

Reemplace todas las macros de la plantilla por una diferente.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "template.update",
    "params": {
        "templateid": "10086",
        "macros": [
            {
                "macro": "{$MY_MACRO}",
                "value": "new_value"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10086"
        ]
    },
    "id": 1
}
```

Actualización de múltiples macros de template

Reemplazar todas las macros de usuario con la macro de usuario dada en múltiples templates.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "template.update",
    "params": [
            {
                "templateid": "10087",
                "macros": [
                    {
                        "macro": "{$AGENT.TIMEOUT}",
                        "value": "5m",
                        "description": "Timeout after which agent is considered unavailable. Works only fo
                    }
                ]
            },
            {
```

```
                "templateid": "10088",
                "macros": [
                    {
                        "macro": "{$AGENT.TIMEOUT}",
                        "value": "5m",
                        "description": "Timeout after which agent is considered unavailable. Works only fo
                    }
                ]
            }
        ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10087",
            "10088"
        ]
    },
    "id": 1
}
```

Actualizar las plantillas enlazadas a la template

Desvincula (sin limpiar) todas las templates de la template dada y enlaza una diferente a ella.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "template.update",
    "params": {
        "templateid": "10086",
        "templates": [
            {
                "templateid": "10001"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10086"
        ]
    },
    "id": 1
}
```

Eliminando varias plantillas vinculadas a plantillas

Desvincula y elimina una plantilla vinculada específica de las plantillas indicadas.

Request:

```
{
    "jsonrpc": "2.0",
    "method": "template.update",
    "params": [
```

```
        {
            "templateid": "10087",
            "templates_clear": [
                {
                    "templateid": "10001"
                }
            ]
        },
        {
            "templateid": "10088",
            "templates_clear": [
                {
                    "templateid": "10001"
                }
            ]
        }
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10087",
            "10088"
        ]
    },
    "id": 1
}
```

Fuente

CTemplate::update() en *ui/include/classes/api/services/CTemplate.php*.

Tipo de medio

Esta clase está diseñada para trabajar con tipos de medios.

Referencias de objetos:

- Tipo de medio
  - Parámetros de webhook
  - Parámetros de script
- Plantilla de mensaje

Métodos disponibles:

- mediatype.create - crear nuevos tipos de medios
- mediatype.delete - eliminar tipos de medios
- mediatype.get - obtener tipos de medios
- mediatype.update - actualizar tipos de medios

Objeto tipo de medio

Los siguientes objetos están directamente relacionados con la API `mediatype`.

Tipo de medio

El objeto de tipo de medio tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
mediatypeid	ID	ID del tipo de medio. Comportamiento de la propiedad: - *solo lectura* - *obligatoria* para operaciones de actualización
name	string	Nombre del tipo de medio. Comportamiento de la propiedad: - *obligatoria* para operaciones de creación
type	integer	Transporte utilizado por el tipo de medio. Valores posibles: 0 - Correo electrónico; 1 - Script; 2 - SMS; 4 - Webhook; 5 - Push. Comportamiento de la propiedad: - *obligatoria* para operaciones de creación
exec_path	string	Nombre del archivo de script (por ejemplo, notification.sh) ubicado en el directorio especificado en el parámetro de configuración del server AlertScriptsPath. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”Script”
gsm_modem	string	Nombre del dispositivo serie del módem GSM. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”SMS”
passwd	string	Contraseña de autenticación. Comportamiento de la propiedad: - *compatible* si `smtp_authentication` está establecido en ”Contraseña normal”
provider	integer	Proveedor de correo electrónico. Valores posibles: 0 - *(predeterminado)* SMTP genérico; 1 - Gmail; 2 - Retransmisión de Gmail; 3 - Office365; 4 - Retransmisión de Office365.
smtp_email	string	Dirección de correo electrónico desde la que se enviarán las notificaciones. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”Email”
smtp_helo	string	HELO de SMTP. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”Email”
smtp_server	string	server SMTP. Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”Email”
smtp_port	integer	Puerto del server SMTP al que conectarse. Predeterminado: 25. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”Email”

Propiedad	Tipo	Descripción
smtp_security	integer	Nivel de seguridad de la conexión SMTP que se utilizará. Valores posibles: 0 - *(predeterminado)* Ninguno; 1 - STARTTLS; 2 - SSL/TLS. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”Email”
smtp_verify_host	integer	Verificación SSL del host para SMTP. Valores posibles: 0 - *(predeterminado)* No; 1 - Sí. Comportamiento de la propiedad: - *compatible* si `smtp_security` está establecido en ”STARTTLS” o ”SSL/TLS”
smtp_verify_peer	integer	Verificación SSL del par para SMTP. Valores posibles: 0 - *(predeterminado)* No; 1 - Sí. Comportamiento de la propiedad: - *compatible* si `smtp_security` está establecido en ”STARTTLS” o ”SSL/TLS”
smtp_authentication	integer	Método de autenticación SMTP que se utilizará. Valores posibles: 0 - *(predeterminado)* Ninguno; 1 - Contraseña normal; 2 - Token de OAuth. La autenticación OAuth no está permitida para el proveedor de correo electrónico *Retransmisión de Office365*. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”Email”
redirection_url	string	URL del frontend de Zabbix a la que se redirigirá después de la autorización de OAuth. Predeterminado: Valor de la propiedad `url` de la configuración de API con la parte `zabbix.php?action=oauth.authorize` Comportamiento de la propiedad: - *obligatoria* si `smtp_authentication` está establecido en ”Token de OAuth”
client_id	string	Identificador del cliente registrado en el server de autorización OAuth. Comportamiento de la propiedad: - *obligatoria* si `smtp_authentication` está establecido en ”Token de OAuth”
client_secret	string	Secreto del cliente registrado en el server de autorización OAuth. Solo está disponible para usuarios de tipo Superadministrador. Comportamiento de la propiedad: - *obligatoria* si `smtp_authentication` está establecido en ”Token de OAuth”

Propiedad	Tipo	Descripción
authorization_url	string	URL de OAuth, con parámetros, para obtener tokens de acceso y actualización. Comportamiento de la propiedad: - *obligatoria* si `smtp_authentication` está establecido en ”Token de OAuth”
token_url	string	URL de OAuth para intercambiar el token de autorización por tokens de acceso y actualización. El server también utiliza esta URL para actualizar un token de acceso no válido. Comportamiento de la propiedad: - *obligatoria* si `smtp_authentication` está establecido en ”Token de OAuth”
tokens_status	integer	Máscara de bits del estado de los tokens. Valores posibles: 0 - (predeterminado) Ambos tokens contienen un valor no válido 1 - El token de acceso contiene un valor válido 2 - El token de actualización contiene un valor válido 3 - Ambos tokens contienen valores válidos. Comportamiento de la propiedad: - *compatible* si `smtp_authentication` está establecido en ”Token de OAuth”
access_token	string	Valor del token de acceso de OAuth. Comportamiento de la propiedad: - *obligatoria* si `smtp_authentication` está establecido en ”Token de OAuth”
access_token_updated	timestamp	Marca de tiempo de la última modificación de `access_token` realizada por el server al actualizarlo con `refresh_token` o mediante la API cuando cambian los tokens. Comportamiento de la propiedad: - *compatible* si `smtp_authentication` está establecido en ”Token de OAuth”
access_expires_in	integer	Tiempo, en segundos, hasta que `access_token` quede obsoleto y sea necesario realizar una solicitud a `refresh_url`. El server de Zabbix lo establece al actualizar `access_token`, o la API cuando cambian los tokens. La marca de tiempo se calcula sumando el valor de `access_token_updated`. Comportamiento de la propiedad: - *compatible* si `smtp_authentication` está establecido en ”Token de OAuth”
refresh_token	string	Valor del token de actualización de OAuth. Comportamiento de la propiedad: - *obligatoria* si `smtp_authentication` está establecido en ”Token de OAuth”
status	integer	Indica si el tipo de medio está habilitado. Valores posibles: 0 - *(predeterminado)* Habilitado; 1 - Deshabilitado.

Propiedad	Tipo	Descripción
username	string	Nombre de usuario. Comportamiento de la propiedad: - *compatible* si `smtp_authentication` está establecido en ”Contraseña normal”
maxsessions	integer	Número máximo de alertas que se pueden procesar en paralelo. Valores posibles si `type` está establecido en ”SMS”: 1. Valores posibles si `type` está establecido en ”Email”, ”Script” o ”Webhook”: 0-100. Predeterminado: 1.
maxattempts	integer	Número máximo de intentos para enviar una alerta. Valores posibles: 1-100. Predeterminado: 3.
attempt_interval	string	Intervalo entre los intentos de reenvío. Acepta un número de segundos (por ejemplo, 1, 30, 60) o un valor con un sufijo de tiempo (por ejemplo, `30s`, `1m`, `1h`). Valores posibles: 0-1h. Predeterminado: 10s.
message_format	integer	Formato del mensaje. Valores posibles: 0 - Texto sin formato; 1 - *(predeterminado)* HTML. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”Email”
script	text	Cuerpo del script del Webhook (JavaScript). Comportamiento de la propiedad: - *obligatoria* si `type` está establecido en ”Webhook”
timeout	string	Tiempo de espera del script del Webhook. Acepta un número de segundos (por ejemplo, 1, 60) o un valor con un sufijo de tiempo (por ejemplo, `30s`, `1m`). Valores posibles: 1-60s. Predeterminado: 30s.
process_tags	integer	Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”Webhook” Procesa los valores de las propiedades JSON de la respuesta del script del Webhook como etiquetas. Estas etiquetas se añaden a las etiquetas de problema existentes. Valores posibles: 0 - *(predeterminado)* Ignorar la respuesta del script del Webhook; 1 - Procesar la respuesta del script del Webhook como etiquetas. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”Webhook”

Propiedad	Tipo	Descripción
show_event_menu	integer	Incluye una entrada en el menú de eventos que enlaza con una URL personalizada. También añade la propiedad `urls` a la salida de problem.get y event.get. Valores posibles: 0 - *(predeterminado)* No incluir la entrada del menú de eventos ni la propiedad `urls`; 1 - Incluir la entrada del menú de eventos y la propiedad `urls`. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”Webhook”
event_menu_url	string	URL utilizada en la entrada del menú de eventos y en la propiedad `urls` devuelta por problem.get y event.get. Comportamiento de la propiedad: - *compatible* si `show_event_menu` está establecido en ”Incluir la entrada del menú de eventos y la propiedad `urls`”
event_menu_name	string	Nombre utilizado para la entrada del menú de eventos y en la propiedad `urls` devuelta por problem.get y event.get. Comportamiento de la propiedad: - *compatible* si `show_event_menu` está establecido en ”Incluir la entrada del menú de eventos y la propiedad `urls`”
parameters	array	Parámetros de Webhook o de script. Comportamiento de la propiedad: - *compatible* si `type` está establecido en ”Webhook” o ”Script”
description	text	Descripción del tipo de medio.

Parámetros de webhook

Los parámetros de webhook tienen las siguientes propiedades.

Propiedad	Tipo	Descripción
name	string	Nombre del parámetro. Comportamiento de la propiedad: - *obligatorio*
value	string	Valor del parámetro, admite macros. Las macros admitidas se describen en la página Macros admitidas.

Parámetros de script

Los parámetros de script tienen las siguientes propiedades.

Propiedad	Tipo	Descripción
sortorder	integer	El orden en el que los valores de los parámetros se pasarán al script como argumentos de línea de comandos, comenzando con 0 como el primero. Comportamiento de la propiedad: - *requerido*
value	string	Valor del parámetro, admite macros. Las macros admitidas se describen en la página Macros soportadas.

Plantilla de mensaje

El objeto de plantilla de mensaje define una plantilla que se utilizará como mensaje por defecto para las operaciones de acción para enviar una notificación. Tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
eventsource	integer	Fuente del evento. Valores posibles: 0 - Triggers; 1 - Descubrimiento; 2 - Autoregistro; 3 - Interno; 4 - Servicios. Comportamiento de la propiedad: - *obligatorio*
recovery	integer	Modo de operación. Valores posibles: 0 - Operaciones; 1 - Operaciones de recuperación; 2 - Operaciones de actualización. Comportamiento de la propiedad: - *obligatorio*
subject	string	Asunto del mensaje.
message	string	Texto del mensaje.

mediatype.create

Descripción

`object mediatype.create(object/array mediaTypes)`

Este método permite crear nuevos tipos de medios.

Note:
Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.
 Si se especifican métodos de entrega en el indicador de funcionalidad `$ZBX_FEATURE_FLAGS['media_type_denylist']` del archivo de configuración del frontend, el usuario no puede llamar al método para los tipos de medios correspondientes.

Parámetros

`(object/array)` Tipos de medios a crear.

Además de las propiedades estándar del tipo de medio, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
message_templates	array	Plantillas de mensajes que se crearán para el tipo de medio.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los tipos de medio creados bajo la propiedad `mediatypeids`. El orden de los IDs devueltos coincide con el orden de los tipos de medio enviados.

Ejemplos

Creando un tipo de medio de correo electrónico

Cree un nuevo tipo de medio de correo electrónico con un puerto SMTP personalizado y plantillas de mensajes.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "mediatype.create",
    "params": {
```

```
        "type": "0",
        "name": "Email",
        "smtp_server": "mail.example.com",
        "smtp_helo": "example.com",
        "smtp_email": "zabbix@example.com",
        "smtp_port": "587",
        "message_format": "1",
        "message_templates": [
            {
                "eventsource": "0",
                "recovery": "0",
                "subject": "Problema: {EVENT.NAME}",
                "message": "El problema \"{EVENT.NAME}\" en el equipo \"{HOST.NAME}\" comenzó a las {EVENT
            },
            {
                "eventsource": "0",
                "recovery": "1",
                "subject": "Resuelto en {EVENT.DURATION}: {EVENT.NAME}",
                "message": "El problema \"{EVENT.NAME}\" en el equipo \"{HOST.NAME}\" se ha resuelto a las
            },
            {
                "eventsource": "0",
                "recovery": "2",
                "subject": "Problema actualizado en {EVENT.AGE}: {EVENT.NAME}",
                "message": "{USER.FULLNAME} {EVENT.UPDATE.ACTION} problema \"{EVENT.NAME}\" en el equipo \
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "mediatypeids": [
            "7"
        ]
    },
    "id": 1
}
```

Creando un tipo de medio de script

Cree un nuevo tipo de medio de script con un valor personalizado para el número de intentos y el intervalo entre ellos.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "mediatype.create",
    "params": {
        "type": "1",
        "name": "Notificaciones push",
        "exec_path": "push-notification.sh",
        "maxattempts": "5",
        "attempt_interval": "11s",
        "parameters": [
            {
                "sortorder": "0",
                "value": "{ALERT.SENDTO}"
            },
            {
                "sortorder": "1",
```

```
                "value": "{ALERT.SUBJECT}"
            },
            {
                "sortorder": "2",
                "value": "{ALERT.MESSAGE}"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "mediatypeids": [
            "8"
        ]
    },
    "id": 1
}
```

Creando un tipo de medio webhook

Cree un nuevo tipo de medio webhook.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "mediatype.create",
    "params": {
        "type": "4",
        "name": "Webhook",
        "script": "var Webhook = {\r\n    token: null,\r\n    to: null,\r\n    subject: null,\r\n    messa
        "parameters": [
            {
                "name": "Message",
                "value": "{ALERT.MESSAGE}"
            },
            {
                "name": "Subject",
                "value": "{ALERT.SUBJECT}"
            },
            {
                "name": "To",
                "value": "{ALERT.SENDTO}"
            },
            {
                "name": "Token",
                "value": "<Token>"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "mediatypeids": [
            "9"
        ]
```

```
    },
    "id": 1
}
```

Fuente

CMediaType::create() en *ui/include/classes/api/services/CMediaType.php*.

mediatype.delete

Descripción

`object mediatype.delete(array mediaTypeIds)`

Este método permite eliminar tipos de medios.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.
 Si se especifican métodos de entrega en el indicador de funcionalidad `$ZBX_FEATURE_FLAGS['media_type_denylist']` del archivo de configuración del frontend, el usuario no podrá llamar al método para los tipos de medios correspondientes.

Parámetros

`(array)` IDs de los tipos de medios a eliminar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los tipos de medios eliminados bajo la propiedad `mediatypeids`.

Ejemplos

Eliminando múltiples tipos de medios

Eliminar dos tipos de medios.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "mediatype.delete",
    "params": [
        "3",
        "5"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "mediatypeids": [
            "3",
            "5"
        ]
    },
    "id": 1
}
```

Fuente

CMediaType::delete() en *ui/include/classes/api/services/CMediaType.php*.

mediatype.get

Descripción

`integer/array mediatype.get(object parameters)`

El método permite recuperar tipos de medios según los parámetros especificados.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración del rol de usuario. Consulte Roles de usuario para obtener más información.
 Si se especifican métodos de entrega en el indicador de funcionalidad `$ZBX_FEATURE_FLAGS['media_type_denylist']` del archivo de configuración del frontend, el usuario no podrá llamar al método para los tipos de medios correspondientes.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

> **Note:**
> Al solicitar información relacionada con el usuario de los tipos de medios, los usuarios de tipo *Admin* y *User* solo pueden recuperar datos sobre su propio usuario. Para un ejemplo, consulte Recuperar tipos de medios como *Admin*.

Parámetro	Tipo	Descripción
mediatypeids	ID/array	Devolver solo los tipos de medios con los IDs dados.
mediaids	ID/array	Devolver solo los tipos de medios utilizados por los medios dados.
userids	ID/array	Devolver solo los tipos de medios utilizados por los usuarios dados.
selectActions	query	Devolver una propiedad `actions` con las acciones que utilizan el tipo de medio.
selectMessageTemplates	query	Devolver una propiedad `message_templates` con un array de mensajes de tipo de medio. Comportamiento del parámetro: - *soportado* para usuarios de tipo *Super admin*
selectUsers	query	Devolver una propiedad `users` con los usuarios que utilizan el tipo de medio. Consulte `user.get` para restricciones basadas en el tipo de usuario.
sortfield	string/array	Ordenar el resultado por las propiedades dadas. Valores posibles: `mediatypeid`.
filter	object	Devolver solo aquellos resultados que coincidan exactamente con el filtro dado. Acepta un objeto, donde las claves son nombres de propiedades y los valores son un único valor o un array de valores con los que comparar. No admite propiedades del tipo de dato `text` data type.
output	query	Posibles propiedades del objeto Media type para usuarios de tipo *Admin* y *User*: `mediatypeid`, `name`, `type`, `status`, `maxattempts`. Propiedades del objeto Media type que se devolverán. Los usuarios de tipo *Admin* y *User* solo pueden recuperar las siguientes propiedades: `mediatypeid`, `name`, `type`, `status`, `maxattempts`, `description`. Para un ejemplo, consulte Recuperar tipos de medios como *Admin*. Por defecto: `extend`.

Parámetro	Tipo	Descripción
search	object	Devolver resultados que coincidan con el patrón dado (no distingue mayúsculas de minúsculas). Acepta un objeto, donde las claves son nombres de propiedades y los valores son cadenas a buscar. Si no se dan opciones adicionales, esto realizará una búsqueda `LIKE "%…%"`. Solo admite propiedades de tipo de dato `string` y `text` data type. Posibles propiedades del objeto Media type para usuarios de tipo *Admin* y *User*: `name`, `description`.
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
editable	boolean	
excludeSearch	boolean	
limit	integer	
preservekeys	boolean	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperar tipos de medios

Recupera todos los tipos de medios configurados. El siguiente ejemplo devuelve dos tipos de medios:

- tipo de medio de correo electrónico;
- tipo de medio SMS.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "mediatype.get",
    "params": {
        "output": "extend",
        "selectMessageTemplates": "extend"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "mediatypeid": "1",
            "type": "0",
            "name": "Email",
            "smtp_server": "mail.example.com",
            "smtp_helo": "example.com",
            "smtp_email": "zabbix@example.com",
            "exec_path": "",
            "gsm_modem": "",
            "username": "",
            "passwd": "",
```

```
        "status": "0",
        "smtp_port": "25",
        "smtp_security": "0",
        "smtp_verify_peer": "0",
        "smtp_verify_host": "0",
        "smtp_authentication": "0",
        "maxsessions": "1",
        "maxattempts": "3",
        "attempt_interval": "10s",
        "message_format": "0",
        "script": "",
        "timeout": "30s",
        "process_tags": "0",
        "show_event_menu": "1",
        "event_menu_url": "",
        "event_menu_name": "",
        "description": "",
        "provider": "0",
        "message_templates": [
            {
                "eventsource": "0",
                "recovery": "0",
                "subject": "Problem: {EVENT.NAME}",
                "message": "Problem started at {EVENT.TIME} on {EVENT.DATE}\r\nProblem name: {EVENT.NA
            },
            {
                "eventsource": "0",
                "recovery": "1",
                "subject": "Resolved: {EVENT.NAME}",
                "message": "Problem has been resolved at {EVENT.RECOVERY.TIME} on {EVENT.RECOVERY.DATE
            },
            {
                "eventsource": "0",
                "recovery": "2",
                "subject": "Updated problem: {EVENT.NAME}",
                "message": "{USER.FULLNAME} {EVENT.UPDATE.ACTION} problem at {EVENT.UPDATE.DATE} {EVEN
            },
            {
                "eventsource": "1",
                "recovery": "0",
                "subject": "Discovery: {DISCOVERY.DEVICE.STATUS} {DISCOVERY.DEVICE.IPADDRESS}",
                "message": "Discovery rule: {DISCOVERY.RULE.NAME}\r\n\r\nDevice IP: {DISCOVERY.DEVICE.
            },
            {
                "eventsource": "2",
                "recovery": "0",
                "subject": "Autoregistration: {HOST.HOST}",
                "message": "Host name: {HOST.HOST}\r\nHost IP: {HOST.IP}\r\nAgent port: {HOST.PORT}"
            }
        ],
        "parameters": []
    },
    {
        "mediatypeid": "3",
        "type": "2",
        "name": "SMS",
        "smtp_server": "",
        "smtp_helo": "",
        "smtp_email": "",
        "exec_path": "",
        "gsm_modem": "/dev/ttyS0",
        "username": "",
```

```
            "passwd": "",
            "status": "0",
            "smtp_port": "25",
            "smtp_security": "0",
            "smtp_verify_peer": "0",
            "smtp_verify_host": "0",
            "smtp_authentication": "0",
            "maxsessions": "1",
            "maxattempts": "3",
            "attempt_interval": "10s",
            "message_format": "1",
            "script": "",
            "timeout": "30s",
            "process_tags": "0",
            "show_event_menu": "1",
            "event_menu_url": "",
            "event_menu_name": "",
            "description": "",
            "provider": "0",
            "message_templates": [
                {
                    "eventsource": "0",
                    "recovery": "0",
                    "subject": "",
                    "message": "{EVENT.SEVERITY}: {EVENT.NAME}\r\nHost: {HOST.NAME}\r\n{EVENT.DATE} {EVENT
                },
                {
                    "eventsource": "0",
                    "recovery": "1",
                    "subject": "",
                    "message": "RESOLVED: {EVENT.NAME}\r\nHost: {HOST.NAME}\r\n{EVENT.DATE} {EVENT.TIME}"
                },
                {
                    "eventsource": "0",
                    "recovery": "2",
                    "subject": "",
                    "message": "{USER.FULLNAME} {EVENT.UPDATE.ACTION} problem at {EVENT.UPDATE.DATE} {EVEN
                },
                {
                    "eventsource": "1",
                    "recovery": "0",
                    "subject": "",
                    "message": "Discovery: {DISCOVERY.DEVICE.STATUS} {DISCOVERY.DEVICE.IPADDRESS}"
                },
                {
                    "eventsource": "2",
                    "recovery": "0",
                    "subject": "",
                    "message": "Autoregistration: {HOST.HOST}\r\nHost IP: {HOST.IP}\r\nAgent port: {HOST.P
                }
            ],
            "parameters": []
        }
    ],
    "id": 1
}
```


Recuperar tipos de medios como *Admin*

Como usuario de tipo *Admin*, recupere todos los tipos de medios que están habilitados, junto con los usuarios que utilizan estos tipos de medios. El siguiente ejemplo devuelve dos tipos de medios:

- tipo de medio de correo electrónico con un usuario (solo el propio usuario de tipo *Admin*);
- tipo de medio SMS sin usuarios.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "mediatype.get",
    "params": {
        "output": "extend",
        "filter": {
            "status": 0
        },
        "selectUsers": "extend"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "mediatypeid": "1",
            "type": "0",
            "name": "Email",
            "status": "0",
            "description": "",
            "maxattempts": "3",
            "users": [
                {
                    "userid": "3",
                    "username": "database-admin",
                    "name": "John",
                    "surname": "Doe",
                    "url": "",
                    "autologin": "0",
                    "autologout": "0",
                    "lang": "default",
                    "refresh": "30s",
                    "theme": "default",
                    "attempt_failed": "0",
                    "attempt_ip": "",
                    "attempt_clock": "0",
                    "rows_per_page": "50",
                    "timezone": "default",
                    "roleid": "2",
                    "provisioned": "0"
                }
            ]
        },
        {
            "mediatypeid": "3",
            "type": "2",
            "name": "SMS",
            "status": "0",
            "description": "",
            "maxattempts": "3",
            "users": []
        }
    ],
    "id": 1
}
```

Recuperar tipos de medios de script y webhook

El siguiente ejemplo devuelve tres tipos de medios:

- tipo de medio script con parámetros;
- tipo de medio script sin parámetros;
- tipo de medio webhook con parámetros.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "mediatype.get",
    "params": {
        "output": ["mediatypeid", "name", "parameters"],
        "filter": {
            "type": [1, 4]
        }
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "mediatypeid": "10",
            "name": "Script with parameters",
            "parameters": [
                {
                    "sortorder": "0",
                    "value": "{ALERT.SENDTO}"
                },
                {
                    "sortorder": "1",
                    "value": "{EVENT.NAME}"
                },
                {
                    "sortorder": "2",
                    "value": "{ALERT.MESSAGE}"
                },
                {
                    "sortorder": "3",
                    "value": "Zabbix alert"
                }
            ]
        },
        {
            "mediatypeid": "13",
            "name": "Script without parameters",
            "parameters": []
        },
        {
            "mediatypeid": "11",
            "name": "Webhook",
            "parameters": [
                {
                    "name": "alert_message",
                    "value": "{ALERT.MESSAGE}"
                },
                {
                    "name": "event_update_message",
                    "value": "{EVENT.UPDATE.MESSAGE}"
                },
                {
                    "name": "host_name",
                    "value": "{HOST.NAME}"
```

```
                },
                {
                    "name": "trigger_description",
                    "value": "{TRIGGER.DESCRIPTION}"
                },
                {
                    "name": "trigger_id",
                    "value": "{TRIGGER.ID}"
                },
                {
                    "name": "alert_source",
                    "value": "Zabbix"
                }
            ]
        }
    ],
    "id": 1
}
```

Véase también

- Usuario

Fuente

CMediaType::get() en *ui/include/classes/api/services/CMediaType.php*.

mediatype.update

Descripción

`object mediatype.update(object/array mediaTypes)`

Este método permite actualizar tipos de medios existentes.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.
 Si se especifican métodos de entrega en el indicador de funcionalidad `$ZBX_FEATURE_FLAGS['media_type_denylist']` del archivo de configuración del frontend, el usuario no podrá llamar al método para los tipos de medios correspondientes.

Parámetros

`(object/array)` Propiedades del tipo de medio a actualizar.

La propiedad `mediatypeid` debe definirse para cada tipo de medio, todas las demás propiedades son opcionales. Solo se actualizarán las propiedades pasadas, todas las demás permanecerán sin cambios.

Además de las propiedades estándar del tipo de medio, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
message_templates	array	Plantillas de mensajes para reemplazar las plantillas de mensajes actuales.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los tipos de medios actualizados bajo la propiedad `mediatypeids`.

Ejemplos

Habilitar un tipo de medio

Habilite un tipo de medio, es decir, establezca su estado en "0".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "mediatype.update",
    "params": {
        "mediatypeid": "6",
        "status": "0"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "mediatypeids": [
            "6"
        ]
    },
    "id": 1
}
```

Origen

CMediaType::update() en *ui/include/classes/api/services/CMediaType.php*.

Token

Esta clase está diseñada para trabajar con tokens.

Referencias de objetos:

- Token

Métodos disponibles:

- token.create - crear nuevos tokens
- token.delete - eliminar tokens
- token.get - recuperar tokens
- token.update - actualizar tokens
- token.generate - generar tokens

Objeto token

Los siguientes objetos están directamente relacionados con la API `token`.

Token

El objeto token tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
tokenid	ID	ID del token. Comportamiento de la propiedad: - *solo lectura* - *requerido* para operaciones de actualización
name	string	Nombre del token. Comportamiento de la propiedad: - *requerido* para operaciones de creación
description	text	Descripción del token.

Propiedad	Tipo	Descripción
userid	ID	ID del usuario al que se ha asignado el token. Por defecto: *usuario actual*. Comportamiento de la propiedad: - *constante*
lastaccess	timestamp	Fecha y hora más reciente en que se autenticó el token. "0" si el token nunca se ha autenticado. Comportamiento de la propiedad: - *solo lectura*
status	integer	Estado del token. Valores posibles: 0 - *(por defecto)* token habilitado; 1 - token deshabilitado.
expires_at	timestamp	Fecha y hora de expiración del token. "0" para tokens que nunca expiran.
created_at	timestamp	Fecha y hora de creación del token. Comportamiento de la propiedad: - *solo lectura*
creator_userid	ID	ID del usuario que creó el token. Comportamiento de la propiedad: - *solo lectura*

token.create

Descripción

`object token.create(object/array tokens)`

Este método le permite crear nuevos tokens.

> **Note:**
> Se requiere el permiso *Manage API tokens* para que el rol de usuario pueda administrar tokens de otros usuarios.

> **Attention:**
> Un token creado por este método también debe ser generado antes de poder usarse.

Parámetros

`(objeto/array)` Tokens a crear.

El método acepta tokens con las propiedades estándar del token.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los tokens creados bajo la propiedad `tokenids`. El orden de los IDs devueltos coincide con el orden de los tokens enviados.

Ejemplos

Crear un token

Cree un token habilitado que nunca expire y autentique al usuario con ID 2.

Solicitud:

```
{
    "jsonrpc": "2.0",
```

```
    "method": "token.create",
    "params": {
        "name": "Su token",
        "userid": "2"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "tokenids": [
            "188"
        ]
    },
    "id": 1
}
```

Cree un token deshabilitado que expire el 21 de enero de 2021. Este token autenticará al usuario actual.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "token.create",
    "params": {
        "name": "Su token",
        "status": "1",
        "expires_at": "1611238072"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "tokenids": [
            "189"
        ]
    },
    "id": 1
}
```

Fuente

CToken::create() en *ui/include/classes/api/services/CToken.php*.

token.delete

Descripción

`object token.delete(array tokenids)`

Este método le permite eliminar tokens.

> **Note:**
> Se requiere el permiso *Manage API tokens* para que el rol de usuario pueda administrar tokens de otros usuarios.

Parámetros

`(array)` IDs de los tokens a eliminar.

Valores de retorno

(object) Devuelve un objeto que contiene los IDs de los tokens eliminados bajo la propiedad tokenids.

Ejemplos

Eliminar varios tokens

Eliminar dos tokens.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "token.delete",
    "params": [
        "188",
        "192"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "tokenids": [
            "188",
            "192"
        ]
    },
    "id": 1
}
```

Fuente

CToken::delete() en *ui/include/classes/api/services/CToken.php*.

token.generate

Descripción

`object token.generate(array tokenids)`

Este método le permite generar tokens.

Note:
Se requiere el permiso *Manage API tokens* para que el rol de usuario pueda administrar tokens de otros usuarios.

Attention:
Un token puede generarse mediante este método solo si ha sido creado.

Parámetros

(array) IDs de los tokens a generar.

Valores de retorno

(array) Devuelve un array de objetos que contiene el ID del token generado bajo la propiedad tokenid y la cadena de autorización generada bajo la propiedad token.

Propiedad	Tipo	Descripción
tokenid	ID	ID del token.
token	string	La cadena de autorización generada para este token.

Ejemplos

Generar varios tokens

Generar dos tokens.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "token.generate",
    "params": [
        "1",
        "2"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "tokenid": "1",
            "token": "bbcfce79a2d95037502f7e9a534906d3466c9a1484beb6ea0f4e7be28e8b8ce2"
        },
        {
            "tokenid": "2",
            "token": "fa1258a83d518eabd87698a96bd7f07e5a6ae8aeb8463cae33d50b91dd21bd6d"
        }
    ],
    "id": 1
}
```

Fuente

CToken::generate() en *ui/include/classes/api/services/CToken.php*.

token.get

Descripción

`integer/array token.get(object parameters)`

El método permite recuperar tokens según los parámetros proporcionados.

> **Note:**
> Solo el tipo de usuario *Super admin* tiene □□□□□□ para ver tokens de otros usuarios.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
tokenids	ID/array	Devolver solo los tokens con los IDs dados.
userids	ID/array	Devolver solo los tokens creados para los usuarios dados.
token	string	Devolver solo los tokens creados para el *Auth token* dado.
valid_at	timestamp	Devolver solo los tokens que sean válidos (no expirados) en la fecha y hora dadas.
expired_at	timestamp	Devolver solo los tokens que estén expirados (no válidos) en la fecha y hora dadas.
sortfield	string/array	Ordenar el resultado por las propiedades dadas. Valores posibles: `tokenid`, `name`, `lastaccess`, `status`, `expires_at`, `created_at`.
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
excludeSearch	boolean	

Parámetro	Tipo	Descripción
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

`(entero/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperar un token

Recuperar todos los datos del token con ID "2".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "token.get",
    "params": {
        "output": "extend",
        "tokenids": "2"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "tokenid": "1",
            "name": "The Token",
            "description": "",
            "userid": "1",
            "lastaccess": "0",
            "status": "0",
            "expires_at": "1609406220",
            "created_at": "1611239454",
            "creator_userid": "1"
        }
    ],
    "id": 1
}
```

Fuente

CToken::get() en *ui/include/classes/api/services/CToken.php*.

token.update

Descripción

`object token.update(object/array tokens)`

Este método permite actualizar tokens existentes.

Note:
Se requiere el permiso *Manage API tokens* para que el rol de usuario pueda administrar tokens de otros usuarios.

Parámetros

`(objeto/array)` Propiedades del token que se van a actualizar.

La propiedad `tokenid` debe estar definida para cada token, todas las demás propiedades son opcionales. Sólo se actualizarán las propiedades indicadas, todas las demás permanecerán sin cambios.

El método acepta tokens con las propiedades estándar del token.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los tokens actualizados bajo la propiedad `tokenids`.

Ejemplos

Eliminar la expiración del token

Eliminar la fecha de expiración del token.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "token.update",
    "params": {
        "tokenid": "2",
        "expires_at": "0"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "tokenids": [
            "2"
        ]
    },
    "id": 1
}
```

Fuente

CToken::update() en *ui/include/classes/api/services/CToken.php*.

Traza de APM

Esta clase está diseñada para trabajar con trazas recopiladas para la monitorización del rendimiento de aplicaciones (APM).

Referencias de objetos:

- Traza de APM
- Evento
- Enlace

Métodos disponibles:

- apmtrace.get - recuperar trazas

Objeto de traza de APM

Los siguientes objetos están directamente relacionados con la API `apmtrace`.

Note:
Cada propiedad, excepto `span_count` y `error_count`, corresponde a una columna de la tabla `otel_traces` del origen de datos de telemetría. La mayoría de las propiedades corresponden a la columna con el mismo nombre (por ejemplo, la propiedad `span_name` corresponde a la columna `SpanName`). Cuando los nombres difieren, las columnas del origen de datos se especifican en las tablas siguientes. Para consultar la lista de columnas, consulte Columnas de datos de telemetría compatibles.

Traza de APM

El objeto de traza de APM representa una traza (la ruta que sigue una solicitud a través de sus aplicaciones y servicios). Una traza consta de uno o más spans; cada span representa una única operación.

El objeto de traza de APM se basa en el span raíz de la traza (el span que inicia la traza y no tiene un span principal). Todas las propiedades contienen datos del span raíz, excepto `span_count` y `error_count`, que se calculan a partir de todos los spans de la traza.

El objeto de traza de APM tiene las siguientes propiedades:

Propiedad	Tipo	Descripción
timestamp	integer	Hora Unix (en nanosegundos) en la que se inició el span raíz. Comportamiento de la propiedad: - *solo lectura*
traceid	string	ID de la traza, compartido por todos los spans que pertenecen a la traza. Comportamiento de la propiedad: - *solo lectura*
spanid	string	ID del span raíz. Comportamiento de la propiedad: - *solo lectura*
trace_state	string	Datos adicionales, si los hay, que las herramientas de trazado adjuntan a la traza (consulte W3C Trace Context). Comportamiento de la propiedad: - *solo lectura*
span_name	string	Nombre de la operación que representa el span raíz (por ejemplo, `POST /zabbix.php`). Comportamiento de la propiedad: - *solo lectura*
span_kind	string	Rol del span raíz. Valores posibles: `Unspecified`: el tipo de span no está especificado; `Internal`: operación realizada dentro de una aplicación; `Server`: gestión de una solicitud entrante; `Client`: solicitud saliente; `Producer`: creación de una operación asíncrona (por ejemplo, el envío de un mensaje); `Consumer`: procesamiento de una operación asíncrona. Comportamiento de la propiedad: - *solo lectura*
service_name	string	Nombre del servicio que produjo el span raíz (por ejemplo, `zabbix-nginx`), obtenido del atributo de recurso `service.name` de OpenTelemetry. Comportamiento de la propiedad: - *solo lectura*

Propiedad	Tipo	Descripción
resource_attributes	object	Atributos de recurso clave-valor del span raíz (por ejemplo, `host.name: eabf8c2a5ddc`). Comportamiento de la propiedad: - *solo lectura*
scope_name	string	Nombre del ámbito de instrumentación que produjo el span raíz. Comportamiento de la propiedad: - *solo lectura*
scope_version	string	Versión del ámbito de instrumentación. Comportamiento de la propiedad: - *solo lectura*
span_attributes	object	Atributos del span clave-valor del span raíz (por ejemplo, `http.method: POST`). Comportamiento de la propiedad: - *solo lectura*
duration	integer	Nanosegundos de duración de la traza. Comportamiento de la propiedad: - *solo lectura*
status_code	string	Estado de la operación que representa el span raíz. Valores posibles: `Unset`: no se ha asignado ningún estado explícito al span raíz; `Ok`: la operación se completó correctamente; `Error`: la operación se completó con un error. Comportamiento de la propiedad: - *solo lectura*
status_message	string	Información adicional, si la hay, sobre el estado del span raíz. Comportamiento de la propiedad: - *solo lectura*
events	array	Matriz de objetos event del span raíz, es decir, momentos relevantes durante la operación (por ejemplo, excepciones). Comportamiento de la propiedad: - *solo lectura*
links	array	Matriz de objetos link del span raíz, que apuntan a spans relacionados en la misma traza o en otra. Comportamiento de la propiedad: - *solo lectura*
span_count	integer	Número total de spans de la traza. Comportamiento de la propiedad: - *solo lectura*
error_count	integer	Número de spans de la traza cuyo `status_code` está establecido en `Error`. Comportamiento de la propiedad: - *solo lectura*

Evento

El objeto de evento tiene las siguientes propiedades.

Property	Type	Description
timestamp	integer	Hora Unix (en nanosegundos) en la que ocurrió el evento. Columna de origen de datos: `Events.Timestamp`
name	string	Nombre del evento. Columna de origen de datos: `Events.Name`
attributes	object	Atributos clave-valor con información adicional, si la hay, sobre el evento. Columna de origen de datos: `Events.Attributes`

Enlace

El objeto de enlace tiene las siguientes propiedades.

Property	Type	Description
traceid	string	ID del trace que contiene el span enlazado. Columna de origen de datos: `Links.TraceId`
spanid	string	ID del span enlazado. Junto con `traceid`, identifica de forma única el span enlazado. Columna de origen de datos: `Links.SpanId`
trace_state	string	Estado del trace, si existe, del span enlazado (consulte W3C Trace Context). Columna de origen de datos: `Links.TraceState`
attributes	object	Atributos clave-valor con información adicional sobre el enlace. Columna de origen de datos: `Links.Attributes`

apmtrace.get

Descripción

`integer/array apmtrace.get(object parameters)`

El método permite recuperar trazas según los parámetros especificados.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(object)` Parámetros que definen el resultado deseado.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
time_from	timestamp	Devuelve únicamente los seguimientos cuyo span raíz se inició en el momento indicado o después. Comportamiento del parámetro: - *obligatorio*
time_till	timestamp	Devuelve únicamente los seguimientos cuyo span raíz se inició antes del momento indicado. Comportamiento del parámetro: - *obligatorio*
traceids	string/array	Devuelve únicamente los seguimientos con los ID indicados.

Parámetro	Tipo	Descripción
resource_attributes	object/array	Devuelve únicamente los seguimientos cuyo span raíz tiene los atributos de recurso indicados. Formato: `[{"key": "<key>", "value": "<value>", "operator": "<operator>"}, ...]`. Un array vacío devuelve todos los seguimientos. Valores posibles de operator: 0 - *(predeterminado)* Contiene; 1 - Es igual a; 2 - No contiene; 3 - No es igual a; 4 - Existe; 5 - No existe.
resource_attributes_evaltype	integer	Método de evaluación de las condiciones de `resource_attributes`. Valores posibles: 0 - *(predeterminado)* And/Or; 2 - Or.
span_attributes	object/array	Devuelve únicamente los seguimientos cuyo span raíz tiene los atributos de span indicados. Formato: `[{"key": "<key>", "value": "<value>", "operator": "<operator>"}, ...]`. Un array vacío devuelve todos los seguimientos. Valores posibles de operator: 0 - *(predeterminado)* Contiene; 1 - Es igual a; 2 - No contiene; 3 - No es igual a; 4 - Existe; 5 - No existe.
span_attributes_evaltype	integer	Método de evaluación de las condiciones de `span_attributes`. Valores posibles: 0 - *(predeterminado)* And/Or; 2 - Or.
min_duration	integer	Devuelve únicamente los seguimientos cuya `duration` del span raíz es mayor o igual que el valor indicado, en nanosegundos.
max_duration	integer	Devuelve únicamente los seguimientos cuya `duration` del span raíz es menor que el valor indicado, en nanosegundos.
filter	object	Devuelve únicamente los resultados que coinciden exactamente con el filtro indicado. Acepta un objeto cuyas claves son nombres de propiedades y cuyos valores son un único valor o un array de valores con los que se debe comparar. Propiedades compatibles: `traceid`, `spanid`, `trace_state`, `span_name`, `span_kind`, `service_name`, `scope_name`, `scope_version`, `status_code`, `status_message`.
search	object	Devuelve los resultados que coinciden con el patrón indicado (sin distinguir entre mayúsculas y minúsculas). Acepta un objeto cuyas claves son nombres de propiedades y cuyos valores son cadenas que se deben buscar. Si no se proporcionan opciones adicionales, se realizará una búsqueda `LIKE "%…%"`. Propiedades compatibles: `trace_state`, `span_name`, `service_name`, `scope_name`, `scope_version`, `status_message`.

Parámetro	Tipo	Descripción
sortfield	string/array	Ordena el resultado según las propiedades indicadas. Valores posibles: `timestamp`, `traceid`, `spanid`, `trace_state`, `span_name`, `span_kind`, `service_name`, `scope_name`, `scope_version`, `duration`, `status_code`, `status_message`.
offset	integer	Número de registros que se deben omitir antes de devolver el resultado.
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
excludeSearch	boolean	
limit	integer	
output	query	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores devueltos

`(integer/array)` Devuelve uno de los siguientes valores:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperación de trazas lentas

Recupere las dos trazas más lentas del servicio `zabbix-nginx`. Busque trazas desde las 10:00 hasta las 11:00 del 1 de octubre de 2026 (UTC) e incluya únicamente aquellas que tardaron 1 segundo o más.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "apmtrace.get",
    "params": {
        "output": ["timestamp", "traceid", "span_name", "service_name", "duration", "status_code", "span_c
        "time_from": 1790848800,
        "time_till": 1790852400,
        "filter": {
            "service_name": "zabbix-nginx"
        },
        "min_duration": 1000000000,
        "sortfield": "duration",
        "sortorder": "DESC",
        "limit": 2
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "timestamp": "1790849274910226417",
            "traceid": "8a4f1c7e2b9d4e6fa0c3b5d7e9f1a2c4",
            "span_name": "POST /zabbix.php",
            "service_name": "zabbix-nginx",
            "duration": "30412907815",
            "status_code": "Error",
            "span_count": "49",
            "error_count": "2"
        },
```

```
        {
            "timestamp": "1790850512338104952",
            "traceid": "dee625f33cb0994e86c5cd4640e3b09c",
            "span_name": "POST /zabbix.php",
            "service_name": "zabbix-nginx",
            "duration": "1104826531",
            "status_code": "Unset",
            "span_count": "27",
            "error_count": "0"
        }
    ],
    "id": 1
}
```


Véase también

- APM span
- APM metric
- APM log

Origen

CApmTrace::get() en *ui/include/classes/api/services/CApmTrace.php*.

Trend

Esta clase está diseñada para trabajar con datos de tendencias.

Referencias de objetos:

- Tendencia de tipo float
- Tendencia de tipo integer

Métodos disponibles:

- trend.get - recuperar tendencias

Objeto de tendencia

Los siguientes objetos están directamente relacionados con la API de `trend`.

> **Note:**
> Los objetos de tendencia difieren según el tipo de información de la métrica. Son creados por el servidor Zabbix y no pueden ser modificados a través de la API.

Tendencia de tipo float

El objeto de tendencia de tipo float tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
clock	timestamp	Marca de tiempo de la hora para la que se calculó el valor. Por ejemplo, la marca de tiempo "04:00:00" significa valores calculados para el período "04:00:00-04:59:59".
itemid	ID	ID del item relacionado.
num	integer	Número de valores que estuvieron disponibles para la hora.
value_min	float	Valor mínimo por hora.
value_avg	float	Valor promedio por hora.
value_max	float	Valor máximo por hora.

Tendencia de enteros

El objeto de tendencia de enteros tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
clock	timestamp	Marca de tiempo de una hora para la que se calculó el valor. Por ejemplo, la marca de tiempo ”04:00:00” significa valores calculados para el período ”04:00:00-04:59:59”.
itemid	ID	ID del item relacionado.
num	integer	Número de valores que estaban disponibles para la hora.
value_min	integer	Valor mínimo por hora.
value_avg	integer	Valor promedio por hora.
value_max	integer	Valor máximo por hora.

trend.get

Descripción

`integer/array trend.get(object parameters)`

El método permite recuperar datos de tendencias según los parámetros proporcionados.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo.
> Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario.
> Consulte Roles de usuario para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
itemids	ID/array	Devolver solo las tendencias con los IDs de item dados.
time_from	timestamp	Devolver solo los valores que se hayan recogido después o en el momento dado.
time_till	timestamp	Devolver solo los valores que se hayan recogido antes o en el momento dado.
countOutput	boolean	Contar el número de objetos recuperados.
limit	integer	Limitar la cantidad de objetos recuperados.
output	query	Establecer las propiedades del objeto Trend que se devolverán.

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperando datos de tendencias de métricas

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "trend.get",
    "params": {
        "output": [
            "itemid",
            "clock",
            "num",
            "value_min",
            "value_avg",
            "value_max",
        ],
```

```
        "itemids": [
            "23715"
        ],
        "limit": "1"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "itemid": "23715",
            "clock": "1446199200",
            "num": "60",
            "value_min": "0.165",
            "value_avg": "0.2168",
            "value_max": "0.35",
        }
    ],
    "id": 1
}
```

Fuente

CTrend::get() en *ui/include/classes/api/services/CTrend.php*.

Trigger

Esta clase está diseñada para trabajar con triggers.

Referencias de objetos:

- Trigger
- Etiqueta de trigger

Métodos disponibles:

- trigger.create - crear nuevos triggers
- trigger.delete - eliminar triggers
- trigger.get - recuperar triggers
- trigger.update - actualizar triggers

Objeto trigger

Los siguientes objetos están directamente relacionados con la API de `trigger`.

Trigger

El objeto trigger tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
triggerid	ID	ID del trigger. Comportamiento de la propiedad: - *solo lectura* - *requerido* para operaciones de actualización
description	string	Nombre del trigger. Comportamiento de la propiedad: - *requerido* para operaciones de creación

Propiedad	Tipo	Descripción
expression	string	Expresión del trigger. Los items con `value_type` establecido en 5 (binario) o 6 (JSON) no pueden usarse en expresiones de trigger. Comportamiento de la propiedad: - *requerido* para operaciones de creación
event_name	string	Nombre del evento generado por el trigger.
opdata	string	Datos operativos.
comments	string	Descripción adicional del trigger.
error	string	Texto de error si ha habido algún problema al actualizar el estado del trigger. Comportamiento de la propiedad: - *solo lectura*
flags	integer	Origen del trigger. Valores posibles: 0 - *(por defecto)* un trigger simple; 4 - un trigger convertido desde prototipo. Comportamiento de la propiedad: - *solo lectura*
lastchange	timestamp	Hora en la que el trigger cambió por última vez su estado. Comportamiento de la propiedad: - *solo lectura*
priority	integer	Severidad del trigger. Valores posibles: 0 - *(por defecto)* no clasificado; 1 - información; 2 - advertencia; 3 - media; 4 - alta; 5 - desastre.
state	integer	Estado de la expresión del trigger. Valores posibles: 0 - *(por defecto)* el estado del trigger está actualizado; 1 - el estado actual del trigger es desconocido. Comportamiento de la propiedad: - *solo lectura*
status	integer	Indica si el trigger está habilitado o deshabilitado. Valores posibles: 0 - *(por defecto)* habilitado; 1 - deshabilitado.
templateid	ID	ID del trigger de la template padre. Comportamiento de la propiedad: - *solo lectura*
type	integer	Indica si el trigger puede generar múltiples eventos de problema. Valores posibles: 0 - *(por defecto)* no generar múltiples eventos; 1 - generar múltiples eventos.
url	string	URL asociada al trigger.
url_name	string	Etiqueta para la URL asociada al trigger.

Propiedad	Tipo	Descripción
value	integer	Indica si el trigger está en estado OK o de problema. Valores posibles: 0 - *(por defecto)* OK; 1 - problema. Comportamiento de la propiedad: - *solo lectura*
recovery_mode	integer	Modo de generación de eventos OK. Valores posibles: 0 - *(por defecto)* Expresión; 1 - Expresión de recuperación; 2 - Ninguno.
recovery_expression	string	Expresión de recuperación del trigger. Los items con `value_type` establecido en 5 (binario) o 6 (JSON) no pueden usarse en expresiones de recuperación de trigger.
correlation_mode	integer	Cierre de eventos OK. Valores posibles: 0 - *(por defecto)* Todos los problemas; 1 - Todos los problemas si los valores de las etiquetas coinciden.
correlation_tag	string	Etiqueta para coincidencia.
manual_close	integer	Permitir cierre manual. Valores posibles: 0 - *(por defecto)* No; 1 - Sí.
uuid	string	Identificador único universal, utilizado para vincular triggers importados con los ya existentes. Se genera automáticamente si no se proporciona. Comportamiento de la propiedad: - *soportado* si el trigger pertenece a una template

Etiqueta de trigger

El objeto de etiqueta de trigger tiene las siguientes propiedades.

Propiedad	Tipo	Descripción
tag	string	Nombre de la etiqueta de trigger. Comportamiento de la propiedad: - *obligatorio*
value	string	Valor de la etiqueta de trigger. Comportamiento de la propiedad: - *solo lectura*.
object	integer	Tipo de objeto del que se heredó la etiqueta. Valores posibles: 0 - template; 1 - host; 3 - item. Comportamiento de la propiedad: - *solo lectura*.

Propiedad	Tipo	Descripción
objectid	ID	ID del objeto del que se heredó la etiqueta. Comportamiento de la propiedad: - *solo lectura*.
automatic	integer	Tipo de etiqueta de trigger: Valores posibles: 0 - *(por defecto)* etiqueta creada por el usuario; 1 - etiqueta creada por LLD;Comportamiento de la propiedad: - *solo lectura*.

trigger.create

Descripción

`object trigger.create(object/array triggers)`

Este método permite crear nuevos triggers.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(objeto/array)` Triggers a crear.

Además de las propiedades estándar de trigger, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
dependencies	array	Triggers de los que depende el trigger. Los triggers deben tener definida únicamente la propiedad `triggerid`.
tags	array	Etiquetas de trigger.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los disparadores creados bajo la propiedad `triggerids`. El orden de los IDs devueltos coincide con el orden de los disparadores enviados.

Ejemplos

Crear un trigger

Cree dos triggers, cada uno dependiendo de otro trigger.

Request:

```
{
    "jsonrpc": "2.0",
    "method": "trigger.create",
    "params": [
        {
            "description": "Processor load is too high on {HOST.NAME}",
            "expression": "last(/Linux server/system.cpu.load[percpu,avg1])>5",
            "dependencies": [
                {
                    "triggerid": "17367"
                }
            ]
        },
        {
```

```
            "description": "Service has stopped",
            "expression": "length(last(/Linux server/log[/var/log/system,Service .* has stopped]))<>0",
            "dependencies": [
                {
                    "triggerid": "17368"
                }
            ],
            "tags": [
                {
                    "tag": "service",
                    "value": "{{ITEM.VALUE}.regsub(\"Service (.*) has stopped\", \"\\1\")}"
                },
                {
                    "tag": "error",
                    "value": ""
                }
            ]
        }
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "triggerids": [
            "17369",
            "17370"
        ]
    },
    "id": 1
}
```

Fuente

CTrigger::create() en *ui/include/classes/api/services/CTrigger.php*.

trigger.delete

Descripción

`object trigger.delete(array triggerIds)`

Este método le permite eliminar triggers.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

`(array)` IDs de los disparadores a eliminar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los disparadores eliminados bajo la propiedad `triggerids`.

Ejemplos

Eliminar múltiples disparadores

Eliminar dos disparadores.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "trigger.delete",
    "params": [
        "12002",
        "12003"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "triggerids": [
            "12002",
            "12003"
        ]
    },
    "id": 1
}
```

Fuente

CTrigger::delete() en *ui/include/classes/api/services/CTrigger.php*.

trigger.get

Descripción

`integer/array trigger.get(object parameters)`

El método permite recuperar triggers según los parámetros proporcionados.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object)` Parámetros que definen el resultado deseado.

El método admite los siguientes parámetros.

Parámetro	Tipo	Descripción
triggerids	ID/array	Devuelve solo los triggers con los ID indicados.
groupids	ID/array	Devuelve solo los triggers que pertenecen a hosts o templates de los grupos de hosts o grupos de templates indicados.
templateids	ID/array	Devuelve solo los triggers que pertenecen a los templates indicados.
hostids	ID/array	Devuelve solo los triggers que pertenecen a los hosts indicados.
itemids	ID/array	Devuelve solo los triggers que contienen los items indicados.
maintenanceids	ID/array	Devuelve solo los triggers afectados por los mantenimientos indicados.
functions	string/array	Devuelve solo los triggers que utilizan las funciones indicadas. Consulte la página de funciones compatibles para obtener una lista de las funciones compatibles.
group	string	Devuelve solo los triggers que pertenecen a hosts o templates del grupo de hosts o grupo de templates con el nombre indicado.
host	string	Devuelve solo los triggers que pertenecen al host con el nombre técnico indicado.
inherited	boolean	Si se establece en `true`, devuelve solo los triggers heredados de un template.
templated	boolean	Si se establece en `true`, devuelve solo los triggers que pertenecen a templates.

Parámetro	Tipo	Descripción
dependent	boolean	Si se establece en `true`, devuelve solo los triggers que tienen dependencias. Si se establece en `false`, devuelve solo los triggers que no tienen dependencias.
monitored	flag	Devuelve solo los triggers habilitados que pertenecen a hosts monitorizados y contienen únicamente items habilitados.
active	flag	Devuelve solo los triggers habilitados que pertenecen a hosts monitorizados.
maintenance	boolean	Si se establece en `true`, devuelve solo los triggers habilitados que pertenecen a hosts en mantenimiento.
withUnacknowledgedEvents	flag	Devuelve solo los triggers que tienen eventos no reconocidos.
withAcknowledgedEvents	flag	Devuelve solo los triggers cuyos eventos han sido reconocidos.
withLastEventUnacknowledged	flag	Devuelve solo los triggers cuyo último evento no ha sido reconocido.
skipDependent	flag	Omite los triggers en estado de problema que dependen de otros triggers. Tenga en cuenta que los otros triggers se ignoran si están deshabilitados, tienen items deshabilitados o pertenecen a hosts con items deshabilitados.
lastChangeSince	timestamp	Devuelve solo los triggers cuyo estado haya cambiado después de la hora indicada.
lastChangeTill	timestamp	Devuelve solo los triggers cuyo estado haya cambiado antes de la hora indicada.
only_true	flag	Devuelve solo los triggers que hayan estado recientemente en estado de problema (dentro de `ok_period`).
min_severity	integer	Devuelve solo los triggers cuya gravedad sea mayor o igual que la gravedad indicada.
evaltype	integer	Método de evaluación de las etiquetas. Valores posibles: 0 - *(predeterminado)* And/Or; 2 - Or.
tags	array	Devuelve solo los triggers con las etiquetas indicadas. Formato: `[{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]`. Una matriz vacía devuelve todos los triggers. Valores posibles de operator: 0 - *(predeterminado)* Contiene; 1 - Es igual a; 2 - No contiene; 3 - No es igual a; 4 - Existe; 5 - No existe.
expandComment	flag	Expande las macros en la descripción del trigger.
expandDescription	flag	Expande las macros en el nombre del trigger.
expandExpression	flag	Expande las funciones y macros en la expresión del trigger.
selectHostGroups	query	Devuelve los grupos de hosts a los que pertenece el trigger en la propiedad `hostgroups`.
selectHosts	query	Devuelve los hosts a los que pertenece el trigger en la propiedad `hosts`.
selectItems	query	Devuelve los items contenidos por el trigger en la propiedad `items`.
selectFunctions	query	Devuelve las funciones utilizadas en el trigger en la propiedad `functions`. Los objetos de función representan las funciones utilizadas en la expresión del trigger y tienen las siguientes propiedades: `functionid` - `(ID)` ID de la función; `itemid` - `(ID)` ID del item utilizado en la función; `function` - `(string)` nombre de la función; `parameter` - `(string)` parámetro pasado a la función. El parámetro de consulta se reemplaza por el símbolo $ en la cadena devuelta.
selectDependencies	query	Devuelve los triggers de los que depende el trigger en la propiedad `dependencies`.

Parámetro	Tipo	Descripción
selectDiscoveryData	query	Devuelve una propiedad `discoveryData` con los datos del objeto de descubrimiento del trigger. El objeto de descubrimiento del trigger vincula un trigger descubierto con un prototipo de trigger a partir del cual se descubrió. Tiene las siguientes propiedades: `parent_triggerid` - `(ID)` ID del prototipo de trigger a partir del cual se creó el trigger; `status` - `(int)` estado de descubrimiento del trigger: 0 - *(predeterminado)* el trigger se ha descubierto, 1 - el trigger ya no se ha descubierto; `ts_delete` - `(timestamp)` momento en que se eliminará un trigger que ya no se ha descubierto; `ts_disable` - `(timestamp)` momento en que se deshabilitará un trigger que ya no se ha descubierto; `disable_source` - `(int)` indicador de si el trigger fue deshabilitado por una regla de LLD o manualmente: 0 - *(predeterminado)* deshabilitado automáticamente, 1 - deshabilitado por una regla de LLD.
selectDiscoveryRule	query	Devuelve la regla de descubrimiento de bajo nivel que creó el trigger en la propiedad `discoveryRule`.
selectLastEvent	query	Devuelve el último evento significativo del trigger en la propiedad `lastEvent`.
selectTags	query	Devuelve las etiquetas del trigger en la propiedad `tags`.
inheritedTags	boolean	Devuelve los triggers que también tienen las `tags` indicadas en el template/host/templates vinculados. Valores posibles: `true` - el template/host/templates vinculados también deben tener las etiquetas indicadas; `false` - *(predeterminado)* se ignoran las etiquetas del template/host/templates vinculados.
selectInheritedTags	query	Devuelve una propiedad `inheritedTags` con las etiquetas heredadas de templates, templates vinculados y hosts, así como de los items referenciados en las expresiones del trigger o de recuperación.
selectTemplateGroups	query	Devuelve los grupos de templates a los que pertenece el trigger en la propiedad `templategroups`.
filter	object	Devuelve solo los resultados que coinciden exactamente con el filtro indicado. Acepta un objeto cuyas claves son nombres de propiedades y cuyos valores son un único valor o una matriz de valores con los que se debe comparar. No admite propiedades del tipo de datos `text`. Admite propiedades adicionales: `host` - nombre técnico del host al que pertenece el trigger; `hostid` - ID del host al que pertenece el trigger.
limitSelects	integer	Limita el número de registros devueltos por las subselecciones. Se aplica a las siguientes subselecciones: `selectHosts` - los resultados se ordenarán por `host`.
sortfield	string/array	Ordena el resultado por las propiedades indicadas. Valores posibles: `triggerid`, `description`, `status`, `priority`, `lastchange`, `hostname`.
countOutput	boolean	Estos parámetros se describen en los comentarios de referencia.
editable	boolean	
excludeSearch	boolean	
limit	integer	

Parámetro	Tipo	Descripción
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	
selectTriggerDiscovery	query	Devuelve el objeto de descubrimiento del trigger en la propiedad `triggerDiscovery`. Los objetos de descubrimiento del trigger vinculan el trigger con un prototipo de trigger a partir del cual se creó. Esta consulta está **obsoleta**; utilice `selectDiscoveryData` en su lugar.

Valores de retorno

`(integer/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperar datos por ID de trigger

Recuperar todos los datos y las funciones utilizadas en el trigger "14062".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "trigger.get",
    "params": {
        "triggerids": "14062",
        "output": "extend",
        "selectFunctions": "extend"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "triggerid": "14062",
            "expression": "{13513}<10m",
            "description": "{HOST.NAME} has been restarted (uptime < 10m)",
            "url": "",
            "status": "0",
            "value": "0",
            "priority": "2",
            "lastchange": "0",
            "comments": "The host uptime is less than 10 minutes",
            "error": "",
            "templateid": "10016",
            "type": "0",
            "state": "0",
            "flags": "0",
            "recovery_mode": "0",
            "recovery_expression": "",
            "correlation_mode": "0",
            "correlation_tag": "",
            "manual_close": "0",
```

```
            "opdata": "",
            "event_name": "",
            "uuid": "",
            "url_name": "",
            "functions": [
                {
                    "functionid": "13513",
                    "itemid": "24350",
                    "triggerid": "14062",
                    "parameter": "$",
                    "function": "last"
                }
            ]
        }
    ],
    "id": 1
}
```

Recuperar disparadores en estado de problema

Recuperar el ID, nombre y gravedad de todos los disparadores en estado de problema y ordenarlos por gravedad en orden descendente.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "trigger.get",
    "params": {
        "output": [
            "triggerid",
            "description",
            "priority"
        ],
        "filter": {
            "value": 1
        },
        "sortfield": "priority",
        "sortorder": "DESC"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "triggerid": "13907",
            "description": "Zabbix self-monitoring processes < 100% busy",
            "priority": "4"
        },
        {
            "triggerid": "13824",
            "description": "Zabbix discoverer processes more than 75% busy",
            "priority": "3"
        }
    ],
    "id": 1
}
```

Recuperar un trigger específico con etiquetas

Recuperar un trigger específico con etiquetas.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "trigger.get",
    "params": {
        "output": [
            "triggerid",
            "description"
        ],
        "selectTags": "extend",
        "triggerids": [
            "17578"
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "triggerid": "17370",
            "description": "Service status",
            "tags": [
                {
                    "tag": "service",
                    "value": "{{ITEM.VALUE}.regsub(\"Service (.*) has stopped\", \"\\1\")}",
                    "automatic": "1"
                },
                {
                    "tag": "error",
                    "value": "",
                    "automatic": "1"
                }
            ]
        }
    ],
    "id": 1
}
```

Ver también

- Regla de descubrimiento
- Item
- Host
- Grupo de hosts
- Grupo de templates

Fuente

CTrigger::get() en *ui/include/classes/api/services/CTrigger.php*.

trigger.update

Descripción

`object trigger.update(object/array triggers)`

Este método permite actualizar triggers existentes.

> **Note:**
> Este método solo está disponible para los tipos de usuario *Admin* y *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte Roles de usuario para obtener más información.

Parámetros

(objeto/array) Propiedades del trigger que se actualizarán.

La propiedad triggerid debe definirse para cada trigger, todas las demás propiedades son opcionales. Solo se actualizarán las propiedades pasadas, todas las demás permanecerán sin cambios.

Además de las propiedades estándar del trigger, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
dependencies	array	Triggers de los que depende el trigger. Los triggers deben tener definida solo la propiedad triggerid.
tags	array	Etiquetas de trigger.

Valores de retorno

(object) Devuelve un objeto que contiene los IDs de los disparadores actualizados bajo la propiedad triggerids.

Ejemplos

Habilitar un disparador

Habilite un disparador, es decir, establezca su estado en "0".

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "trigger.update",
    "params": {
        "triggerid": "13938",
        "status": 0
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "triggerids": [
            "13938"
        ]
    },
    "id": 1
}
```

Reemplazar etiquetas de disparadores

Reemplazar etiquetas para el disparador.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "trigger.update",
    "params": {
        "triggerid": "13938",
        "tags": [
            {
                "tag": "service",
                "value": "{{ITEM.VALUE}.regsub(\"Service (.*) has stopped\", \"\\1\")}"
            },
            {
                "tag": "error",
                "value": ""
            }
```

```
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "triggerids": [
            "13938"
        ]
    },
    "id": 1
}
```

Reemplazando dependencias

Reemplazar dependencias para el disparador.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "trigger.update",
    "params": {
        "triggerid": "22713",
        "dependencies": [
            {
                "triggerid": "22712"
            },
            {
                "triggerid": "22772"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "triggerids": [
            "22713"
        ]
    },
    "id": 1
}
```

Fuente

CTrigger::update() en *ui/include/classes/api/services/CTrigger.php*.

Usuario

Esta clase está diseñada para trabajar con usuarios.

Referencias de objetos:

- Usuario
- Medio

Métodos disponibles:

- user.checkauthentication - comprobar y prolongar sesiones de usuario

- user.create - crear nuevos usuarios
- user.delete - eliminar usuarios
- user.get - obtener usuarios
- user.login - iniciar sesión en la API
- user.logout - cerrar sesión en la API
- user.provision - aprovisionar usuarios LDAP
- user.resettotp - restablecer secretos TOTP de usuario
- user.unblock - desbloquear usuarios
- user.update - actualizar usuarios

Objeto usuario

Los siguientes objetos están directamente relacionados con la API de `usuario`.

Usuario

El objeto de usuario tiene las siguientes propiedades.

Property	Type	Description
userid	ID	ID del usuario. Comportamiento de la propiedad: - *solo lectura* - *obligatoria* para las operaciones de actualización
username	string	Nombre del usuario. Comportamiento de la propiedad: - *obligatoria* para las operaciones de creación - *solo lectura* para usuarios aprovisionados si el usuario está vinculado a un directorio de usuarios (`userdirectoryid` está establecido en un valor válido distinto de ”0”), el estado de aprovisionamiento del directorio de usuarios está habilitado (`provision_status` del objeto de directorio de usuarios está establecido en ”1”) y el estado de autenticación de todo el aprovisionamiento LDAP o SAML está habilitado (`ldap_jit_status` del objeto de autenticación está establecido en ”Enabled for configured LDAP IdPs” o `saml_jit_status` del objeto de autenticación está establecido en ”Enabled for configured SAML IdPs”)
passwd	string	Contraseña del usuario. El valor de este parámetro puede ser una cadena vacía si el usuario está vinculado a un directorio de usuarios. Comportamiento de la propiedad: - *solo escritura*
roleid	ID	ID del rol del usuario. Tenga en cuenta que los usuarios sin un rol solo pueden iniciar sesión en Zabbix mediante la autenticación LDAP o SAML, siempre que su información LDAP/SAML coincida con las asignaciones de grupos de usuarios configuradas en Zabbix.
attempt_clock	timestamp	Hora del último intento de inicio de sesión fallido. Comportamiento de la propiedad: - *solo lectura*
attempt_failed	integer	Cantidad de intentos de inicio de sesión fallidos recientes. Comportamiento de la propiedad: - *solo lectura*

Property	Type	Description
attempt_ip	string	Dirección IP desde la que se realizó el último intento de inicio de sesión fallido. Comportamiento de la propiedad: - *solo lectura*
autologin	integer	Indica si se debe habilitar el inicio de sesión automático. Valores posibles: 0 - *(predeterminado)* inicio de sesión automático deshabilitado; 1 - inicio de sesión automático habilitado.
autologout	string	Tiempo de vida de la sesión del usuario. Acepta un número de segundos (por ejemplo, 1, 60, 300) o un valor con un sufijo de tiempo (por ejemplo, 30s, 1m, 2h). Si se establece en 0s, la sesión nunca caducará. Predeterminado: 15m.
default_maintenance_period	string	Período de mantenimiento predeterminado para un mantenimiento creado mediante un acceso directo. Acepta segundos o una unidad de tiempo con sufijo (300s, 5m, 1h, 1d, 1w). El valor indicado se redondeará hacia abajo a minutos completos. Predeterminado: 1h.
lang	string	Código de idioma del usuario, por ejemplo, en_US. Predeterminado: default - valor predeterminado del sistema.
name	string	Nombre del usuario.
provisioned	integer	Indica si el usuario ha sido aprovisionado. Valores posibles: 0 - no aprovisionado; 1 - aprovisionado. Comportamiento de la propiedad: - *solo lectura*
refresh	string	Período de actualización automática. Acepta un número de segundos (por ejemplo, 1, 60, 300) o un valor con un sufijo de tiempo (por ejemplo, 30s, 1m, 1h). Predeterminado: 30s.
rows_per_page	integer	Cantidad de filas de objetos que se mostrarán por página. Predeterminado: 50.
surname	string	Apellido del usuario.
theme	string	Tema del usuario. Valores posibles: default - *(predeterminado)* valor predeterminado del sistema; blue-theme - Azul; blue-classic-theme - Azul (clásico); dark-theme - Oscuro; dark-blue-theme - Azul oscuro; dark-classic-theme - Oscuro (clásico); hc-light - Claro de alto contraste; hc-dark - Oscuro de alto contraste.
ts_provisioned	timestamp	Hora en la que se realizó la última operación de aprovisionamiento. Comportamiento de la propiedad: - *solo lectura* - *compatible* con usuarios de tipo *Super admin*
url	string	URL de la página a la que se redirigirá al usuario después de iniciar sesión.

Property	Type	Description
userdirectoryid	ID	ID del directorio de usuarios al que está vinculado el usuario. Se utiliza para el aprovisionamiento (creación o actualización), así como para iniciar sesión con un usuario vinculado a un directorio de usuarios. Para las operaciones de inicio de sesión, el valor de esta propiedad tendrá prioridad sobre la propiedad `userdirectoryid` de los grupos de usuarios a los que pertenece el usuario. Predeterminado: 0. Comportamiento de la propiedad: - *solo lectura* - *compatible* con usuarios de tipo *Super admin*
timezone	string	Zona horaria del usuario, por ejemplo, `Europe/London`, `UTC`. Predeterminado: `default` - valor predeterminado del sistema. Para consultar la lista completa de zonas horarias compatibles, consulte la documentación de PHP.

Medios

El objeto de medios tiene las siguientes propiedades.

Property	Type	Description
mediaid	ID	ID de los medios del usuario. Comportamiento de la propiedad: - *solo lectura*
mediatypeid	ID	ID del tipo de medio utilizado por los medios del usuario. Comportamiento de la propiedad: - *obligatorio*
sendto	string/array	Dirección, nombre de usuario u otro identificador del destinatario. Si `type` de Tipo de medio está establecido en ”Email” o ”Push”, los valores se representan como una matriz. Para ”Push”, la matriz puede contener un comodín (*) o uno o más ID de dispositivos móviles. Si `type` de Tipo de medio está establecido en ”Script”, ”SMS” o ”Webhook”, el valor se representa como una cadena. Comportamiento de la propiedad: - *obligatorio*
active	integer	Indica si los medios están habilitados. Valores posibles: 0 - *(predeterminado)* habilitados; 1 - deshabilitados.

Property	Type	Description
severity	integer	Gravedades de los trigger sobre las que se deben enviar notificaciones. Valores de mapa de bits posibles: 1 - No clasificado; 2 - Información; 4 - Advertencia; 8 - Promedio; 16 - Alto; 32 - Desastre. Este es un campo de máscara de bits; se acepta cualquier suma de los valores de mapa de bits posibles (por ejemplo, 48 para Promedio, Alto y Desastre). Predeterminado: 63.
period	string	Hora en la que se pueden enviar las notificaciones, como un período de tiempo o macros de usuario separadas por un punto y coma. Predeterminado: 1-7,00:00-24:00.
provisioned	integer	Indica si el usuario ha sido aprovisionado. Valores posibles: 0 - no aprovisionado; 1 - aprovisionado.
userdirectory_mediaid	ID	ID de asignación de medios del directorio de usuarios para los medios aprovisionados. Comportamiento de la propiedad: - *solo lectura* - *compatible* con usuarios de tipo *Super admin*

user.checkAuthentication

Descripción

`object user.checkAuthentication`

Este método verifica y prolonga la sesión del usuario.

> **Attention:**
> Llamar al método `user.checkAuthentication` usando el parámetro `sessionid` prolonga la sesión del usuario por defecto.

Parámetros

El método acepta los siguientes parámetros.

Parameter	Type	Description
extend	boolean	Si se debe prolongar la sesión del usuario. Valor predeterminado: ”true”. Establecer el valor en ”false” permite comprobar la sesión del usuario sin prolongarla.
sessionid	string	Comportamiento del parámetro: - *supported* si `sessionid` está establecido Token de autenticación del usuario. Comportamiento del parámetro: - *required* si `token` no está establecido

Parameter	Type	Description
token	string	Token de API del usuario. Comportamiento del parámetro: - *required* si `sessionid` no está establecido

Valores de retorno

`(object)` Devuelve un objeto que contiene información sobre el usuario.

Además de las propiedades estándar del usuario, se devuelve la siguiente información.

Propiedad	Tipo	Descripción
auth_type	integer	Autenticación predeterminada para el usuario. Consulte la propiedad `authentication_type` del objeto Authentication para ver una lista de valores posibles.
debug_mode	integer	Indica si el modo de depuración está habilitado o deshabilitado para el usuario. Consulte la propiedad `debug_mode` del objeto User group para ver una lista de valores posibles.
deprovisioned	boolean	Indica si el usuario pertenece a un grupo de usuarios desprovisionados.
gui_access	string	Método de autenticación del usuario en el frontend. Consulte la propiedad `gui_access` del objeto User group para ver una lista de valores posibles.
secret	string	Cadena aleatoria de 32 caracteres. Se genera al iniciar sesión el usuario. La propiedad `secret` no se devuelve si la sesión del usuario se comprueba utilizando un token de API.
sessionid	string	Token de autenticación, que debe utilizarse en las siguientes solicitudes de la API. La propiedad `sessionid` no se devuelve si la sesión del usuario se comprueba utilizando un token de API.
type	integer	Tipo de usuario. Consulte la propiedad `type` del objeto Role para ver una lista de valores posibles.
userip	string	Dirección IP del usuario.

Ejemplos

Comprobar la autenticación utilizando el token de autenticación

Comprueba y prolonga una sesión de usuario utilizando el token de autenticación del usuario, y devuelve información adicional sobre el usuario.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "user.checkAuthentication",
    "params": {
        "sessionid": "673b8ba11562a35da902c66cf5c23fa2"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userid": "1",
        "username": "Admin",
        "name": "Zabbix",
        "surname": "Administrator",
        "url": "",
        "autologin": "1",
        "autologout": "0",
        "lang": "ru_RU",
        "refresh": "0",
        "theme": "default",
        "attempt_failed": "0",
        "attempt_ip": "127.0.0.1",
        "attempt_clock": "1355919038",
        "rows_per_page": "50",
        "timezone": "Europe/Riga",
        "roleid": "3",
        "userdirectoryid": "0",
        "ts_provisioned": "0",
        "type": 3,
        "userip": "127.0.0.1",
        "debug_mode": 0,
        "gui_access": "0",
        "deprovisioned": false,
        "auth_type": 0,
        "sessionid": "673b8ba11562a35da902c66cf5c23fa2",
        "secret": "0e329b933e46984e49a5c1051ecd0751"
    },
    "id": 1
}
```


Comprobar la autenticación utilizando el token de la API

Comprueba una sesión de usuario utilizando el token de la API de usuario y devuelve información adicional sobre el usuario.

Petición:

```
{
    "jsonrpc": "2.0",
    "method": "user.checkAuthentication",
    "params": {
        "token": "00aff470e07c12d707e50d98cfe39edef9e6ec349c14728dbdfbc8ddc5ea3eae"
    },
    "id": 1
}
```

Respuesta:


```
{
    "jsonrpc": "2.0",
    "result": {
        "userid": "1",
        "username": "Admin",
        "name": "Zabbix",
        "surname": "Administrator",
        "url": "",
        "autologin": "1",
        "autologout": "0",
        "lang": "ru_RU",
        "refresh": "0",
        "theme": "default",
        "attempt_failed": "0",
        "attempt_ip": "127.0.0.1",
```

```
        "attempt_clock": "1355919338",
        "rows_per_page": "50",
        "timezone": "Europe/Riga",
        "roleid": "3",
        "userdirectoryid": "0",
        "ts_provisioned": "0",
        "debug_mode": 0,
        "deprovisioned": false,
        "gui_access": "1",
        "mfaid": 0,
        "auth_type": 0,
        "type": 3,
        "userip": "127.0.0.1"
    },
    "id": 1
}
```


Fuente

CUser::checkAuthentication() en *ui/include/classes/api/services/CUser.php*.

user.create

Descripción

`object user.create(object/array users)`

Este método permite crear nuevos usuarios.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración del rol de usuario. Consulta Roles de usuario para obtener más información.

> **Note:**
> La seguridad de la contraseña del usuario se valida según las reglas de la política de contraseñas definidas por la API de autenticación. Consulta objeto authentication para obtener más información.

Parámetros

`(object/array)` Usuarios a crear.

Además de las propiedades estándar del usuario, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
usrgrps	array	Grupos de usuarios a los que añadir el usuario. Los grupos de usuarios deben tener sólo la propiedad `usrgrpid` definida.
medias	array	Medios de usuario que se crearán.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los usuarios creados bajo la propiedad `userids`. El orden de los IDs devueltos coincide con el orden de los usuarios proporcionados.

Ejemplos

Creación de un usuario

Cree un usuario nuevo, agréguelo a un grupo de usuarios y cree un medio nuevo para él.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "user.create",
    "params": {
        "username": "John",
        "passwd": "Doe12345",
        "roleid": "5",
        "usrgrps": [
            {
                "usrgrpid": "7"
            }
        ],
        "medias": [
            {
                "mediatypeid": "1",
                "sendto": [
                    "support@company.com"
                ],
                "active": 0,
                "severity": 63,
                "period": "1-7,00:00-24:00"
            }
        ]
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userids": [
            "12"
        ]
    },
    "id": 1
}
```

Ver también

- Autenticación
- Medios
- Grupo de usuarios
- Rol

Fuente

CUser::create() en *ui/include/classes/api/services/CUser.php*.

user.delete

Descripción

`object user.delete(array users)`

Este método permite eliminar usuarios.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(array)` IDs de los usuarios a eliminar.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los usuarios eliminados bajo la propiedad `userids`.

Ejemplos

Eliminando varios usuarios

Eliminar dos usuarios.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "user.delete",
    "params": [
        "1",
        "5"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userids": [
            "1",
            "5"
        ]
    },
    "id": 1
}
```

Fuente

CUser::delete() en *ui/include/classes/api/services/CUser.php*.

user.get

Descripción

`integer/array user.get(object parameters)`

El método permite recuperar usuarios según los parámetros proporcionados.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

`(object)` Parámetros que definen la salida deseada.

El método admite los siguientes parámetros.

> **Note:**
> Al solicitar información sobre medios de usuario, permisos o roles, los usuarios de tipo *Admin* y *User* solo pueden recuperar datos sobre su propio usuario. Para un ejemplo, consulte Recuperar usuarios como *Admin*.

Parámetro	Tipo	Descripción
mediaids	ID/array	Devolver solo los usuarios que utilizan los medios indicados.
mediatypeids	ID/array	Devolver solo los usuarios que utilizan los tipos de medios indicados.
userids	ID/array	Devolver solo los usuarios con los IDs indicados.
usrgrpids	ID/array	Devolver solo los usuarios que pertenecen a los grupos de usuarios indicados.

Parámetro	Tipo	Descripción
getAccess	flag	Agrega información adicional sobre los permisos del usuario. Agrega las siguientes propiedades para cada usuario: `gui_access` - `(integer)` método de autenticación del usuario en el frontend. Consulte la propiedad `gui_access` del objeto grupo de usuarios para ver una lista de valores posibles. `debug_mode` - `(integer)` indica si la depuración está habilitada para el usuario. Valores posibles: 0 - depuración deshabilitada, 1 - depuración habilitada. `users_status` - `(integer)` indica si el usuario está deshabilitado. Valores posibles: 0 - usuario habilitado, 1 - usuario deshabilitado.
selectMedias	query	Devolver los medios utilizados por el usuario en la propiedad `medias`.
selectMediatypes	query	Devolver los tipos de medios utilizados por el usuario en la propiedad `mediatypes`. Consulte `mediatype.get` para conocer las restricciones según el tipo de usuario.
selectUsrgrps	query	Devolver los grupos de usuarios a los que pertenece el usuario en la propiedad `usrgrps`. Consulte `usergroup.get` para conocer las restricciones según el tipo de usuario.
selectRole	query	Devolver el rol de usuario en la propiedad `role`.
filter	object	Devolver solo aquellos resultados que coincidan exactamente con el filtro dado. Acepta un objeto, donde las claves son nombres de propiedades y los valores son un único valor o una matriz de valores con los que comparar. No admite propiedades del tipo de datos `text` data type. Posibles propiedades del objeto Usuario para usuarios de tipo *Admin* y *User* al solicitar datos de usuarios en su grupo de usuarios: `userid`, `name`, `surname`, `username`.
output	query	Propiedades del objeto Usuario que se devolverán. Los usuarios de tipo *Admin* y *User* solo pueden recuperar las siguientes propiedades: - Para su propio usuario: `userid`, `attempt_clock`, `attempt_failed`, `attempt_ip`, `autologin`, `autologout`, `lang`, `name`, `provisioned`, `refresh`, `roleid`, `rows_per_page`, `surname`, `theme`, `timezone`, `url`, `username`. - Para usuarios en su grupo de usuarios: `userid`, `name`, `surname`, `username`. Por defecto: `extend`.
search	object	Devolver resultados que coincidan con el patrón dado (no distingue mayúsculas de minúsculas). Acepta un objeto, donde las claves son nombres de propiedades y los valores son cadenas a buscar. Si no se dan opciones adicionales, esto realizará una búsqueda `LIKE "%…%"`. Solo admite propiedades de tipo de datos `string` y `text` data type. Posibles propiedades del objeto Usuario para usuarios de tipo *Admin* y *User* al solicitar datos de usuarios en su grupo de usuarios: `name`, `surname`, `username`.
sortfield	string/array	Ordenar el resultado por las propiedades indicadas. Valores posibles: `userid`, `username`.

Parámetro	Tipo	Descripción
countOutput	boolean	Estos parámetros se describen en el comentario de referencia.
editable	boolean	
excludeSearch	boolean	
limit	integer	
preservekeys	boolean	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

`(entero/array)` Devuelve:

- un array de objetos;
- el número de objetos recuperados, si se ha utilizado el parámetro `countOutput`.

Ejemplos

Recuperación de usuarios

Recupere todos los usuarios configurados.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "user.get",
    "params": {
        "output": "extend"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "userid": "1",
            "username": "Admin",
            "name": "Zabbix",
            "surname": "Administrator",
            "url": "",
            "autologin": "1",
            "autologout": "0",
            "lang": "en_US",
            "default_maintenance_period": "1h",
            "refresh": "30s",
            "theme": "default",
            "attempt_failed": "0",
            "attempt_ip": "",
            "attempt_clock": "0",
            "rows_per_page": "50",
            "timezone": "default",
            "roleid": "3",
            "userdirectoryid": "0",
            "ts_provisioned": "0",
            "provisioned": "0"
        },
        {
            "userid": "2",
            "username": "guest",
            "name": "",
```

```
            "surname": "",
            "url": "",
            "autologin": "0",
            "autologout": "15m",
            "lang": "default",
            "default_maintenance_period": "1h",
            "refresh": "30s",
            "theme": "default",
            "attempt_failed": "0",
            "attempt_ip": "",
            "attempt_clock": "0",
            "rows_per_page": "50",
            "timezone": "default",
            "roleid": "4",
            "userdirectoryid": "0",
            "ts_provisioned": "0",
            "provisioned": "0"
        }
    ],
    "id": 1
}
```


Recuperación de usuarios como *Admin*

Como usuario de tipo *Admin*, recupere datos detallados sobre su propio usuario y datos limitados de los usuarios de su grupo de usuarios.

Solicitud:


```
{
    "jsonrpc": "2.0",
    "method": "user.get",
    "params": {
        "output": "extend",
        "getAccess": true,
        "selectMedias": "extend",
        "selectMediatypes": "extend",
        "selectUsrgrps": "extend",
        "selectRole": "extend"
    },
    "id": 1
}
```


Respuesta:


```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "userid": "1",
      "username": "Admin",
      "name": "Zabbix",
      "surname": "Administrator",
      "url": "",
      "autologin": "1",
      "autologout": "0",
      "lang": "default",
      "default_maintenance_period": "1h",
      "refresh": "30s",
      "theme": "default",
      "attempt_failed": "0",
      "attempt_ip": "",
      "attempt_clock": "0",
      "rows_per_page": "50",
      "timezone": "default",
```

```
      "roleid": "3",
      "userdirectoryid": "0",
      "ts_provisioned": "0",
      "provisioned": "0",
      "gui_access": "1",
      "debug_mode": "0",
      "users_status": "0",
      "usrgrps": [
        {
          "usrgrpid": "7",
          "name": "Zabbix administrators",
          "gui_access": "0",
          "users_status": "0",
          "debug_mode": "0",
          "userdirectoryid": "0",
          "mfa_status": "0",
          "mfaid": "0",
          "proxy_mode": "0",
          "proxy_group_mode": "0"
        },
        {
          "usrgrpid": "13",
          "name": "Internal",
          "gui_access": "1",
          "users_status": "0",
          "debug_mode": "0",
          "userdirectoryid": "0",
          "mfa_status": "0",
          "mfaid": "0",
          "proxy_mode": "0",
          "proxy_group_mode": "0"
        }
      ],
      "medias": [],
      "mediatypes": [],
      "role": {
        "roleid": "3",
        "name": "Super admin role",
        "type": "3",
        "readonly": "1"
      }
    },
    {
      "userid": "2",
      "username": "guest",
      "name": "",
      "surname": "",
      "url": "",
      "autologin": "0",
      "autologout": "15m",
      "lang": "default",
      "default_maintenance_period": "1h",
      "refresh": "30s",
      "theme": "default",
      "attempt_failed": "0",
      "attempt_ip": "",
      "attempt_clock": "0",
      "rows_per_page": "50",
      "timezone": "default",
      "roleid": "4",
      "userdirectoryid": "0",
      "ts_provisioned": "0",
```

```
      "provisioned": "0",
      "gui_access": "1",
      "debug_mode": "0",
      "users_status": "1",
      "usrgrps": [
        {
          "usrgrpid": "8",
          "name": "Guests",
          "gui_access": "0",
          "users_status": "0",
          "debug_mode": "0",
          "userdirectoryid": "0",
          "mfa_status": "0",
          "mfaid": "0",
          "proxy_mode": "0",
          "proxy_group_mode": "0"
        },
        {
          "usrgrpid": "9",
          "name": "Disabled",
          "gui_access": "0",
          "users_status": "1",
          "debug_mode": "0",
          "userdirectoryid": "0",
          "mfa_status": "0",
          "mfaid": "0",
          "proxy_mode": "0",
          "proxy_group_mode": "0"
        },
        {
          "usrgrpid": "13",
          "name": "Internal",
          "gui_access": "1",
          "users_status": "0",
          "debug_mode": "0",
          "userdirectoryid": "0",
          "mfa_status": "0",
          "mfaid": "0",
          "proxy_mode": "0",
          "proxy_group_mode": "0"
        }
      ],
      "medias": [],
      "mediatypes": [],
      "role": {
        "roleid": "4",
        "name": "Guest role",
        "type": "1",
        "readonly": "0"
      }
    }
  ],
  "id": 1
}
```


Recuperación de datos del usuario

Recupere los datos del usuario con el ID ”2”.

Solicitud:


```
{
    "jsonrpc": "2.0",
    "method": "user.get",
```

```
    "params": {
        "output": ["userid", "username"],
        "selectRole": "extend",
        "userids": "2"
    },
    "id": 1
}
```

Respuesta:

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "userid": "2",
      "username": "guest",
      "role": {
        "roleid": "4",
        "name": "Guest role",
        "type": "1",
        "readonly": "0"
      }
    }
  ],
  "id": 1
}
```

Ver también

- Media
- Tipo de media
- Grupo de usuarios
- Rol

Fuente

CUser::get() en *ui/include/classes/api/services/CUser.php*.

user.login

Descripción

`string/object user.login(object parameters)`

Este método le permite iniciar sesión en la API y generar un token de autenticación.

Warning:
Al usar este método, también debe ejecutar user.logout para evitar la generación de un gran número de registros de sesiones abiertas.

Attention:
Este método solo está disponible para usuarios no autenticados que no pertenecen a ningún grupo de usuarios con la autenticación multifactor habilitada.

Parámetros

`(object)` Parámetros que contienen el nombre de usuario y la contraseña.

El método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
password	string	Contraseña del usuario. Comportamiento del parámetro: - *obligatorio*

Parámetro	Tipo	Descripción
username	string	Nombre de usuario. Comportamiento del parámetro: - *obligatorio*
userData	flag	Devuelve información sobre el usuario autenticado.

Valores de retorno

`(string/object)` Si se utiliza el parámetro `userData`, devuelve un objeto que contiene información sobre el usuario autenticado.

Además de las propiedades estándar del usuario, se devuelve la siguiente información:

Propiedad	Tipo	Descripción
auth_type	integer	Autenticación predeterminada para el usuario. Consulte la propiedad `authentication_type` del objeto Authentication para ver una lista de valores posibles.
debug_mode	integer	Indica si el modo de depuración está habilitado o deshabilitado para el usuario. Consulte la propiedad `debug_mode` del objeto User group para ver una lista de valores posibles.
deprovisioned	boolean	Indica si el usuario pertenece a un grupo de usuarios desprovisionados.
gui_access	string	Método de autenticación del usuario en el frontend. Consulte la propiedad `gui_access` del objeto User group para ver una lista de valores posibles.
mfaid	integer	ID del método MFA que se utilizará para el usuario durante el inicio de sesión. Devuelve "0" si MFA está deshabilitado globalmente o para todos los grupos de usuarios a los que pertenece el usuario.
secret	string	Cadena aleatoria de 32 caracteres. Se genera al iniciar sesión el usuario.
sessionid	string	Token de autenticación, que debe utilizarse en las siguientes solicitudes de la API.
type	integer	Tipo de usuario. Consulte la propiedad `type` del objeto Role para ver una lista de valores posibles.
userip	string	Dirección IP del usuario.

Note:
Si un usuario ha sido autenticado correctamente después de uno o más intentos fallidos, el método devolverá los valores actuales de las propiedades `attempt_clock`, `attempt_failed` y `attempt_ip` y luego las restablecerá.

Si no se utiliza el parámetro `userData`, el método devuelve un token de autenticación que se requiere para la autenticación.

Ejemplos

Autenticando un usuario

Autenticar un usuario.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "user.login",
    "params": {
        "username": "Admin",
```

```
        "password": "zabbix"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": "0424bd59b807674191e7d77572075f33",
    "id": 1
}
```

Solicitar información del usuario autenticado

Autentica y devuelve información adicional sobre el usuario.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "user.login",
    "params": {
        "username": "Admin",
        "password": "zabbix",
        "userData": true
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userid": "1",
        "username": "Admin",
        "name": "Zabbix",
        "surname": "Administrator",
        "url": "",
        "autologin": "1",
        "autologout": "0",
        "lang": "ru_RU",
        "refresh": "0",
        "theme": "default",
        "attempt_failed": "0",
        "attempt_ip": "127.0.0.1",
        "attempt_clock": "1355919038",
        "rows_per_page": "50",
        "timezone": "Europe/Riga",
        "roleid": "3",
        "userdirectoryid": "0",
        "type": 3,
        "userip": "127.0.0.1",
        "debug_mode": 0,
        "gui_access": "0",
        "mfaid": "1",
        "deprovisioned": false,
        "auth_type": 0,
        "sessionid": "5b56eee8be445e98f0bd42b435736e42",
        "secret": "cd0ba923319741c6586f3d866423a8f4"
    },
    "id": 1
}
```

Véase también

- user.logout

Fuente

CUser::login() en *ui/include/classes/api/services/CUser.php*.

user.logout

Descripción

`string/object user.logout(array)`

Este método le permite cerrar sesión de la API e invalida el token de autenticación actual.

> **Note:**
> Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método pueden revocarse en la configuración del rol de usuario. Consulte User roles para obtener más información.

Parámetros

`(array)` El método acepta un array vacío.

Valores de retorno

`(boolean)` Devuelve `true` si el usuario ha cerrado sesión correctamente.

Ejemplos

Cerrar sesión

Cerrar sesión en la API.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "user.logout",
    "params": [],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": true,
    "id": 1
}
```

Véase también

- user.login

Fuente

CUser::login() en *ui/include/classes/api/services/CUser.php*.

user.provision

Descripción

`object user.provision(object/array users)`

Este método permite aprovisionar usuarios LDAP.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración de roles de usuario. Consulte User roles para obtener más información.

Parámetros

(`array`) IDs de los usuarios a aprovisionar.

Valores de retorno

(`object`) Devuelve un objeto que contiene los IDs de los usuarios aprovisionados bajo la propiedad `userids`.

Ejemplos

Aprovisionamiento de varios usuarios

Aprovisionar dos usuarios.

Petición:

```
{
    "jsonrpc": "2.0",
    "method": "user.provision",
    "params": [
        "1",
        "5"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userids": [
            "1",
            "5"
        ]
    },
    "id": 1
}
```

Fuente

CUser::provision() en *ui/include/classes/api/services/CUser.php*.

user.resettotp

Descripción

`object user.resettotp(object/array users)`

Este método permite restablecer los secretos TOTP de los usuarios.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método se pueden revocar en la configuración del rol de usuario. Consulte User roles para obtener más información.

Parámetros

(`array`) IDs de los usuarios para los que se restablecerán los secretos TOTP.

> **Note:**
> Las sesiones de usuario para los usuarios especificados también se eliminarán (excepto para el usuario que envía la solicitud).

Valores de retorno

(`object`) Devuelve un objeto que contiene los IDs de los usuarios para los que se han restablecido los secretos TOTP, bajo la propiedad `userids`.

Ejemplos

Restablecer secretos TOTP para varios usuarios

Restablecer secretos TOTP para dos usuarios.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "user.resettotp",
    "params": [
        "1",
        "5"
    ],
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userids": [
            "1",
            "5"
        ]
    },
    "id": 1
}
```

Véase también

- Objeto MFA

Fuente

CUser::resettotp() en *ui/include/classes/api/services/CUser.php*.

user.unblock

Descripción

`object user.unblock(array userids)`

Este método permite desbloquear usuarios.

> **Note:**
> Este método solo está disponible para el tipo de usuario *Super admin*. Los permisos para llamar al método pueden revocarse en la configuración del rol de usuario. Consulte User roles para obtener más información.

Parámetros

`(array)` IDs de usuarios a desbloquear.

Valores de retorno

`(object)` Devuelve un objeto que contiene los IDs de los usuarios desbloqueados bajo la propiedad `userids`.

Ejemplos

Desbloquear varios usuarios

Desbloquear dos usuarios.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "user.unblock",
    "params": [
        "1",
        "5"
    ],
```

```
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userids": [
            "1",
            "5"
        ]
    },
    "id": 1
}
```

Fuente

CUser::unblock() en *ui/include/classes/api/services/CUser.php*.

user.update

Descripción

`object user.update(object/array users)`

Este método le permite actualizar usuarios existentes.

Note:
Este método está disponible para usuarios de cualquier tipo. Los permisos para llamar al método se pueden revocar en la configuración de roles de usuario. Consulte User roles para obtener más información.

Note:
La fortaleza de la contraseña del usuario se valida de acuerdo con las reglas de la directiva de contraseñas definidas por Authentication API. Consulte Authentication API para obtener más información.

Parámetros

`(object/array)` Propiedades del usuario que se van a actualizar.

La propiedad `userid` debe estar definida para cada usuario, todas las demás propiedades son opcionales. Sólo se actualizarán las propiedades pasadas, todas las demás permanecerán sin cambios.

Además de las propiedades estándar del usuario, el método acepta los siguientes parámetros.

Parámetro	Tipo	Descripción
current_passwd	string	Contraseña actual del usuario. El valor de este parámetro puede ser una cadena vacía si el usuario está vinculado a un directorio de usuarios. Comportamiento del parámetro: - *sólo escritura* - *obligatorio* si se establece `passwd` del objeto Usuario y el usuario cambia su propia contraseña
usrgrps	array	Grupos de usuarios para reemplazar los grupos de usuarios existentes. Los grupos de usuarios sólo deben tener definida la propiedad `usrgrpid`.
medias	array	Medios de usuario para reemplazar los medios existentes no aprovisionados. Los medios aprovisionados pueden omitirse al actualizar los medios.

Valores de retorno

(`object`) Devuelve un objeto que contiene los IDs de los usuarios actualizados bajo la propiedad `userids`.

Ejemplos

Renombrar un usuario

Renombrar un usuario a John Doe.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "user.update",
    "params": {
        "userid": "1",
        "name": "John",
        "surname": "Doe"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userids": [
            "1"
        ]
    },
    "id": 1
}
```

Cambiar el rol de usuario

Cambiar el rol de un usuario.

Petición:

```
{
    "jsonrpc": "2.0",
    "method": "user.update",
    "params": {
        "userid": "12",
        "roleid": "6"
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userids": [
            "12"
        ]
    },
    "id": 1
}
```

Véase también

- Autenticación

Fuente

CUser::update() en *ui/include/classes/api/services/CUser.php*.

Notación Firma del método

Cada página de método de la API contiene la firma del método.

Ejemplo: `object action.create(object/array actions)`

Firma	Descripción
`object`	El tipo de datos que devuelve el método.
`action.create`	El nombre completo del método.
`object/array`	El tipo de datos que acepta el método.
`actions`	Un breve resumen de lo que acepta el método.

Tipos de datos

La API de Zabbix admite los siguientes tipos de datos como entrada:

Tipo	Descripción
ID	Un identificador único utilizado para hacer referencia a una entidad.
boolean	Un valor booleano (ya sea `true` o `false`).
flag	Un valor que se considera `true` si se pasa y no es igual a `null`; de lo contrario, el valor se considera `false`.
integer	Un número entero.
float	Un número de punto flotante.
string	Una cadena de texto.
text	Una cadena de texto más larga.
timestamp	Una marca de tiempo Unix.
array	Una secuencia ordenada de valores (un array simple).
object	Un array asociativo.
query	Un valor que define los datos a devolver. El valor puede definirse como un array de nombres de propiedades (para devolver solo propiedades específicas), o como uno de los valores predefinidos: `extend` - devuelve todas las propiedades del objeto; `count` - devuelve el número de registros recuperados, solo es compatible con ciertos subselects.

Attention:
La API de Zabbix siempre devuelve valores solo como cadenas o arrays.

Comportamiento de la propiedad

Algunas de las propiedades del objeto están marcadas con etiquetas cortas para describir su comportamiento. Se utilizan las siguientes etiquetas:

- ***solo lectura***: el valor de la propiedad se establece automáticamente y no puede ser definido o cambiado por el usuario, incluso en algunas condiciones específicas (por ejemplo, *solo lectura* para objetos heredados u objetos descubiertos);
- ***solo escritura***: el valor de la propiedad puede establecerse, pero no puede accederse después;
- ***constante***: el valor de la propiedad puede establecerse al crear un objeto, pero no puede cambiarse después;
- ***soportado***: no es obligatorio establecer el valor de la propiedad, pero se permite establecerlo en algunas condiciones específicas (por ejemplo, *soportado* si `type` se establece en ”Comprobación simple”, ”Comprobación externa”, ”Agente SSH”, ”Agente TELNET” o ”Agente HTTP”); sin embargo, tenga en cuenta que las propiedades *soportadas* aún pueden establecerse en sus valores predeterminados independientemente de las condiciones;
- ***requerido***: es obligatorio establecer el valor de la propiedad para todas las operaciones (excepto las operaciones get) o en algunas condiciones específicas (por ejemplo, *requerido* para operaciones de creación; *requerido* si `operationtype` se establece en ”script global” y `opcommand_hst` no está establecido).

Note:
Para las operaciones de actualización, una propiedad se considera como ”establecida” cuando se establece durante la operación de actualización.

Las propiedades que no están marcadas con etiquetas son opcionales.

Comportamiento de los parámetros

Algunos de los parámetros de operación están marcados con etiquetas cortas para describir su comportamiento para la operación. Se utilizan las siguientes etiquetas:

- ***solo-lectura*** - el valor del parámetro se establece automáticamente y no puede ser definido o cambiado por el usuario, incluso en algunas condiciones específicas (por ejemplo, *solo-lectura* para objetos heredados u objetos descubiertos);
- ***solo-escritura*** - el valor del parámetro puede establecerse, pero no puede accederse después;
- ***soportado*** - no se requiere que el valor del parámetro se establezca, pero se permite establecerlo en algunas condiciones específicas (por ejemplo, *soportado* si `operating_mode` del objeto Proxy se establece en "proxy pasivo"); tenga en cuenta, sin embargo, que los parámetros *soportados* aún pueden establecerse en sus valores predeterminados independientemente de las condiciones;
- ***requerido*** - se requiere que el valor del parámetro se establezca.

Los parámetros que no están marcados con etiquetas son opcionales.

Valor de ID reservado "0" El valor de ID reservado "0" puede usarse para filtrar elementos y para eliminar objetos referenciados. Por ejemplo, para eliminar un proxy referenciado de un equipo, proxyid debe establecerse en 0 ("proxyid": "0") o para filtrar equipos monitorizados por la opción server, proxyids debe establecerse en 0 ("proxyids": "0").

Parámetros comunes de los métodos "get" Los siguientes parámetros son compatibles con todos los métodos `get`:

Parámetro	Tipo	Descripción
countOutput	boolean	Devuelve el número de registros en el resultado en lugar de los datos reales.
editable	boolean	Si se establece en `true`, devuelve solo los objetos para los que el usuario tiene permisos de escritura. Por defecto: `false`.
excludeSearch	boolean	Devuelve resultados que no coinciden con los criterios dados en el parámetro `search`.
filter	object	Devuelve solo aquellos resultados que coinciden exactamente con el filtro dado. Acepta un objeto, donde las claves son nombres de propiedades (por ejemplo, propiedades del objeto Host en `host.get`, propiedades del objeto Item en `item.get`, etc.), y los valores son un único valor o un array de valores con los que comparar. No admite propiedades del tipo de dato `text` data type. Tenga en cuenta que algunos métodos tienen una funcionalidad específica para este parámetro, que se describe en la página del método (por ejemplo, el parámetro `filter` en host.get también admite propiedades de la interfaz de Host).
limit	integer	Limita el número de registros devueltos.
output	query	Propiedades del objeto que se devolverán. Tenga en cuenta que el ID del objeto (es decir, `hostid`, `itemid`, etc.) siempre se incluye en la respuesta, incluso si no se especifica en el parámetro `output`. Por defecto: `extend`.
preservekeys	boolean	Utiliza los IDs como claves en el array resultante.
search	object	Devuelve resultados que coinciden con el patrón dado (no distingue entre mayúsculas y minúsculas). Acepta un objeto, donde las claves son nombres de propiedades (por ejemplo, propiedades del objeto Host en `host.get`, propiedades del objeto Item en `item.get`, etc.), y los valores son cadenas a buscar. Si no se dan opciones adicionales, esto realizará una búsqueda `LIKE "%...%"`. Solo admite propiedades de tipo de dato `string` y `text` data type. Tenga en cuenta que algunos métodos tienen una funcionalidad específica para este parámetro, que se describe en la página del método (por ejemplo, el parámetro `search` en host.get también admite propiedades de la interfaz de Host).

Parámetro	Tipo	Descripción
searchByAny	boolean	Si se establece en `true`, devuelve resultados que coinciden con cualquiera de los criterios dados en el parámetro `filter` o `search` en lugar de todos ellos. Por defecto: `false`.
searchWildcardsEnabled	boolean	Si se establece en `true`, habilita el uso de "*" como carácter comodín en el parámetro `search`. Por defecto: `false`.
sortfield	string/array	Ordena el resultado por las propiedades dadas. Consulte la descripción de un método get específico de la API para obtener una lista de las propiedades que se pueden utilizar para ordenar. Las macros no se expanden antes de ordenar. Si no se especifica ningún valor, los datos se devolverán sin ordenar.
sortorder	string/array	Orden de clasificación. Si se pasa un array, cada valor se emparejará con la propiedad correspondiente dada en el parámetro `sortfield`. Valores posibles: `ASC` - *(por defecto)* ascendente; `DESC` - descendente.
startSearch	boolean	El parámetro `search` comparará el comienzo de los campos, es decir, realizará una búsqueda `LIKE "…%"` en su lugar. Se ignora si `searchWildcardsEnabled` está establecido en `true`.

Indicadores de origen de entidad Los métodos Get devuelven una propiedad `flags` para las entidades relacionadas con el descubrimiento de bajo nivel (regla LLD/prototipo de regla LLD, métrica/prototipo de métrica, etc). Esta propiedad es útil para indicar si la entidad ha sido descubierta o no, ya que la edición de entidades descubiertas está limitada.

La propiedad `flags` devuelve un resultado basado en una combinación (operación "+") de estos valores:

Valor	Descripción
0	Entidad base (métrica, disparador, gráfico, equipo)
1	Regla de descubrimiento de bajo nivel
2	Cualquier prototipo (prototipo de métrica, prototipo de disparador, prototipo de regla LLD, etc)
4	Entidad descubierta (métrica, disparador, gráfico, equipo, regla LLD descubierta)

El valor **combinado** devuelto por la propiedad `flags` puede ser:

Valor	Combinación de	Descripción
0	0	Entidad simple (métrica, disparador, gráfico, equipo).
2	2	Prototipo de entidad (prototipo de métrica, prototipo de disparador, etc).
6	2+4	Métrica, disparador, gráfico, equipo descubierto (convertido desde prototipo).
1	1	Regla de descubrimiento de bajo nivel.
3	1+2	Prototipo de regla de descubrimiento de bajo nivel.
5	1+4	Regla de descubrimiento de bajo nivel descubierta (convertida desde prototipo).
7	1+2+4	Prototipo de regla de descubrimiento de bajo nivel descubierta.

Ejemplos Comprobación de permisos de usuario

¿Tiene el usuario permiso para escribir en equipos cuyos nombres comienzan con "MySQL" o "Linux"?

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
```

```
        "countOutput": true,
        "search": {
            "host": ["MySQL", "Linux"]
        },
        "editable": true,
        "startSearch": true,
        "searchByAny": true
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": "0",
    "id": 1
}
```

Note:
Un resultado cero significa que no hay equipos con permisos de lectura/escritura.

Conteo de no coincidencias

Cuenta el número de equipos cuyos nombres no contienen la subcadena ”ubuntu”

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "countOutput": true,
        "search": {
            "host": "ubuntu"
        },
        "excludeSearch": true
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": "44",
    "id": 1
}
```

Búsqueda de equipos usando comodines

Encuentra equipos cuyo nombre contenga la palabra ”server” y tengan puertos de interfaz ”10050” o ”10071”. Ordena el resultado por nombre de equipo en orden descendente y limítalo a 5 equipos.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": ["hostid", "host"],
        "selectInterfaces": ["port"],
        "filter": {
            "port": ["10050", "10071"]
        },
        "search": {
            "host": "*server*"
```

```
        },
        "searchWildcardsEnabled": true,
        "searchByAny": true,
        "sortfield": "host",
        "sortorder": "DESC",
        "limit": 5
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "hostid": "50003",
            "host": "WebServer-Tomcat02",
            "interfaces": [
                {
                    "port": "10071"
                }
            ]
        },
        {
            "hostid": "50005",
            "host": "WebServer-Tomcat01",
            "interfaces": [
                {
                    "port": "10071"
                }
            ]
        },
        {
            "hostid": "50004",
            "host": "WebServer-Nginx",
            "interfaces": [
                {
                    "port": "10071"
                }
            ]
        },
        {
            "hostid": "99032",
            "host": "MySQL server 01",
            "interfaces": [
                {
                    "port": "10050"
                }
            ]
        },
        {
            "hostid": "99061",
            "host": "Linux server 01",
            "interfaces": [
                {
                    "port": "10050"
                }
            ]
        }
    ],
    "id": 1
}
```

Búsqueda de equipos usando comodines con ”preservekeys”

Si añade el parámetro ”preservekeys” a la solicitud anterior, el resultado se devuelve como un array asociativo, donde las claves son el id de los objetos.

Solicitud:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": ["hostid", "host"],
        "selectInterfaces": ["port"],
        "filter": {
            "port": ["10050", "10071"]
        },
        "search": {
            "host": "*server*"
        },
        "searchWildcardsEnabled": true,
        "searchByAny": true,
        "sortfield": "host",
        "sortorder": "DESC",
        "limit": 5,
        "preservekeys": true
    },
    "id": 1
}
```

Respuesta:

```
{
    "jsonrpc": "2.0",
    "result": {
        "50003": {
            "hostid": "50003",
            "host": "WebServer-Tomcat02",
            "interfaces": [
                {
                    "port": "10071"
                }
            ]
        },
        "50005": {
            "hostid": "50005",
            "host": "WebServer-Tomcat01",
            "interfaces": [
                {
                    "port": "10071"
                }
            ]
        },
        "50004": {
            "hostid": "50004",
            "host": "WebServer-Nginx",
            "interfaces": [
                {
                    "port": "10071"
                }
            ]
        },
        "99032": {
            "hostid": "99032",
            "host": "MySQL server 01",
            "interfaces": [
```

```
                {
                    "port": "10050"
                }
            ]
        },
        "99061": {
            "hostid": "99061",
            "host": "Linux server 01",
            "interfaces": [
                {
                    "port": "10050"
                }
            ]
        }
    },
    "id": 1
}
```

Apéndice 2. Cambios de 7.4 a 8.0

Cambios incompatibles

Otros cambios y correcciones de errores

Apéndice 3. Cambios en 8.0

8.0.1 Esta versión aún no ha sido publicada.

Extensiones

Descripción general Aunque Zabbix ofrece una multiplicidad de funciones, siempre hay espacio para funcionalidades adicionales. Las extensiones son una forma conveniente de modificar y mejorar las capacidades de monitorización de Zabbix sin cambiar su código fuente.

Puede ampliar la funcionalidad de Zabbix utilizando opciones de extensión integradas (métricas trapper, parámetros de usuario, etc.) o utilizando o creando extensiones personalizadas (módulos cargables, plugins, etc.).

Esta sección proporciona una descripción general con referencias a todas las opciones para ampliar Zabbix.

Recopilación de datos con comandos personalizados Items trapper

Los items trapper son items que aceptan datos entrantes en lugar de consultarlos. Los items trapper son útiles para enviar datos específicos al server o proxy de Zabbix, por ejemplo, datos periódicos de disponibilidad y rendimiento en el caso de scripts de usuario de larga ejecución. El envío de datos al server o proxy de Zabbix es posible mediante la utilidad Zabbix sender o el protocolo de Zabbix sender. También es posible enviar datos al server de Zabbix mediante el método de API `history.push`.

Comprobaciones externas

Una comprobación externa es una métrica para ejecutar comprobaciones mediante la ejecución de un ejecutable, por ejemplo, un script de shell o un binario.

Las comprobaciones externas son ejecutadas por el servidor Zabbix o el proxy (cuando el equipo es monitorizado por un proxy), y no requieren que un agente se esté ejecutando en el equipo monitorizado.

Parámetros de usuario

Un parámetro de usuario es un comando definido por el usuario (asociado a una clave definida por el usuario) que, cuando se ejecuta, puede recuperar los datos que necesita del equipo donde se está ejecutando el agente Zabbix. Los parámetros de usuario son útiles para configurar métricas de agente o agente 2 que no están predefinidas en Zabbix.

Items de Zabbix agent `system.run[]`

`system.run[]` de Zabbix agent item es un item para un comando definido por el usuario (asociado con una clave predefinida `system.run[]`, por ejemplo, `system.run[myscript.sh]`) que se puede ejecutar en el host donde se está ejecutando Zabbix agent.

Note:
Los items `system.run[]` están deshabilitados de forma predeterminada y, si se utilizan, deben habilitarse (permitirse) y definirse en el archivo de configuración de Zabbix agent o Zabbix agent 2 (parámetro de configuración `AllowKey` o `AllowKeyRegexp`).

Attention:
Los comandos definidos por el usuario en métricas como comprobaciones externas, parámetros de usuario y métricas del agente Zabbix `system.run[]` se ejecutan desde el usuario del sistema operativo que se utiliza para ejecutar los componentes de Zabbix. Para ejecutar estos comandos, este usuario debe tener los permisos necesarios.

Items de HTTP agent

Un item de HTTP agent es un item para ejecutar solicitudes de datos mediante HTTP/HTTPS.
Los items de HTTP agent son útiles para enviar solicitudes a endpoints HTTP con el fin de recuperar datos de servicios como *Elasticsearch* y *OpenWeatherMap*, comprobar el estado de la API de Zabbix o el estado de un web server Apache o NGINX, entre otros.
Los items de HTTP agent (con la recepción de datos habilitada) también pueden funcionar como items trapper.

Métricas de script

Una métrica de script es una métrica para ejecutar código JavaScript definido por el usuario que recupera datos a través de HTTP/HTTPS. Las métricas de script son útiles cuando la funcionalidad proporcionada por las métricas de agente HTTP no es suficiente. Por ejemplo, en escenarios exigentes de recopilación de datos que requieren múltiples pasos o lógica compleja, se puede configurar una métrica de script para realizar una llamada HTTP, luego procesar los datos recibidos, y luego pasar el valor transformado a una segunda llamada HTTP.

Note:
Las métricas de agente HTTP y las métricas de script son compatibles con el servidor y el proxy de Zabbix, y no requieren que un agente se esté ejecutando en el equipo monitorizado.

Extensiones avanzadas Módulos cargables

Los módulos cargables, escritos en C, son una opción versátil y orientada al rendimiento para ampliar la funcionalidad de los componentes de Zabbix (servidor, proxy, agente) en plataformas UNIX. Un módulo cargable es básicamente una biblioteca compartida utilizada por el demonio de Zabbix y cargada al inicio. La biblioteca debe contener ciertas funciones, de modo que un proceso de Zabbix pueda detectar que el archivo es realmente un módulo que puede cargar y con el que puede trabajar.

Los módulos cargables tienen una serie de ventajas, incluyendo la capacidad de añadir nuevas métricas o implementar cualquier otra lógica (por ejemplo, la exportación de datos históricos de Zabbix), un gran rendimiento y la opción de desarrollar, utilizar y compartir la funcionalidad que proporcionan. Esto contribuye a un mantenimiento sin problemas y ayuda a ofrecer nuevas funcionalidades de forma más sencilla e independiente del núcleo de código de Zabbix.

Los módulos cargables son especialmente útiles en una configuración de monitorización compleja. Al monitorizar sistemas embebidos, tener un gran número de parámetros monitorizados o scripts pesados con lógica compleja o un largo tiempo de inicio, extensiones como parámetros de usuario, métricas del agente Zabbix `system.run[]` y comprobaciones externas tendrán un impacto en el rendimiento. Los módulos cargables ofrecen una forma de ampliar la funcionalidad de Zabbix sin sacrificar el rendimiento.

Plugins

Los plugins amplían las capacidades de monitorización del agent 2 de Zabbix. Escritos en el lenguaje de programación Go, proporcionan una alternativa a los módulos cargables (escritos en C).

Es posible desarrollar su propio plugin (recolector de datos independiente) para el agent 2 de Zabbix. Consulte el Centro de desarrollo.

Personalización de alertas Webhooks

Un webhook es un tipo de medio de Zabbix que proporciona una opción para ampliar las capacidades de alerta de Zabbix a software externo como sistemas de helpdesk, chats o mensajería. De manera similar a las métricas de script, los webhooks son útiles para realizar llamadas HTTP utilizando código JavaScript personalizado, por ejemplo, para enviar notificaciones a diferentes plataformas como Microsoft Teams, Discord y Jira. También es posible devolver algunos datos (por ejemplo, sobre tickets de helpdesk creados) que luego se muestran en Zabbix.

Los webhooks existentes están disponibles en el repositorio Git de Zabbix. Para el desarrollo de webhooks personalizados, consulte la Guía de desarrollo de Webhooks.

Scripts de alerta

Un script de alerta es un tipo de medio de Zabbix que proporciona una opción para crear una forma alternativa (script) de gestionar las alertas de Zabbix. Los scripts de alerta son útiles si no está satisfecho con los tipos de medios existentes para enviar alertas en Zabbix.

Personalización de la interfaz Temas personalizados

Es posible cambiar la apariencia visual del frontend de Zabbix utilizando temas personalizados. Consulte las instrucciones para crear y aplicar sus propios temas.

Módulos de frontend

Los módulos de frontend ofrecen la opción de ampliar la funcionalidad del frontend de Zabbix añadiendo módulos de terceros o desarrollando los suyos propios. Con los módulos de frontend puede añadir nuevos elementos de menú, sus vistas respectivas, acciones, etc.

Scripts globales Un script global es un conjunto de comandos definidos por el usuario que pueden ejecutarse en un objetivo de monitorización (por el intérprete de shell (/bin/sh)), dependiendo del alcance configurado y los permisos del usuario. Los scripts globales pueden configurarse para las siguientes acciones:

- Operación de acción
- Acción manual de equipo
- Acción manual de evento

Los scripts globales son útiles en muchos casos. Por ejemplo, si se configuran para operaciones de acción o acciones manuales de equipo, puede utilizar scripts globales para ejecutar automáticamente o manualmente comandos remotos como reiniciar una aplicación (servidor web, middleware, CRM, etc.) o liberar espacio en disco (eliminando archivos antiguos, limpiando `/tmp`, etc). O, otro ejemplo, si se configuran para acciones manuales de eventos, puede utilizar scripts globales para gestionar tickets de problemas en sistemas externos.

Los scripts globales pueden ser ejecutados por el servidor, proxy o agente de Zabbix.

> **Attention:**
> Los comandos definidos por el usuario se ejecutan desde el usuario del sistema operativo que se utiliza para ejecutar los componentes de Zabbix. Para ejecutar estos comandos, este usuario debe tener los permisos necesarios.

Zabbix API Zabbix API es una API basada en HTTP que forma parte del frontend de Zabbix. Con Zabbix API, puede realizar cualquiera de las siguientes operaciones:

- Recuperar y modificar programáticamente la configuración de Zabbix.
- Importar y exportar la configuración de Zabbix.
- Acceder a los datos históricos y de tendencias de Zabbix.
- Configurar aplicaciones para que funcionen con Zabbix.
- Integrar Zabbix con software de terceros.
- Automatizar tareas rutinarias.

Zabbix API consta de una multiplicidad de métodos que se agrupan nominalmente en APIs separadas. Cada método realiza una tarea específica. Para conocer los métodos disponibles, así como una visión general de las funciones proporcionadas por Zabbix API, consulte la Referencia de métodos de Zabbix API.

Módulos cargables

Descripción general

Los módulos cargables ofrecen una opción orientada al rendimiento para ampliar la funcionalidad de Zabbix.

Puede ampliar la funcionalidad de Zabbix de muchas maneras, por ejemplo, con parámetros de usuario, comprobaciones externas y elementos del agent de Zabbix system.run. Estos funcionan muy bien, pero tienen un gran inconveniente, concretamente fork(). Zabbix tiene que bifurcar un nuevo proceso cada vez que maneja una métrica de usuario, lo que no es bueno para el rendimiento. Normalmente no es un gran problema, sin embargo, podría ser un problema grave al monitorizar sistemas embebidos, tener un gran número de parámetros monitorizados o scripts pesados con lógica compleja o un largo tiempo de inicio.

El soporte de módulos cargables ofrece formas de ampliar el agent, el server y el proxy de Zabbix sin sacrificar el rendimiento.

Un módulo cargable es básicamente una biblioteca compartida utilizada por el demonio de Zabbix y cargada al inicio. La biblioteca debe contener ciertas funciones, de modo que un proceso de Zabbix pueda detectar que el archivo es realmente un módulo que puede cargar y con el que puede trabajar.

Los módulos cargables tienen una serie de ventajas. El gran rendimiento y la capacidad de implementar cualquier lógica son muy importantes, pero quizás la ventaja más importante es la capacidad de desarrollar, utilizar y compartir módulos de Zabbix. Contribuye a un mantenimiento sin problemas y ayuda a ofrecer nuevas funcionalidades más fácilmente e independientemente de la base de código principal de Zabbix.

La licencia de los módulos y su distribución en forma binaria se rige por la licencia AGPL-3.0 (los módulos se enlazan con Zabbix en tiempo de ejecución y utilizan cabeceras de Zabbix; todo el código de Zabbix está licenciado bajo AGPL-3.0 desde Zabbix 7.0). Zabbix no garantiza la compatibilidad binaria.

La estabilidad de la API del módulo está garantizada durante un ciclo de lanzamiento de Zabbix LTS (Long Term Support). No se garantiza la estabilidad de la API de Zabbix (técnicamente es posible llamar a funciones internas de Zabbix desde un módulo, pero no hay garantía de que dichos módulos funcionen).

API de módulos

Para que una biblioteca compartida sea tratada como un módulo de Zabbix, debe implementar y exportar varias funciones. Actualmente hay seis funciones en la API de módulos de Zabbix, de las cuales sólo una es obligatoria y las otras cinco son opcionales.

Interfaz obligatoria

La única función obligatoria es `zbx_module_api_version()`:

```
int zbx_module_api_version(void);
```

Esta función debe devolver la versión de la API implementada por este módulo y, para que el módulo se cargue, esta versión debe coincidir con la versión de la API del módulo compatible con Zabbix. La versión de la API del módulo compatible con Zabbix es `ZBX_MODULE_API_VERSION`. Por lo tanto, esta función debe devolver esta constante. La constante antigua `ZBX_MODULE_API_VERSION_ONE`, utilizada con este propósito, ahora está definida como igual a `ZBX_MODULE_API_VERSION` para preservar la compatibilidad con el código fuente, pero no se recomienda su uso.

Interfaz opcional

Las funciones opcionales son `zbx_module_init()`, `zbx_module_item_list()`, `zbx_module_item_timeout()`, `zbx_module_history` y `zbx_module_uninit()`:

```
int zbx_module_init(void);
```

Esta función debe realizar la inicialización necesaria para el módulo, si la hubiera. Si tiene éxito, debe devolver `ZBX_MODULE_OK`. En caso contrario, debe devolver `ZBX_MODULE_FAIL`. En este último caso, Zabbix no se iniciará.

```
ZBX_METRIC  *zbx_module_item_list(void);
```

Esta función debe devolver una lista de items compatibles con el módulo. Cada item se define en una estructura `ZBX_METRIC`; consulte la sección siguiente para obtener más detalles. La lista termina con una estructura `ZBX_METRIC` cuyo campo "key" es NULL.

```
void    zbx_module_item_timeout(int timeout);
```

Si el módulo exporta `zbx_module_item_list()`, entonces Zabbix utiliza esta función para especificar los ajustes de tiempo de espera en el archivo de configuración de Zabbix que deben respetar las comprobaciones de items implementadas por el módulo. Aquí, el parámetro "timeout" está en segundos.

```
ZBX_HISTORY_WRITE_CBS   zbx_module_history_write_cbs(void);
```

Esta función debe devolver las funciones de callback que Zabbix server utilizará para exportar el historial de distintos tipos de datos. Las funciones de callback se proporcionan como campos de la estructura `ZBX_HISTORY_WRITE_CBS`; los campos pueden ser NULL si al módulo no le interesa el historial de un tipo concreto.

```
int zbx_module_uninit(void);
```

Esta función debe realizar la desinicialización necesaria, si la hubiera, como liberar recursos asignados, cerrar descriptores de archivo, etc.

Todas las funciones se llaman una vez al iniciar Zabbix cuando se carga el módulo, con la excepción de `zbx_module_uninit()`, que se llama una vez al apagar Zabbix cuando se descarga el módulo.

Definición de items

Cada item se define en una estructura `ZBX_METRIC`:

```
typedef struct
{
    char        *key;
    unsigned    flags;
    int     (*function)();
    char        *test_param;
}
ZBX_METRIC;
```

Aquí, **key** es la clave del item (por ejemplo, "dummy.random"), **flags** es `CF_HAVEPARAMS` o 0 (según si el item acepta parámetros o no), **function** es una función C que implementa el item (por ejemplo, `zbx_module_dummy_random`), y **test_param** es la lista de parámetros que se usará cuando el agent de Zabbix se inicie con la opción `-p` (por ejemplo, "1,1000", puede ser NULL). Una definición de ejemplo podría verse así:

```
static ZBX_METRIC keys[] =
{
    { "dummy.random", CF_HAVEPARAMS, zbx_module_dummy_random, "1,1000" },
    { NULL }
}
```

Cada función que implementa un item debe aceptar dos parámetros puntero: el primero de tipo `AGENT_REQUEST` y el segundo de tipo `AGENT_RESULT`:

```
int zbx_module_dummy_random(AGENT_REQUEST *request, AGENT_RESULT *result)
{
    ...

    SET_UI64_RESULT(result, from + rand() % (to - from + 1));

    return SYSINFO_RET_OK;
}
```

Estas funciones deben devolver `SYSINFO_RET_OK` si el valor del item se obtuvo correctamente. De lo contrario, deben devolver `SYSINFO_RET_FAIL`. Consulte el ejemplo del módulo "dummy" a continuación para obtener detalles sobre cómo obtener información de `AGENT_REQUEST` y cómo establecer información en `AGENT_RESULT`.

Proporcionar callbacks de exportación de history

> **Attention:**
> La exportación de history mediante módulo ya no es compatible con Zabbix proxy.

El módulo puede especificar funciones para exportar datos de history por tipo: Numeric (float), Numeric (unsigned), Character, Text y Log:

```
typedef struct
{
    void    (*history_float_cb)(const ZBX_HISTORY_FLOAT *history, int history_num);
    void    (*history_integer_cb)(const ZBX_HISTORY_INTEGER *history, int history_num);
    void    (*history_string_cb)(const ZBX_HISTORY_STRING *history, int history_num);
    void    (*history_text_cb)(const ZBX_HISTORY_TEXT *history, int history_num);
    void    (*history_log_cb)(const ZBX_HISTORY_LOG *history, int history_num);
}
ZBX_HISTORY_WRITE_CBS;
```

Cada una de ellas debe tomar como argumentos el array `history` de `history_num` elementos. Según el tipo de datos de history que se vaya a exportar, `history` es un array de las siguientes estructuras, respectivamente:

```
typedef struct
{
    zbx_uint64_t    itemid;
    int     clock;
    int     ns;
    double      value;
}
ZBX_HISTORY_FLOAT;

typedef struct
{
    zbx_uint64_t    itemid;
    int     clock;
    int     ns;
    zbx_uint64_t    value;
}
ZBX_HISTORY_INTEGER;

typedef struct
{
    zbx_uint64_t    itemid;
    int     clock;
    int     ns;
    const char  *value;
}
ZBX_HISTORY_STRING;

typedef struct
{
    zbx_uint64_t    itemid;
    int     clock;
    int     ns;
    const char  *value;
}
ZBX_HISTORY_TEXT;

typedef struct
{
    zbx_uint64_t    itemid;
    int     clock;
    int     ns;
    const char  *value;
    const char  *source;
    int     timestamp;
    int     logeventid;
    int     severity;
}
ZBX_HISTORY_LOG;
```

Los callbacks serán utilizados por los procesos history syncer de Zabbix server al final del procedimiento de sincronización de history, después de que los datos se escriban en la base de datos de Zabbix y se guarden en la caché de valores.

Attention:
En caso de error interno en el módulo de exportación de history, se recomienda que el módulo esté escrito de forma que no bloquee toda la monitorización hasta recuperarse, sino que descarte los datos y permita que Zabbix server continúe ejecutándose.

Compilación de módulos

Actualmente, los módulos están pensados para compilarse dentro del árbol de código fuente de Zabbix, porque la API del módulo depende de algunas estructuras de datos que están definidas en los encabezados de Zabbix.

El encabezado más importante para los módulos cargables es `include/module.h`, que define estas estructuras de datos. Otros

encabezados del sistema necesarios que ayudan a que `include/module.h` funcione correctamente son `stdlib.h` y `stdint.h`.

Con esta información en mente, todo está listo para compilar el módulo. El módulo debe incluir `stdlib.h`, `stdint.h` y `module.h`, y el script de compilación debe asegurarse de que estos archivos estén en la ruta de inclusión. Consulte el ejemplo del módulo "dummy" a continuación para obtener más detalles.

Otro encabezado útil es `include/zbxcommon.h`, que define la función `zabbix_log()`, que puede utilizarse para fines de registro y depuración.

Parámetros de configuración

El agent, el server y el proxy de Zabbix admiten dos parámetros para trabajar con módulos:

- `LoadModulePath`: ruta completa a la ubicación de los módulos cargables
- `LoadModule`: módulo o módulos que se cargarán durante el inicio.

Los módulos deben encontrarse en el directorio especificado por `LoadModulePath` o la ruta debe preceder al nombre del módulo. Si la ruta precedente es absoluta (comienza por `/`), `LoadModulePath` se ignora. Se permiten varios parámetros `LoadModule`.

Por ejemplo, para ampliar el agent de Zabbix, podemos añadir los siguientes parámetros:

```
LoadModulePath=/usr/local/lib/zabbix/agent/
LoadModule=mariadb.so
LoadModule=apache.so
LoadModule=kernel.so
LoadModule=/usr/local/lib/zabbix/dummy.so
```

Durante el inicio del agent, se cargarán los módulos mariadb.so, apache.so y kernel.so desde el directorio `/usr/local/lib/zabbix/agent`, mientras que dummy.so se cargará desde `/usr/local/lib/zabbix`. El agent no podrá iniciarse si falta un módulo, si los permisos son incorrectos o si una biblioteca compartida no es un módulo de Zabbix.

Configuración del frontend

Los módulos cargables son compatibles con Zabbix agent, server y proxy. Por lo tanto, el tipo de item en el frontend de Zabbix depende de dónde se cargue el módulo. Si el módulo se carga en el agent, entonces el tipo de item debe ser "Zabbix agent" o "Zabbix agent (active)". Si el módulo se carga en server o proxy, entonces el tipo de item debe ser "Simple check".

La exportación del historial a través de módulos de Zabbix no necesita ninguna configuración del frontend. Si el módulo se carga correctamente en server y proporciona la función `zbx_module_history_write_cbs()`, que devuelve al menos una función de devolución de llamada no NULL, entonces la exportación del historial se habilitará automáticamente.

Módulo dummy

Zabbix incluye un módulo de ejemplo escrito en lenguaje C. El módulo se encuentra en `src/modules/dummy`:

```
alex@alex:~trunk/src/modules/dummy$ ls -l
-rw-rw-r-- 1 alex alex 9019 Apr 24 17:54 dummy.c
-rw-rw-r-- 1 alex alex   67 Apr 24 17:54 Makefile
-rw-rw-r-- 1 alex alex  245 Apr 24 17:54 README
```

El módulo está bien documentado y puede usarse como template para sus propios módulos.

Después de ejecutar `./configure` en la raíz del árbol de código fuente de Zabbix, como se describe arriba, solo tiene que ejecutar `make` para compilar `dummy.so`.

```
/*
** Zabbix
** Copyright (C) 2001-2020 Zabbix SIA
**
** This program is free software; you can redistribute it and/or modify
** it under the terms of the GNU General Public License as published by
** the Free Software Foundation; either version 2 of the License, or
** (at your option) any later version.
**
** This program is distributed in the hope that it will be useful,
** but WITHOUT ANY WARRANTY; without even the implied warranty of
** MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
** GNU General Public License for more details.
**
** You should have received a copy of the GNU General Public License
** along with this program; if not, write to the Free Software
```

```
** Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA  02110-1301, USA.
**/

###include <stdlib.h>
###include <string.h>
###include <time.h>
###include <stdint.h>

###include "module.h"

/* la variable mantiene la configuración de tiempo de espera para el procesamiento de item */
static int  item_timeout = 0;

/* el módulo DEBE definir las funciones internas como static y usar un patrón de nombres diferente de los
/* internos de Zabbix (zbx_*) y de las funciones de la API de módulos cargables (zbx_module_*) para evitar
static int  dummy_ping(AGENT_REQUEST *request, AGENT_RESULT *result);
static int  dummy_echo(AGENT_REQUEST *request, AGENT_RESULT *result);
static int  dummy_random(AGENT_REQUEST *request, AGENT_RESULT *result);

static ZBX_METRIC keys[] =
/*  KEY         FLAG        FUNCTION    TEST PARAMETERS */
{
    {"dummy.ping",      0,      dummy_ping, NULL},
    {"dummy.echo",      CF_HAVEPARAMS,  dummy_echo, "a message"},
    {"dummy.random",    CF_HAVEPARAMS,  dummy_random,   "1,1000"},
    {NULL}
};

/******************************************************************************
 *                                                                            *
 * Function: zbx_module_api_version                                           *
 *                                                                            *
 * Purpose: returns version number of the module interface                    *
 *                                                                            *
 * Return value: ZBX_MODULE_API_VERSION - version of module.h module is       *
 *               compiled with, in order to load module successfully Zabbix   *
 *               MUST be compiled with the same version of this header file   *
 *                                                                            *
 ******************************************************************************/
int zbx_module_api_version(void)
{
    return ZBX_MODULE_API_VERSION;
}

/******************************************************************************
 *                                                                            *
 * Function: zbx_module_item_timeout                                          *
 *                                                                            *
 * Purpose: set timeout value for processing of items                         *
 *                                                                            *
 * Parameters: timeout - timeout in seconds, 0 - no timeout set               *
 *                                                                            *
 ******************************************************************************/
void    zbx_module_item_timeout(int timeout)
{
    item_timeout = timeout;
}

/******************************************************************************
 *                                                                            *
 * Function: zbx_module_item_list                                             *
 *                                                                            *
```

```
 * Purpose: returns list of item keys supported by the module                *
 *                                                                            *
 * Return value: list of item keys                                            *
 *                                                                            *
 ******************************************************************************/
ZBX_METRIC  *zbx_module_item_list(void)
{
    return keys;
}

static int  dummy_ping(AGENT_REQUEST *request, AGENT_RESULT *result)
{
    SET_UI64_RESULT(result, 1);

    return SYSINFO_RET_OK;
}

static int  dummy_echo(AGENT_REQUEST *request, AGENT_RESULT *result)
{
    char    *param;

    if (1 != request->nparam)
    {
        /* set optional error message */
        SET_MSG_RESULT(result, strdup("Invalid number of parameters."));
        return SYSINFO_RET_FAIL;
    }

    param = get_rparam(request, 0);

    SET_STR_RESULT(result, strdup(param));

    return SYSINFO_RET_OK;
}

/******************************************************************************
 *                                                                            *
 * Function: dummy_random                                                     *
 *                                                                            *
 * Purpose: a main entry point for processing of an item                      *
 *                                                                            *
 * Parameters: request - structure that contains item key and parameters      *
 *              request->key - item key without parameters                    *
 *              request->nparam - number of parameters                        *
 *              request->params[N-1] - pointers to item key parameters        *
 *              request->types[N-1] - item key parameters types:              *
 *                  REQUEST_PARAMETER_TYPE_UNDEFINED (key parameter is empty) *
 *                  REQUEST_PARAMETER_TYPE_ARRAY (array)                      *
 *                  REQUEST_PARAMETER_TYPE_STRING (quoted or unquoted string) *
 *                                                                            *
 *             result - structure that will contain result                    *
 *                                                                            *
 * Return value: SYSINFO_RET_FAIL - function failed, item will be marked      *
 *                                  as not supported by zabbix                *
 *               SYSINFO_RET_OK - success                                     *
 *                                                                            *
 * Comment: get_rparam(request, N-1) can be used to get a pointer to the Nth  *
 *          parameter starting from 0 (first parameter). Make sure it exists  *
 *          by checking value of request->nparam.                             *
 *          In the same manner get_rparam_type(request, N-1) can be used to   *
 *          get a parameter type.                                             *
 *                                                                            *
```

```
 ******************************************************************************/
static int  dummy_random(AGENT_REQUEST *request, AGENT_RESULT *result)
{
    char    *param1, *param2;
    int from, to;

    if (2 != request->nparam)
    {
        /* set optional error message */
        SET_MSG_RESULT(result, strdup("Invalid number of parameters."));
        return SYSINFO_RET_FAIL;
    }

    param1 = get_rparam(request, 0);
    param2 = get_rparam(request, 1);

    /* there is no strict validation of parameters and types for simplicity sake */
    from = atoi(param1);
    to = atoi(param2);

    if (from > to)
    {
        SET_MSG_RESULT(result, strdup("Invalid range specified."));
        return SYSINFO_RET_FAIL;
    }

    SET_UI64_RESULT(result, from + rand() % (to - from + 1));

    return SYSINFO_RET_OK;
}

/******************************************************************************
 *                                                                            *
 * Function: zbx_module_init                                                  *
 *                                                                            *
 * Purpose: the function is called on agent startup                           *
 *          It should be used to call any initialization routines             *
 *                                                                            *
 * Return value: ZBX_MODULE_OK - success                                      *
 *               ZBX_MODULE_FAIL - module initialization failed               *
 *                                                                            *
 * Comment: the module won't be loaded in case of ZBX_MODULE_FAIL             *
 *                                                                            *
 ******************************************************************************/
int zbx_module_init(void)
{
    /* initialization for dummy.random */
    srand(time(NULL));

    return ZBX_MODULE_OK;
}

/******************************************************************************
 *                                                                            *
 * Function: zbx_module_uninit                                                *
 *                                                                            *
 * Purpose: the function is called on agent shutdown                          *
 *          It should be used to cleanup used resources if there are any      *
 *                                                                            *
 * Return value: ZBX_MODULE_OK - success                                      *
 *               ZBX_MODULE_FAIL - function failed                            *
 *                                                                            *
```

```
 ******************************************************************************/
int zbx_module_uninit(void)
{
    return ZBX_MODULE_OK;
}

/******************************************************************************
 *                                                                            *
 * Functions: dummy_history_float_cb                                          *
 *            dummy_history_integer_cb                                        *
 *            dummy_history_string_cb                                         *
 *            dummy_history_text_cb                                           *
 *            dummy_history_log_cb                                            *
 *                                                                            *
 * Purpose: callback functions for storing historical data of types float,    *
 *          integer, string, text and log respectively in external storage    *
 *                                                                            *
 * Parameters: history     - array of historical data                         *
 *             history_num - number of elements in history array              *
 *                                                                            *
 ******************************************************************************/
static void dummy_history_float_cb(const ZBX_HISTORY_FLOAT *history, int history_num)
{
    int i;

    for (i = 0; i < history_num; i++)
    {
        /* do something with history[i].itemid, history[i].clock, history[i].ns, history[i].value, ... */
    }
}

static void dummy_history_integer_cb(const ZBX_HISTORY_INTEGER *history, int history_num)
{
    int i;

    for (i = 0; i < history_num; i++)
    {
        /* do something with history[i].itemid, history[i].clock, history[i].ns, history[i].value, ... */
    }
}

static void dummy_history_string_cb(const ZBX_HISTORY_STRING *history, int history_num)
{
    int i;

    for (i = 0; i < history_num; i++)
    {
        /* do something with history[i].itemid, history[i].clock, history[i].ns, history[i].value, ... */
    }
}

static void dummy_history_text_cb(const ZBX_HISTORY_TEXT *history, int history_num)
{
    int i;

    for (i = 0; i < history_num; i++)
    {
        /* do something with history[i].itemid, history[i].clock, history[i].ns, history[i].value, ... */
    }
}

static void dummy_history_log_cb(const ZBX_HISTORY_LOG *history, int history_num)
```

```
{
    int i;

    for (i = 0; i < history_num; i++)
    {
        /* do something with history[i].itemid, history[i].clock, history[i].ns, history[i].value, ... */
    }
}

/******************************************************************************
 *                                                                            *
 * Function: zbx_module_history_write_cbs                                     *
 *                                                                            *
 * Purpose: returns a set of module functions Zabbix will call to export      *
 *          different types of historical data                                *
 *                                                                            *
 * Return value: structure with callback function pointers (can be NULL if    *
 *               module is not interested in data of certain types)           *
 *                                                                            *
 ******************************************************************************/
ZBX_HISTORY_WRITE_CBS   zbx_module_history_write_cbs(void)
{
    static ZBX_HISTORY_WRITE_CBS    dummy_callbacks =
    {
        dummy_history_float_cb,
        dummy_history_integer_cb,
        dummy_history_string_cb,
        dummy_history_text_cb,
        dummy_history_log_cb,
    };

    return dummy_callbacks;
}
```

El módulo exporta tres nuevos items:

- `dummy.ping` - siempre devuelve '1'
- `dummy.echo[param1]` - devuelve el primer parámetro tal como está; por ejemplo, `dummy.echo[ABC]` devolverá ABC
- `dummy.random[param1, param2]` - devuelve un número aleatorio dentro del rango de param1-param2; por ejemplo, `dummy.random[1,1000000]`

Limitaciones

La compatibilidad con módulos cargables está implementada solo para la plataforma Unix. Esto significa que no funciona para los agents de Windows.

En algunos casos, un módulo puede necesitar leer parámetros de configuración relacionados con el módulo desde `zabbix_agentd.conf`. Actualmente no se admite. Si necesita que su módulo use algunos parámetros de configuración, probablemente debería implementar el análisis de un archivo de configuración específico del módulo.

Módulos de frontend

Resumen

Es posible mejorar la funcionalidad del frontend de Zabbix añadiendo módulos de terceros o desarrollando sus propios módulos sin necesidad de cambiar el código fuente de Zabbix.

Tenga en cuenta que el código del módulo se ejecutará con los mismos privilegios que el código fuente de Zabbix. Esto significa:

- Los módulos de terceros pueden ser perjudiciales. Debe confiar en los módulos que instala.
- Los errores en el código de un módulo de terceros pueden bloquear el frontend. Si esto ocurre, simplemente elimine el código del módulo del frontend. En cuanto recargue el frontend de Zabbix, verá una nota indicando que faltan algunos módulos. Vaya a Administración de módulos (en *Administración* > *General* > *Módulos*) y haga clic de nuevo en *Explorar directorio* para eliminar de la base de datos los módulos inexistentes.

Instalación

Lea siempre el manual de instalación de un módulo concreto. Se recomienda instalar los módulos nuevos uno por uno para detectar fallos con facilidad.

Justo antes de instalar un módulo:

- Asegúrese de haber descargado el módulo desde una fuente de confianza. La instalación de código malicioso puede provocar consecuencias, como la pérdida de datos.
- Se pueden instalar en paralelo distintas versiones del mismo módulo (mismo ID), pero solo se puede habilitar una única versión a la vez.

Pasos para instalar un módulo:

- Desempaquete su módulo en su propia carpeta dentro del directorio `modules` de la instalación de frontend de Zabbix (por ejemplo, `zabbix/ui/modules`).
- Asegúrese de que la carpeta del módulo contenga al menos el archivo `manifest.json`.
- Vaya a Administración de módulos y haga clic en el botón *Scan directory*.
- El nuevo módulo aparecerá en la lista junto con su versión, autor, descripción y estado.
- Habilite el módulo haciendo clic en su estado.

Solución de problemas:

Problema	Solución
El módulo no apareció en la lista	Si su módulo no apareció en la lista, asegúrese de que `manifest.json` exista en `modules/your-module/`. Si no existe, es posible que haya desempaquetado el módulo en el directorio incorrecto. Si sí existe, es posible que el módulo no sea compatible con su versión de Zabbix. Además, compruebe que el usuario del servidor web tenga al menos permisos de lectura y búsqueda (`r-x`) sobre la carpeta del módulo y todos sus subdirectorios, y permisos de lectura (`r--`) sobre todos los archivos que contiene.
frontend se bloqueó	El código del módulo no es compatible con la versión actual de Zabbix o con la configuración del server. Elimine los archivos del módulo y recargue el frontend. Verá un aviso indicando que faltan algunos módulos. Vaya a Administración de módulos y haga clic de nuevo en *Scan directory* para eliminar de la base de datos los módulos inexistentes.
Aparece un mensaje de error sobre un espacio de nombres, ID o acciones idénticos	El nuevo módulo intentó registrar un espacio de nombres, ID o acciones que ya están registrados por otros módulos habilitados. Deshabilite el módulo en conflicto (mencionado en el mensaje de error) antes de habilitar el nuevo.
Aparecen mensajes de error técnicos	Informe de los errores al desarrollador del módulo.

Desarrollo de módulos

Para obtener información sobre el desarrollo de módulos personalizados, consulte el Centro de desarrolladores.

Apéndices

Resumen Esta sección contiene materiales de referencia, parámetros de configuración, especificaciones técnicas y guías de procedimientos que complementan el manual principal.

Referencia esencial

- Instalación y configuración - creación de la base de datos, instalación y configuración del server y el agent.
- Configuración de procesos - parámetros de configuración para Zabbix server, proxy, agent y más.
- Items - referencia de claves de item y tipos de datos compatibles.
- Macros - sintaxis y uso de macros integradas y de usuario.

Instalación y configuración

Esta sección abarca la creación de la base de datos, la configuración de seguridad, los procedimientos de configuración específicos de la plataforma y la configuración de funciones.

Utilice la barra lateral para encontrar el procedimiento relevante para su entorno.

Creación de la base de datos

Descripción general

Se debe crear una base de datos de Zabbix durante la instalación de Zabbix server o proxy.

Esta sección proporciona instrucciones para crear una base de datos de Zabbix. Hay un conjunto de instrucciones independiente para cada base de datos compatible.

> **Note:**
> Para mejorar la seguridad de la base de datos mediante la creación de roles/usuarios de base de datos con privilegios mínimos, consulte las mejores prácticas para la creación de bases de datos de cada base de datos compatible:

> - MySQL/MariaDB
> - PostgreSQL/TimescaleDB
>
> Para configurar conexiones TLS seguras, consulte Secure connection to the database.

UTF-8 es la única codificación compatible con Zabbix. Se sabe que funciona sin fallos de seguridad. Los usuarios deben tener en cuenta que existen problemas de seguridad conocidos si se utilizan algunas de las otras codificaciones. Para cambiar a UTF-8, consulte Repairing Zabbix database character set and collation. Consulte también Limits of filtering with utf8mb4 collations.

> **Note:**
> Si instala desde el repositorio Git de Zabbix, debe ejecutar el siguiente comando antes de continuar con los siguientes pasos:

 `make dbschema`

MySQL/MariaDB

El juego de caracteres utf8mb4 con la intercalación utf8mb4_bin es compatible para que Zabbix server/proxy funcione correctamente con bases de datos MySQL/MariaDB.
 Para evitar problemas futuros, se recomienda encarecidamente utilizar el juego de caracteres utf8mb4 con la intercalación utf8mb4_bin tanto en instalaciones nuevas como existentes. Para convertir bases de datos que utilizan el juego de caracteres utf8mb3 y la intercalación utf8mb3_bin al juego de caracteres utf8mb4 y la intercalación utf8mb4_bin, siga las instrucciones de Reparación del juego de caracteres y la intercalación de la base de datos de Zabbix.

Los triggers deterministas deben crearse durante la importación del esquema. En MySQL y MariaDB, esto requiere establecer `GLOBAL log_bin_trust_function_creators = 1` si el registro binario está habilitado, no se dispone de privilegios de superusuario y `log_bin_trust_function_creators = 1` no está establecido en el archivo de configuración de MySQL.

Si está instalando Zabbix desde paquetes o contenedores, consulte las instrucciones correspondientes.

Si está instalando Zabbix desde el código fuente:

- Cree y configure una base de datos y un usuario. Por motivos de seguridad, utilice una contraseña segura y única para el usuario de la base de datos de Zabbix.

```
mysql -uroot -p<password>

mysql> create database zabbix character set utf8mb4 collate utf8mb4_bin;
mysql> create user 'zabbix'@'localhost' identified by '<password>';
mysql> grant all privileges on zabbix.* to 'zabbix'@'localhost';
mysql> SET GLOBAL log_bin_trust_function_creators = 1;
mysql> quit;
```

- Importe los datos en la base de datos y establezca el juego de caracteres utf8mb4 como predeterminado (suponiendo que se encuentra en el directorio raíz del código fuente de Zabbix). Para una base de datos de Zabbix proxy, solo debe importarse `schema.sql` (no `images.sql` ni `data.sql`).

```
cd database/mysql
mysql -uzabbix -p<password> zabbix < schema.sql
#### deténgase aquí si está creando una base de datos para Zabbix proxy
```

```
mysql -uzabbix -p<password> zabbix < images.sql
mysql -uzabbix -p<password> --default-character-set=utf8mb4 zabbix < data.sql
```

log_bin_trust_function_creators puede deshabilitarse después de importar correctamente el esquema:

```
mysql -uroot -p<password>

mysql> SET GLOBAL log_bin_trust_function_creators = 0;
mysql> quit;
```

PostgreSQL

Debe tener un usuario de base de datos con permisos para crear objetos de base de datos.

Si está instalando Zabbix desde paquetes o contenedores, consulte las instrucciones correspondientes.

Si está instalando Zabbix desde el código fuente:

- Cree un usuario de base de datos.

El siguiente comando de shell creará el usuario `zabbix`. Especifique una contraseña cuando se le solicite y repítala (tenga en cuenta que primero se le puede pedir la contraseña de sudo):

```
sudo -u postgres createuser --pwprompt zabbix
```

- Cree una base de datos.

El siguiente comando de shell creará la base de datos `zabbix` (último parámetro) con el usuario creado previamente como propietario (`-O zabbix`).

```
sudo -u postgres createdb -O zabbix -E Unicode -T template0 zabbix
```

- Importe el esquema y los datos iniciales (suponiendo que se encuentra en el directorio raíz de las fuentes de Zabbix). Para una base de datos de Zabbix proxy, solo se debe importar `schema.sql` (no `images.sql` ni `data.sql`).

```
cd database/postgresql
cat schema.sql | sudo -u zabbix psql zabbix
#### deténgase aquí si está creando la base de datos para Zabbix proxy
cat images.sql | sudo -u zabbix psql zabbix
cat data.sql | sudo -u zabbix psql zabbix
```

> **Attention:**
> Los comandos anteriores se proporcionan como ejemplo y funcionarán en la mayoría de las instalaciones de GNU/Linux. Puede usar comandos diferentes según cómo esté configurado su sistema/base de datos, por ejemplo:

 `psql -U <username>`

 Si tiene algún problema al configurar la base de datos, consulte a su administrador de bases de datos.

TimescaleDB

Las instrucciones para crear y configurar TimescaleDB se proporcionan en una sección separada.

Elasticsearch y ClickHouse

Zabbix también puede almacenar datos históricos en Elasticsearch o ClickHouse. Para obtener instrucciones de configuración, consulte las páginas enlazadas.

SQLite

¡El uso de SQLite es compatible solo para el **proxy de Zabbix**!

La base de datos se creará automáticamente si no existe.

Vuelva a la sección de instalación.

Reparación del conjunto de caracteres y la intercalación de la base de datos de Zabbix

MySQL/MariaDB

Históricamente, MySQL y sus derivados usaban 'utf8' como un alias de utf8mb3, la implementación propia de MySQL de UTF8 estándar de 3 bytes, que es de 4 bytes. A partir de MySQL 8.0.28 y MariaDB 10.6.1, el conjunto de caracteres 'utf8mb3' está obsoleto y, en algún momento, su compatibilidad se eliminará, mientras que 'utf8' se convertirá en una referencia a 'utf8mb4'.

Desde Zabbix 6.0, se admite 'utf8mb4'. Para evitar problemas futuros, se recomienda encarecidamente usar 'utf8mb4'. Otra ventaja de cambiar a 'utf8mb4' es la compatibilidad con caracteres Unicode suplementarios.

Warning:
Como las versiones anteriores a Zabbix 6.0 no conocen utf8mb4, asegúrese de actualizar primero el server de Zabbix y el esquema de la base de datos a 6.0.x o posterior antes de ejecutar la conversión a utf8mb4.

1. Compruebe el conjunto de caracteres y la intercalación de la base de datos.

Por ejemplo:

```
mysql> SELECT @@character_set_database, @@collation_database;
+--------------------------+----------------------+
| @@character_set_database | @@collation_database |
+--------------------------+----------------------+
| latin2                   | latin2 _general_ci   |
+--------------------------+----------------------+
```

O:

```
mysql> SELECT @@character_set_database, @@collation_database;
+--------------------------+----------------------+
| @@character_set_database | @@collation_database |
+--------------------------+----------------------+
| utf8                     | utf8_bin             |
+--------------------------+----------------------+
```

Como vemos, el conjunto de caracteres aquí no es 'utf8mb4' y la intercalación no es 'utf8mb4_bin', así que debemos corregirlos.

2. Detenga Zabbix.

3. ¡Cree una copia de seguridad de la base de datos!

4. Corrija el conjunto de caracteres y la intercalación a nivel de base de datos:

```
alter database <your DB name> character set utf8mb4 collate utf8mb4_bin;
```

Valores corregidos:

```
mysql> SELECT @@character_set_database, @@collation_database;
+--------------------------+----------------------+
| @@character_set_database | @@collation_database |
+--------------------------+----------------------+
| utf8mb4                  | utf8mb4_bin          |
+--------------------------+----------------------+
```

5. Cargue el script para corregir el conjunto de caracteres y la intercalación a nivel de tabla y columna:

```
mysql <your DB name> < utf8mb4_convert.sql
```

6. Ejecute el script:

```
                SET @ZABBIX_DATABASE = '<your DB name>';
If MariaDB →    set innodb_strict_mode = OFF;
                CALL zbx_convert_utf8();
If MariaDB →    set innodb_strict_mode = ON;
                drop procedure zbx_convert_utf8;
```

Tenga en cuenta que se espera que 'utf8mb4' consuma ligeramente más espacio en disco.

7. Si no hay errores, quizá desee crear una copia de seguridad de la base de datos con la base de datos corregida.

8. Inicie Zabbix.

Actualización de la base de datos a claves primarias

Resumen

Esta sección proporciona instrucciones para actualizar manualmente las tablas en instalaciones existentes a claves primarias.

La actualización a claves primarias optimiza cómo se indexan y acceden los datos, lo que puede acelerar las consultas y ahorrar espacio. También mejora la gestión y sincronización de datos en configuraciones en clúster, lo que ayuda a escalar y garantiza que el sistema siga siendo fiable incluso si fallan algunos servidores.

Attention:
Las instrucciones proporcionadas en esta página están diseñadas para usuarios avanzados y pueden requerir ajustes según su configuración específica. La actualización a claves primarias puede llevar mucho tiempo y consumir muchos recursos. Asegúrese de que haya suficiente espacio libre en disco; según el tamaño de la base de datos y los datos almacenados, el proceso puede requerir hasta 2.5 veces el espacio que actualmente ocupan las tablas de history.

Las claves primarias se utilizan para todas las tablas en las instalaciones nuevas desde Zabbix 6.0.

No existe una actualización automática de la base de datos a claves primarias; sin embargo, las instalaciones existentes pueden actualizarse manualmente **después** de actualizar el server de Zabbix a 6.0 o una versión posterior.

Attention:
Desde Zabbix 7.0, la actualización de tablas a claves primarias también actualiza las tablas para usar tipos de datos de doble precisión.

 Si tiene Zabbix 7.0 (o posterior), las tablas ya usan doble precisión. Sin embargo, aún puede seguir las instrucciones de esta página para actualizar las tablas a claves primarias sin afectar a las tablas que ya usan doble precisión.

 Si tiene Zabbix 6.4 (o anterior), considere actualizar primero las tablas a doble precisión. Para obtener más información, consulte Actualización a valores numéricos de rango extendido en la documentación de Zabbix 7.0.

Hay instrucciones disponibles para:

- MariaDB/MySQL
- PostgreSQL
- TimescaleDB

Notas importantes

Para realizar la actualización de la base de datos:

1. Detenga el servidor Zabbix.

Se recomienda encarecidamente detener el servidor Zabbix durante el tiempo de la actualización. Sin embargo, si es absolutamente necesario, puede realizar la actualización mientras el servidor está en funcionamiento (solo para MySQL, MariaDB y PostgreSQL sin TimescaleDB).

2. Realice una copia de seguridad de su base de datos.
3. Instale el paquete zabbix-sql-scripts más reciente compatible con su versión de Zabbix (por ejemplo, para RHEL: `dnf install zabbix-sql-scripts`).
4. Ejecute los scripts para su base de datos.
5. Inicie el servidor Zabbix.

Warning:
Ejecute los scripts solo para la base de datos del servidor. El proxy no se beneficiará de esta actualización.

Si la base de datos utiliza particiones, contacte al administrador de la base de datos o al Soporte de Zabbix para obtener ayuda.

Los archivos CSV pueden eliminarse después de una actualización exitosa a claves primarias.

Opcionalmente, la interfaz web de Zabbix puede cambiarse a modo de mantenimiento.

MariaDB/MySQL

La exportación y la importación deben realizarse en tmux/screen para garantizar que la sesión no se interrumpa.

Véase también: Notas importantes

MySQL 8.4+ con mysqlsh

Este método se puede usar con un server de Zabbix en ejecución, pero se recomienda detener el server durante el tiempo de la actualización. MySQL Shell (*mysqlsh*) debe estar instalado y poder conectarse a la base de datos.

- Inicie sesión en la consola de MySQL como root (recomendado) o como cualquier usuario con privilegios FILE.
- Inicie MySQL con la variable local_infile habilitada.
- Cambie el nombre de las tablas antiguas y cree las nuevas ejecutando `history_upgrade_prepare.sql`.

```
mysql -uzabbix -p<password> zabbix < /usr/share/zabbix/sql-scripts/mysql/option-patches/history_upgrade_pr
```

- Exporte e importe los datos.

Conéctese mediante mysqlsh. Si usa una conexión por socket, puede ser necesario especificar la ruta.

```
sudo mysqlsh -uroot -S /run/mysqld/mysqld.sock --no-password -Dzabbix
```

Cambie al modo JavaScript usando:

```
\js
```

Luego ejecute el código siguiente (CSVPATH se puede cambiar según sea necesario):

```
CSVPATH="/var/lib/mysql-files";

util.exportTable("history_old", CSVPATH + "/history.csv", { dialect: "csv" });
util.importTable(CSVPATH + "/history.csv", {"dialect": "csv", "table": "history" });

util.exportTable("history_uint_old", CSVPATH + "/history_uint.csv", { dialect: "csv" });
util.importTable(CSVPATH + "/history_uint.csv", {"dialect": "csv", "table": "history_uint" });

util.exportTable("history_str_old", CSVPATH + "/history_str.csv", { dialect: "csv" });
util.importTable(CSVPATH + "/history_str.csv", {"dialect": "csv", "table": "history_str" });

util.exportTable("history_log_old", CSVPATH + "/history_log.csv", { dialect: "csv" });
util.importTable(CSVPATH + "/history_log.csv", {"dialect": "csv", "table": "history_log" });

util.exportTable("history_text_old", CSVPATH + "/history_text.csv", { dialect: "csv" });
util.importTable(CSVPATH + "/history_text.csv", {"dialect": "csv", "table": "history_text" });
```

Si recibe el mensaje "JavaScript is not supported", su instalación de MySQL Shell no tiene compatibilidad con JS. En ese caso, instale el paquete oficial de MySQL Shell de Oracle (o compílelo desde el código fuente) para que el modo JavaScript esté habilitado.

- Siga las instrucciones posteriores a la migración para eliminar las tablas antiguas.

MariaDB/MySQL 8.4+ sin mysqlsh

Este método de actualización lleva más tiempo y solo debe usarse si no es posible una actualización con *mysqlsh*.

Actualización de tablas

- Inicie sesión en la consola de MySQL como root (recomendado) o como cualquier usuario con privilegios FILE.
- Si realiza la migración con un server Zabbix en ejecución, inicie MySQL con la variable local_infile habilitada.
- Cambie el nombre de las tablas antiguas y cree las nuevas ejecutando `history_upgrade_prepare.sql`:

```
mysql -uzabbix -p<password> zabbix < /usr/share/zabbix/sql-scripts/mysql/option-patches/history_upgrade_pr
```

Migración con el server detenido

`max_execution_time` (en MySQL) o `max_statement_time` (en MariaDB) debe deshabilitarse antes de migrar los datos para evitar un tiempo de espera durante la migración.

Para MySQL:

```
SET @@max_execution_time=0;
```

Para MariaDB:

```
SET @@max_statement_time=0;
```

```
INSERT IGNORE INTO history SELECT * FROM history_old;
INSERT IGNORE INTO history_uint SELECT * FROM history_uint_old;
INSERT IGNORE INTO history_str SELECT * FROM history_str_old;
INSERT IGNORE INTO history_log SELECT * FROM history_log_old;
INSERT IGNORE INTO history_text SELECT * FROM history_text_old;
```

Siga las instrucciones posteriores a la migración para eliminar las tablas antiguas.

Migración con el server en ejecución

Compruebe para qué rutas está habilitada la importación/exportación:

```
mysql> SELECT @@secure_file_priv;
+-----------------------+
| @@secure_file_priv    |
+-----------------------+
| /var/lib/mysql-files/ |
+-----------------------+
```

Si el valor de *secure_file_priv* es una ruta a un directorio, la exportación/importación se realizará para los archivos de ese directorio. En este caso, edite las rutas a los archivos en las consultas en consecuencia o establezca el valor de *secure_file_priv* en una cadena vacía durante el tiempo de actualización.

Si el valor de *secure_file_priv* está vacío, la exportación/importación puede realizarse desde cualquier ubicación.

Si el valor de *secure_file_priv* es NULL, establézcalo en la ruta que contiene los datos exportados de la tabla ('/var/lib/mysql-files/' en el ejemplo anterior).

Para obtener más información, consulte la documentación de MySQL o la documentación de MariaDB.

`max_execution_time` (en MySQL) o `max_statement_time` (en MariaDB) debe deshabilitarse antes de exportar los datos para evitar un tiempo de espera durante la exportación.

Para MySQL:

```
SET @@max_execution_time=0;
```

Para MariaDB:

```
SET @@max_statement_time=0;
```

```
SELECT * INTO OUTFILE '/var/lib/mysql-files/history.csv' FIELDS TERMINATED BY ',' ESCAPED BY '"' LINES TER
LOAD DATA INFILE '/var/lib/mysql-files/history.csv' IGNORE INTO TABLE history FIELDS TERMINATED BY ',' ESC

SELECT * INTO OUTFILE '/var/lib/mysql-files/history_uint.csv' FIELDS TERMINATED BY ',' ESCAPED BY '"' LINE
LOAD DATA INFILE '/var/lib/mysql-files/history_uint.csv' IGNORE INTO TABLE history_uint FIELDS TERMINATED

SELECT * INTO OUTFILE '/var/lib/mysql-files/history_str.csv' FIELDS TERMINATED BY ',' ESCAPED BY '"' LINES
LOAD DATA INFILE '/var/lib/mysql-files/history_str.csv' IGNORE INTO TABLE history_str FIELDS TERMINATED BY

SELECT * INTO OUTFILE '/var/lib/mysql-files/history_log.csv' FIELDS TERMINATED BY ',' ESCAPED BY '"' LINES
LOAD DATA INFILE '/var/lib/mysql-files/history_log.csv' IGNORE INTO TABLE history_log FIELDS TERMINATED BY

SELECT * INTO OUTFILE '/var/lib/mysql-files/history_text.csv' FIELDS TERMINATED BY ',' ESCAPED BY '"' LINE
LOAD DATA INFILE '/var/lib/mysql-files/history_text.csv' IGNORE INTO TABLE history_text FIELDS TERMINATED
```

Siga las instrucciones posteriores a la migración para eliminar las tablas antiguas.

PostgreSQL

La exportación y la importación deben realizarse en tmux/screen para garantizar que la sesión no se interrumpa. Para instalaciones con TimescaleDB, omita esta sección y continúe con PostgreSQL + TimescaleDB.

Véase también: Notas importantes

Actualización de tablas

- Renombre las tablas usando `history_upgrade_prepare.sql`:

```
sudo -u zabbix psql zabbix < /usr/share/zabbix/sql-scripts/postgresql/option-patches/history_upgrade_prepa
```

Migración con el server detenido

- Exporte el historial actual, impórtelo en la tabla temporal y luego inserte los datos en las nuevas tablas ignorando los duplicados:

```
INSERT INTO history SELECT * FROM history_old ON CONFLICT (itemid,clock,ns) DO NOTHING;

INSERT INTO history_uint SELECT * FROM history_uint_old ON CONFLICT (itemid,clock,ns) DO NOTHING;

INSERT INTO history_str SELECT * FROM history_str_old ON CONFLICT (itemid,clock,ns) DO NOTHING;

INSERT INTO history_log SELECT * FROM history_log_old ON CONFLICT (itemid,clock,ns) DO NOTHING;
```

```
INSERT INTO history_text SELECT * FROM history_text_old ON CONFLICT (itemid,clock,ns) DO NOTHING;
```

Consulte consejos para mejorar el rendimiento de INSERT: PostgreSQL: Bulk Loading Huge Amounts of Data, Checkpoint Distance and Amount of WAL.

- Siga las instrucciones posteriores a la migración para eliminar las tablas antiguas.

Migración con server en ejecución

- Exporte el historial actual, impórtelo en la tabla temporal y luego inserte los datos en las nuevas tablas mientras ignora los duplicados:

```
\copy history_old TO '/tmp/history.csv' DELIMITER ',' CSV
CREATE TEMP TABLE temp_history (
    itemid                   bigint                                    NOT NULL,
    clock                    integer         DEFAULT '0'               NOT NULL,
    value                    DOUBLE PRECISION DEFAULT '0.0000'          NOT NULL,
    ns                       integer         DEFAULT '0'               NOT NULL
);
\copy temp_history FROM '/tmp/history.csv' DELIMITER ',' CSV
INSERT INTO history SELECT * FROM temp_history ON CONFLICT (itemid,clock,ns) DO NOTHING;

\copy history_uint_old TO '/tmp/history_uint.csv' DELIMITER ',' CSV
CREATE TEMP TABLE temp_history_uint (
    itemid                   bigint                                    NOT NULL,
    clock                    integer         DEFAULT '0'               NOT NULL,
    value                    numeric(20)     DEFAULT '0'               NOT NULL,
    ns                       integer         DEFAULT '0'               NOT NULL
);
\copy temp_history_uint FROM '/tmp/history_uint.csv' DELIMITER ',' CSV
INSERT INTO history_uint SELECT * FROM temp_history_uint ON CONFLICT (itemid,clock,ns) DO NOTHING;

\copy history_str_old TO '/tmp/history_str.csv' DELIMITER ',' CSV
CREATE TEMP TABLE temp_history_str (
    itemid                   bigint                                    NOT NULL,
    clock                    integer         DEFAULT '0'               NOT NULL,
    value                    varchar(255)    DEFAULT ''                NOT NULL,
    ns                       integer         DEFAULT '0'               NOT NULL
);
\copy temp_history_str FROM '/tmp/history_str.csv' DELIMITER ',' CSV
INSERT INTO history_str (itemid,clock,value,ns) SELECT * FROM temp_history_str ON CONFLICT (itemid,clock,n

\copy history_log_old TO '/tmp/history_log.csv' DELIMITER ',' CSV
CREATE TEMP TABLE temp_history_log (
    itemid                   bigint                                    NOT NULL,
    clock                    integer         DEFAULT '0'               NOT NULL,
    timestamp                integer         DEFAULT '0'               NOT NULL,
    source                   varchar(64)     DEFAULT ''                NOT NULL,
    severity                 integer         DEFAULT '0'               NOT NULL,
    value                    text            DEFAULT ''                NOT NULL,
    logeventid               integer         DEFAULT '0'               NOT NULL,
    ns                       integer         DEFAULT '0'               NOT NULL
);
\copy temp_history_log FROM '/tmp/history_log.csv' DELIMITER ',' CSV
INSERT INTO history_log SELECT * FROM temp_history_log ON CONFLICT (itemid,clock,ns) DO NOTHING;

\copy history_text_old TO '/tmp/history_text.csv' DELIMITER ',' CSV
CREATE TEMP TABLE temp_history_text (
    itemid                   bigint                                    NOT NULL,
    clock                    integer         DEFAULT '0'               NOT NULL,
    value                    text            DEFAULT ''                NOT NULL,
    ns                       integer         DEFAULT '0'               NOT NULL
);
\copy temp_history_text FROM '/tmp/history_text.csv' DELIMITER ',' CSV
```

```
INSERT INTO history_text SELECT * FROM temp_history_text ON CONFLICT (itemid,clock,ns) DO NOTHING;
```

- Siga las instrucciones posteriores a la migración para eliminar las tablas antiguas.

PostgreSQL + TimescaleDB

La exportación y la importación deben realizarse en tmux/screen para garantizar que la sesión no se interrumpa. Zabbix server debe estar detenido durante la actualización.

Véase también: Notas importantes

- Renombre las tablas usando `history_upgrade_prepare.sql`.
  - Si la compresión está habilitada (en la instalación predeterminada), ejecute el script desde `/usr/share/zabbix/sql-scripts/`

    ```
    cat /usr/share/zabbix/sql-scripts/postgresql/timescaledb/option-patches/with-compression/history_
    ```

  - Si la compresión está deshabilitada, ejecute el script desde `/usr/share/zabbix/sql-scripts/postgresql/timescaledb/`

    ```
    cat /usr/share/zabbix/sql-scripts/postgresql/timescaledb/option-patches/without-compression/histo
    ```

- Ejecute los scripts de migración de hypertable de TimescaleDB según la configuración de compresión:
  - Si la compresión está habilitada (en la instalación predeterminada), ejecute los scripts desde `/usr/share/zabbix/sql-scripts`

    ```
    cat /usr/share/zabbix/sql-scripts/postgresql/timescaledb/option-patches/with-compression/history_
    cat /usr/share/zabbix/sql-scripts/postgresql/timescaledb/option-patches/with-compression/history_
    cat /usr/share/zabbix/sql-scripts/postgresql/timescaledb/option-patches/with-compression/history_
    cat /usr/share/zabbix/sql-scripts/postgresql/timescaledb/option-patches/with-compression/history_
    cat /usr/share/zabbix/sql-scripts/postgresql/timescaledb/option-patches/with-compression/history_
    cat /usr/share/zabbix/sql-scripts/postgresql/timescaledb/option-patches/with-compression/trends_u
    ```

  - Si la compresión está deshabilitada, ejecute los scripts desde `/usr/share/zabbix/sql-scripts/postgresql/timescaledb`

    ```
    cat /usr/share/zabbix/sql-scripts/postgresql/timescaledb/option-patches/without-compression/histo
    cat /usr/share/zabbix/sql-scripts/postgresql/timescaledb/option-patches/without-compression/histo
    cat /usr/share/zabbix/sql-scripts/postgresql/timescaledb/option-patches/without-compression/histo
    cat /usr/share/zabbix/sql-scripts/postgresql/timescaledb/option-patches/without-compression/histo
    cat /usr/share/zabbix/sql-scripts/postgresql/timescaledb/option-patches/without-compression/histo
    cat /usr/share/zabbix/sql-scripts/postgresql/timescaledb/option-patches/without-compression/trend
    ```

Véase también: Consejos para mejorar el rendimiento de INSERT.

- Siga las instrucciones posteriores a la migración para eliminar las tablas antiguas.

Post-migración

Para todas las bases de datos, una vez completada la migración, haga lo siguiente:

- Verifique que todo funcione como se espera.
- Elimine las tablas antiguas:

```
DROP TABLE history_old;
DROP TABLE history_uint_old;
DROP TABLE history_str_old;
DROP TABLE history_log_old;
DROP TABLE history_text_old;
```

- Para TimescaleDB, elimine también la siguiente tabla antigua:

```
DROP TABLE trends_old;
```

Véase también

- Preparando la tabla auditlog para particionamiento

Preparando la tabla auditlog para la partición

Descripción general

Algunas bases de datos (por ejemplo, MySQL) requieren que la columna de partición forme parte de la restricción única de la tabla. Por lo tanto, para particionar la tabla `auditlog` por tiempo, la clave primaria debe cambiarse de `auditid` a una clave compuesta `auditid` + `clock`.

Esta sección proporciona instrucciones para modificar la clave primaria de la tabla `auditlog`.

Attention:
Las instrucciones proporcionadas en esta página están diseñadas para usuarios avanzados. Tenga en cuenta que estas instrucciones pueden necesitar ser ajustadas para su configuración específica. Modificar la clave primaria también puede ser incompatible con futuros parches de actualización, por lo que puede ser necesario gestionar manualmente futuras actualizaciones.

 Modificar la clave primaria puede ser una operación que consume muchos recursos y que lleva mucho tiempo dependiendo del tamaño de la tabla `auditlog`. Se recomienda detener el servidor Zabbix y cambiar el frontend de Zabbix a modo de mantenimiento durante el tiempo que dure la modificación. Sin embargo, si es absolutamente necesario, existe una forma de modificar la clave primaria sin tiempo de inactividad (ver más abajo).

La partición de la tabla `auditlog` puede mejorar, por ejemplo, el mantenimiento en instalaciones grandes. Aunque actualmente el mantenimiento de Zabbix no puede aprovechar las tablas particionadas (excepto para TimescaleDB), puede desactivar el mantenimiento de Zabbix y eliminar particiones utilizando scripts.

Desde Zabbix 7.0, la tabla `auditlog` para TimescaleDB se ha convertido en una hypertable, lo que permite al housekeeper eliminar datos por bloques. Para actualizar la tabla `auditlog` existente a una hypertable, consulte Actualización del esquema de TimescaleDB.

MySQL

Notas importantes sobre la reconstrucción de índices

MySQL reconstruye automáticamente los índices de la clave primaria durante la operación `ALTER TABLE`. Sin embargo, se recomienda encarecidamente reconstruir también manualmente los índices con la instrucción `OPTIMIZE TABLE` para garantizar un rendimiento óptimo de la base de datos.

La reconstrucción de índices puede requerir temporalmente tanto espacio adicional en disco como el que utiliza la propia tabla. Para obtener el tamaño actual de los datos y los índices, puede ejecutar las siguientes instrucciones:

```
ANALYZE TABLE auditlog;
SHOW TABLE STATUS LIKE 'auditlog';
```

Si el espacio disponible en disco es una preocupación, siga las instrucciones de Alterar la clave primaria sin tiempo de inactividad. También hay otras opciones disponibles:

- Aumentar el parámetro MySQL `sort_buffer_size` puede ayudar a reducir el uso de espacio en disco al reconstruir manualmente los índices. Sin embargo, modificar esta variable puede afectar al uso general de memoria de la base de datos.
- Considere liberar espacio eliminando datos potencialmente innecesarios.
- Considere reducir el parámetro *Período de almacenamiento de datos* de housekeeper antes de ejecutar el housekeeper.

Alteración de la clave primaria con tiempo de inactividad

1. Elimine la clave primaria actual de la tabla `auditlog` y agregue la nueva clave primaria.

```
ALTER TABLE auditlog DROP PRIMARY KEY, ADD PRIMARY KEY (auditid, clock);
```

2. Reconstruya los índices (opcional, pero muy recomendable; consulte Notas importantes sobre la reconstrucción de índices).

```
OPTIMIZE TABLE auditlog;
```

Alteración de la clave primaria sin tiempo de inactividad

Aquí se describe el método manual para alterar la clave primaria. Como alternativa, puede usar el toolkit pt-online-schema-change de Percona. Este toolkit realiza automáticamente las siguientes acciones, al mismo tiempo que minimiza el espacio utilizado para alterar la tabla `auditlog`.

1. Crear una nueva tabla con la nueva clave primaria y crear índices.

```
CREATE TABLE `auditlog_new` (
  `auditid`            varchar(25)                         NOT NULL,
  `userid`             bigint unsigned                     NULL,
  `username`           varchar(100)    DEFAULT ''          NOT NULL,
  `clock`              integer         DEFAULT '0'         NOT NULL,
  `ip`                 varchar(39)     DEFAULT ''          NOT NULL,
  `action`             integer         DEFAULT '0'         NOT NULL,
  `resourcetype`       integer         DEFAULT '0'         NOT NULL,
  `resourceid`         bigint unsigned                     NULL,
  `resource_cuid`      varchar(25)                         NULL,
  `resourcename`       varchar(255)    DEFAULT ''          NOT NULL,
  `recordsetid`        varchar(25)                         NOT NULL,
  `details`            longtext                            NOT NULL,
```

```
  PRIMARY KEY (auditid,clock)
) ENGINE=InnoDB;
CREATE INDEX `auditlog_1` ON `auditlog_new` (`userid`,`clock`);
CREATE INDEX `auditlog_2` ON `auditlog_new` (`clock`);
CREATE INDEX `auditlog_3` ON `auditlog_new` (`resourcetype`,`resourceid`);
```

2. Intercambiar las tablas.

```
RENAME TABLE auditlog TO auditlog_old, auditlog_new TO auditlog;
```

3. Copiar los datos de la tabla antigua a la nueva.

```
INSERT INTO auditlog SELECT * FROM auditlog_old;
```

Esto se puede hacer en fragmentos (varias instrucciones `INSERT INTO` con cláusulas `WHERE clock` según sea necesario) para evitar un uso excesivo de recursos.

4. Eliminar la tabla antigua.

```
DROP TABLE auditlog_old;
```

PostgreSQL

Notas importantes sobre la reconstrucción de índices

PostgreSQL reconstruye automáticamente los índices de la clave primaria durante la operación `ALTER TABLE`. Sin embargo, se recomienda encarecidamente reconstruir también manualmente los índices con la instrucción `REINDEX TABLE CONCURRENTLY` para garantizar un rendimiento óptimo de la base de datos.

La reconstrucción de índices puede requerir temporalmente hasta tres veces el espacio en disco que utilizan actualmente los índices. Para obtener el tamaño actual de los índices, puede ejecutar la siguiente consulta:

```
SELECT pg_size_pretty(pg_indexes_size('auditlog'));
```

Si el espacio disponible en disco es una preocupación, siga las instrucciones de Alterar la clave primaria sin tiempo de inactividad. También hay otras opciones disponibles:

- Aumentar el parámetro `maintenance_work_mem` de PostgreSQL puede ayudar a reducir el uso de espacio en disco al reconstruir índices manualmente. Sin embargo, modificar esta variable puede afectar el uso general de memoria de la base de datos.
- Si dispone de otro disco o tablespace con más espacio disponible, puede considerar cambiar la ubicación del almacenamiento temporal para la reconstrucción del índice. Puede establecer el parámetro `temp_tablespaces` de PostgreSQL para especificar un tablespace diferente para los objetos temporales.
- Considere liberar espacio eliminando datos potencialmente innecesarios.
- Considere reducir el parámetro *Período de almacenamiento de datos* housekeeper antes de ejecutar el housekeeper.

Alteración de la clave primaria con tiempo de inactividad

1. Elimine la clave primaria actual de la tabla `auditlog` y agregue la nueva clave primaria.

```
ALTER TABLE auditlog DROP CONSTRAINT auditlog_pkey;
ALTER TABLE auditlog ADD PRIMARY KEY (auditid,clock);
```

2. Reconstruya los índices (opcional, pero muy recomendable; consulte Notas importantes sobre la reconstrucción de índices).

```
REINDEX TABLE CONCURRENTLY auditlog;
```

Alteración de la clave primaria sin tiempo de inactividad

Aquí se describe el método manual para alterar la clave primaria. Como alternativa, se puede considerar la extensión `pg_repack` para crear una nueva tabla, copiar los datos e intercambiar las tablas.

1. Cree una nueva tabla con la nueva clave primaria y cree índices.

```
CREATE TABLE auditlog_new (
  auditid             varchar(25)                          NOT NULL,
  userid              bigint                               NULL,
  username            varchar(100)    DEFAULT ''           NOT NULL,
  clock               integer         DEFAULT '0'          NOT NULL,
  ip                  varchar(39)     DEFAULT ''           NOT NULL,
  action              integer         DEFAULT '0'          NOT NULL,
  resourcetype        integer         DEFAULT '0'          NOT NULL,
```

```
  resourceid           bigint                                  NULL,
  resource_cuid        varchar(25)                             NULL,
  resourcename         varchar(255)    DEFAULT ''              NOT NULL,
  recordsetid          varchar(25)                             NOT NULL,
  details              text            DEFAULT ''              NOT NULL,
  PRIMARY KEY (auditid,clock)
);
CREATE INDEX auditlog_new_1 ON auditlog_new (userid,clock);
CREATE INDEX auditlog_new_2 ON auditlog_new (clock);
CREATE INDEX auditlog_new_3 ON auditlog_new (resourcetype,resourceid);
```

2. Intercambie las tablas.

```
ALTER TABLE auditlog RENAME TO auditlog_old;
ALTER TABLE auditlog_new RENAME TO auditlog;
```

3. Copie los datos de la tabla antigua a la nueva.

```
INSERT INTO auditlog SELECT * FROM auditlog_old;
```

Esto se puede hacer en fragmentos (varias instrucciones `INSERT INTO` con cláusulas `WHERE clock` según sea necesario) para evitar un uso excesivo de recursos.

4. Elimine la tabla antigua.

```
DROP TABLE auditlog_old;
```

Véase también

- Actualización de la base de datos a claves primarias

Configuración de TimescaleDB

Descripción general

Zabbix es compatible con TimescaleDB, una solución de base de datos basada en PostgreSQL que particiona automáticamente los datos en fragmentos basados en el tiempo para admitir un rendimiento más rápido a gran escala.

> **Warning:**
> Actualmente, Zabbix proxy no es compatible con TimescaleDB.

Las instrucciones en esta página pueden utilizarse para los siguientes escenarios:

- Crear una base de datos TimescaleDB o migrar desde tablas PostgreSQL existentes a TimescaleDB (consulte Configuración).
- Actualizar un esquema de base de datos TimescaleDB existente al actualizar Zabbix (consulte Actualización del esquema de TimescaleDB).

Configuración

Requisitos previos: extensión TimescaleDB de una versión compatible instalada en el server de base de datos. Para obtener instrucciones de instalación, consulte la documentación de TimescaleDB.

> **Warning:**
> Antes de instalar TimescaleDB, instale una versión compatible de PostgreSQL desde el repositorio oficial de PostgreSQL.

Habilite la extensión TimescaleDB para la base de datos específica ejecutando:

```
echo "CREATE EXTENSION IF NOT EXISTS timescaledb CASCADE;" | sudo -u postgres psql zabbix
```

La ejecución de este comando requiere privilegios de administrador de la base de datos.

> **Note:**
> Si utiliza un esquema de base de datos distinto de 'public', debe añadir una cláusula SCHEMA al comando anterior. P. ej.:
 `echo "CREATE EXTENSION IF NOT EXISTS timescaledb SCHEMA yourschema CASCADE;" | sudo -u postgres psql zabbix`

A continuación, ejecute el script `postgresql/timescaledb/schema.sql`. En instalaciones nuevas, el script debe ejecutarse después de que la base de datos PostgreSQL normal haya sido creada con el esquema/datos iniciales (consulte creación de la base de datos).

```
cat /usr/share/zabbix/sql-scripts/postgresql/timescaledb/schema.sql | sudo -u zabbix psql zabbix
```

Attention:
Ignore los mensajes de advertencia que informan de que no se siguen las mejores prácticas al ejecutar el script `schema.sql` en TimescaleDB versión 2.9.0 y posteriores. A pesar de esta advertencia, la configuración se completará correctamente.

La migración de los datos existentes de history, trends y audit log puede llevar mucho tiempo. El server de Zabbix y el frontend deben estar detenidos durante el período de migración.

El script `schema.sql` establece los siguientes parámetros de housekeeping:

- Anular el período de history del item
- Anular el período de trends del item

Para utilizar housekeeping particionado para history y trends, ambas opciones deben estar habilitadas. También es posible habilitar la anulación de forma individual, ya sea solo para history o solo para trends.

Para PostgreSQL y TimescaleDB, el script `postgresql/timescaledb/schema.sql` establece dos parámetros adicionales:

- Habilitar compresión
- Comprimir registros con más de 7 días

Para eliminar correctamente los datos comprimidos mediante el housekeeper, se deben habilitar ambas opciones: *Anular el período de historial del item* y *Anular el período de tendencias del item*. Si la anulación está deshabilitada y las tablas tienen fragmentos comprimidos, el housekeeper no eliminará los datos de estas tablas y se mostrarán advertencias sobre una configuración incorrecta en las secciones *Housekeeping* e *Información del sistema*.

Todos estos parámetros se pueden cambiar en *Administración* > *Mantenimiento* después de la instalación.

Note:
Es posible que desee ejecutar la herramienta timescaledb-tune proporcionada por TimescaleDB para optimizar los parámetros de configuración de PostgreSQL en su `postgresql.conf`.

Actualización del esquema de TimescaleDB

Al actualizar Zabbix a una versión que contiene nuevas hypertables de TimescaleDB, el server de Zabbix no configura automáticamente esas hypertables (por ejemplo, al actualizar de Zabbix 6.4 a 8.0, ya que las versiones 7.0.0 y 7.0.2 han introducido nuevas hypertables).

Para configurar las nuevas hypertables de TimescaleDB, siga estos pasos:

1. Inicie el server de Zabbix; esto actualizará la base de datos existente.
2. Compruebe en el archivo de registro del server que la actualización de la base de datos se ha completado; cuando se complete, detenga el server de Zabbix. Tenga en cuenta que el server registra una advertencia si intenta habilitar la compresión para una tabla que no es una hypertable.
3. Ejecute el script `postgresql/timescaledb/schema.sql`; esto configura las nuevas hypertables de TimescaleDB. Tenga en cuenta que, desde Zabbix 7.0.0, la ubicación y el nombre del script han cambiado de `postgresql/timescaledb.sql` a `postgresql/timescaledb/schema.sql`.

Attention:
Por favor, ignore los mensajes de advertencia que informan que no se siguen las mejores prácticas al ejecutar el script `schema.sql` en TimescaleDB versión 2.9.0 y superiores. Independientemente de esta advertencia, la configuración se completará correctamente.

Compresión de TimescaleDB

La compresión nativa de TimescaleDB es compatible con todas las tablas de Zabbix que son hypertables de TimescaleDB. Durante la actualización o migración a TimescaleDB, la compresión inicial de las tablas grandes puede llevar mucho tiempo.

Tenga en cuenta que la compresión es compatible bajo la licencia comunitaria "timescale" de Timescale y no es compatible bajo la licencia "apache" Apache 2.0. Si Zabbix detecta que la compresión no es compatible, se escribe un mensaje de advertencia en el log del server de Zabbix y los usuarios no pueden habilitar la compresión en el frontend.

Note:
Se recomienda a los usuarios familiarizarse con la compresión en la documentación de TimescaleDB antes de usarla.

Tenga en cuenta que la compresión impone ciertas limitaciones, en particular:

- No se permiten modificaciones en chunks comprimidos (inserciones, eliminaciones, actualizaciones)
- No se permiten cambios de esquema en tablas comprimidas.

Los ajustes de compresión se pueden cambiar en el bloque *Compresión de historial, tendencias y log de auditoría* en la sección *Administración > Housekeeping* del frontend de Zabbix.

Parameter	Default	Comments
Enable compression	Enabled	Marcar o desmarcar la casilla no activa/desactiva la compresión de inmediato. Como la compresión es gestionada por Housekeeper, los cambios surtirán efecto en un máximo de 2 veces `HousekeepingFrequency` horas (configurado en `zabbix_server.conf`) Después de deshabilitar la compresión, los nuevos chunks que entren en el período de compresión no se comprimirán. Sin embargo, todos los datos previamente comprimidos permanecerán comprimidos. Para descomprimir chunks previamente comprimidos, siga las instrucciones de la documentación de TimescaleDB. Al actualizar desde versiones anteriores de Zabbix con compatibilidad con TimescaleDB, la compresión no estará habilitada de forma predeterminada.
Compress records older than	7d	Este parámetro no puede ser inferior a 7 días. Debido a la inmutabilidad de los chunks comprimidos, todos los datos tardíos (por ejemplo, datos retrasados por un proxy) que sean más antiguos que este valor se descartarán.

Para mejorar el rendimiento de actualización de tendencias, puede reducir el "chunk_time_interval" de las tablas `trends` y `trends_uint` de 30 días a 7 días o menos, dependiendo de cuántos items utilicen tendencias. El propósito de este ajuste es seguir las mejores prácticas de TimescaleDB y garantizar que el tamaño del chunk se mantenga dentro de los recursos disponibles del sistema.

Configuración de Elasticsearch

Zabbix puede almacenar datos históricos en Elasticsearch como una alternativa a una base de datos relacional.

Attention:
Actualmente, el soporte de Elasticsearch es experimental.

Esta guía cubre la configuración de las versiones compatibles de Elasticsearch. Si usa una versión diferente, es posible que algunas funciones no trabajen como se espera.

Elasticsearch puede almacenar los siguientes tipos de valores:

Item value type	Database table	Elasticsearch type
Numeric (unsigned)	history_uint	uint
Numeric (float)	history	dbl
Character	history_str	str
Log	history_log	log
Text	history_text	text
Binary	history_bin	not supported by Zabbix
JSON	history_json	json

Note:
Elasticsearch no acepta matrices JSON. Un valor JSON debe ser un único objeto o un conjunto de objetos.

Notas importantes

- Elasticsearch requiere libcurl. Consulte requirements para obtener más detalles.
- El housekeeper no elimina datos de Elasticsearch. Para controlar durante cuánto tiempo se conservan los datos, consulte Index Lifecycle Management.
- Zabbix **no** calcula ni almacena tendencias en Elasticsearch. Considere ampliar el periodo de almacenamiento del historial para conservar datos más antiguos.
- Cuando se usa Elasticsearch, las consultas de rango que recuperan valores de la base de datos están limitadas por la marca de tiempo del periodo de almacenamiento de datos.
- Elasticsearch no es compatible con los proxies de Zabbix.

Si Elasticsearch aún no está instalado, consulte la guía oficial de instalación antes de continuar.

Configuración de Elasticsearch

Para almacenar datos históricos en Elasticsearch, debe:

- Crear un índice para cada tipo de valor que desee almacenar; aquí es donde Elasticsearch almacena los datos, similar a una tabla en una base de datos relacional.
- Definir un mapeo para cada índice; esto define la estructura de los datos, similar al esquema de una tabla.
- Configurar una ingest pipeline para procesar los valores antes del almacenamiento (requerido para valores JSON e índices basados en fechas).

Elasticsearch puede almacenar datos en un único índice por tipo de valor o en varios índices basados en fechas. Ambos enfoques se describen a continuación.

Almacenamiento del historial en un único índice

En este enfoque, todos los datos históricos de un tipo de valor determinado se escriben en un único índice (por ejemplo, `uint` o `text`).

Para crear un índice para el tipo de valor *Numérico (sin signo)*, envíe la siguiente solicitud (con `/uint` en la URL) a su instancia de Elasticsearch:

```
curl -X PUT \
 http://localhost:9200/uint \
 -H 'content-type:application/json' \
 -d '{
     "settings": {
      "index": {
         "number_of_replicas": 1,
         "number_of_shards": 5
      }
   },
   "mappings": {
      "properties": {
         "itemid": { "type": "long" },
         "clock": { "format": "epoch_second", "type": "date" },
         "value": { "type": "long" }
      }
   }
}'
```

Elasticsearch responderá con una confirmación de que el índice fue creado:

```
{"acknowledged": true, "shards_acknowledged": true, "index": "uint"}
```

Se deben enviar solicitudes similares para cada tipo de valor adicional que desee almacenar en Elasticsearch.

Note:
Las asignaciones para todos los tipos de valor están disponibles en el repositorio fuente de Zabbix.

Por ejemplo, para crear un índice para el tipo de valor *Texto*:

```
curl -X PUT \
 http://localhost:9200/text \
 -H 'content-type:application/json' \
 -d '{
   "settings": {
      "index": {
         "number_of_replicas": 1,
         "number_of_shards": 5
      }
   },
   "mappings": {
      "properties": {
         "itemid": { "type": "long" },
         "clock": { "format": "epoch_second", "type": "date" },
         "value": {
            "fields": {
               "analyzed": { "index": true, "type": "text", "analyzer": "standard" }
            },
            "index": false,
            "type": "text"
         }
      }
   }
}'
```

Tipo de valor JSON

A diferencia de otros tipos de valores, los valores JSON requieren un procesamiento adicional antes de su almacenamiento.

El índice a continuación utiliza campos separados para los valores analizados y sin procesar, por lo que se necesita un ingest pipeline para analizar cada valor como JSON y almacenarlo en el campo correcto.

Para crear un índice para el tipo de valor *JSON*, envíe la siguiente solicitud (con `/json` en la URL) a su instancia de Elasticsearch.

```
curl -X PUT \
 http://localhost:9200/json \
 -H 'content-type:application/json' \
 -d '{
   "settings": {
      "number_of_shards": 5,
      "number_of_replicas": 1
   },
   "mappings": {
      "dynamic": false,
      "properties": {
         "itemid": { "type": "long" },
         "clock": { "type": "date", "format": "epoch_second" },
         "ns": { "type": "long" },
         "value_parsed": { "type": "flattened" },
         "value_raw": { "type": "keyword", "ignore_above": 1000000 }
    }
  }
}'
```

Luego, cree el ingest pipeline:

```
curl -X PUT \
 http://localhost:9200/_ingest/pipeline/json \
 -H 'content-type:application/json' \
 -d '{
   "processors": [
      {
         "json": {
            "field": "value",
            "target_field": "value_parsed",
            "ignore_failure": true
```

```
         }
      },
      {
         "set": {
            "if": "ctx.value_parsed == null",
            "field": "value_raw",
            "value": "{{{ value }}}"
         }
      }
   ],
   "on_failure": [
      {
         "set": {
            "field": "value_raw",
            "value": "{{{ value }}}"
         }
      }
   ]
}'
```

Elasticsearch responderá con una confirmación de que el ingest pipeline fue creado:

```
{"acknowledged": true}
```

Almacenamiento del historial en índices basados en fecha

En lugar de escribir todos los datos del historial en un único índice (por ejemplo, `uint`), Elasticsearch puede distribuir estos datos entre varios índices basados en fecha (por ejemplo, `uint-2026-01-01`, `uint-2026-01-02`). Esto facilita la gestión del volumen de datos y la retención a lo largo del tiempo.

Para habilitar esto, debe:

- Crear una index template para cada tipo de valor que desee almacenar; esto indica a Elasticsearch qué mapping aplicar cuando crea automáticamente un nuevo índice basado en fecha.
- Crear un ingest pipeline para cada tipo de valor; este procesa cada valor entrante y lo dirige al índice correcto basado en fecha.

Plantillas de índice

Para crear una plantilla para el índice `text`, envíe una solicitud con los siguientes detalles:

- Utilice `_template/text_template` en la URL de su instancia de Elasticsearch.
- Utilice `"text*"` en el campo `"index_patterns"` para que coincida con el nombre del índice.
- Utilice un mapeo para el tipo de valor `text` (consulte los mapeos en el repositorio fuente de Zabbix).

```
curl -X PUT \
 http://localhost:9200/_template/text_template \
 -H 'content-type:application/json' \
 -d '{
   "index_patterns": [ "text*" ],
   "settings": {
      "index": {
         "number_of_replicas": 1,
         "number_of_shards": 5
      }
   },
   "mappings": {
      "properties": {
         "itemid": { "type": "long" },
         "clock": { "format": "epoch_second", "type": "date" },
         "value": {
            "fields": {
               "analyzed": { "index": true, "type": "text", "analyzer": "standard" }
            },
            "index": false,
            "type": "text"
         }
```

```
      }
   }
}'
```

Plantilla para el índice `json`:

```
curl -X PUT \
 http://localhost:9200/_template/json_template \
 -H 'content-type:application/json' \
 -d '{
   "index_patterns": [ "json*" ],
   "settings": {
      "number_of_shards": 5,
      "number_of_replicas": 1
   },
   "mappings": {
      "dynamic": false,
      "properties": {
         "itemid": { "type": "long" },
         "clock": { "type": "date", "format": "epoch_second" },
         "ns": { "type": "long" },
         "value_parsed": { "type": "flattened" },
         "value_raw": { "type": "keyword", "ignore_above": 1000000 }
      }
   }
}'
```

Canalizaciones de ingesta

Para crear una canalización de ingesta para el índice `text`:

- Utilice `_ingest/pipeline/text-pipeline` en la URL de su instancia de Elasticsearch.
- Incluya un procesador `date_index_name` para enrutar cada valor al índice basado en fecha correcto según su marca de tiempo.

```
curl -X PUT \
 http://localhost:9200/_ingest/pipeline/text-pipeline \
 -H 'content-type:application/json' \
 -d '{
   "description": "daily text index naming",
   "processors": [
      {
         "date_index_name": {
            "field": "clock",
            "date_formats": ["UNIX"],
            "index_name_prefix": "text-",
            "date_rounding": "d"
         }
      }
   ]
}'
```

Para el índice `json`, la canalización también necesita analizar el valor JSON antes de enrutarlo al índice correcto:

```
curl -X PUT \
 http://localhost:9200/_ingest/pipeline/json-pipeline \
 -H 'content-type:application/json' \
 -d '{
   "description": "daily json index naming"
   "processors": [
      {
         "json": {
            "field": "value",
            "target_field": "value_parsed",
            "ignore_failure": true
         }
```

```
      },
      {
         "script": {
            "source": "if (ctx.value_parsed == null || !(ctx.value_parsed instanceof Map)) { ctx.value_raw
         }
      },
      {
         "date_index_name": {
            "field": "clock",
            "date_formats": [ "UNIX" ],
            "index_name_prefix": "json-",
            "date_rounding": "d"
         }
      }
   ]
}'
```

Configuración de Zabbix server

En el archivo de configuración de Zabbix server (`zabbix_server.conf`), establezca el parámetro `HistoryProvider`.

Por ejemplo, para almacenar valores de tipo *Character*, *Log*, *Text* y *JSON* en Elasticsearch (manteniendo los valores *Numeric* en una base de datos):

```
HistoryProvider=elasticsearch;value_types="str,log,text,json",url=http://localhost:9200
```

Si utiliza índices basados en fecha, añada `date_index=1` al parámetro:

```
HistoryProvider=elasticsearch;value_types="str,log,text,json",url=http://localhost:9200,date_index=1
```

Después de realizar los cambios, reinicie Zabbix server:

```
systemctl restart zabbix-server
```

Configuración del frontend de Zabbix

En el archivo de configuración del frontend de Zabbix (`zabbix.conf.php`), establezca la variable `$HISTORY_PROVIDERS` para que coincida con la configuración del server:

```
$HISTORY_PROVIDERS[] = [
  'types' => ['str','log','text','json'],
  'provider' => 'elasticsearch',
  'url' => 'http://localhost:9200'
];
```

Solución de problemas

Los siguientes pasos pueden ayudarle a solucionar problemas con su configuración de Elasticsearch:

1. Revise los registros de Elasticsearch o del server de Zabbix para detectar errores.
2. Para identificar consultas lentas, use la opción `log_slow_queries` en el parámetro de configuración del server de Zabbix `HistoryProvider`.
3. Verifique que Elasticsearch permita el acceso desde el server de Zabbix y el frontend de Zabbix.
4. Verifique que `auto_create_index` esté habilitado:

```
curl -X GET \
 "http://localhost:9200/_cluster/settings?include_defaults=true&filter_path=**.auto_create_index"

#### {"defaults": {"action": {"auto_create_index": "false"} } }
```

Para habilitarlo, envíe la siguiente solicitud:

```
curl -X PUT \
 http://localhost:9200/_cluster/settings \
 -H 'content-type:application/json' \
 -d '{
   "persistent": {
      "action.auto_create_index": "true"
```

```
    }
}'

#### {"acknowledged": true, "persistent": {"action": {"auto_create_index": "true"} }, "transient": {} }
```

5. Verifique que los mappings, templates y ingest pipelines sean correctos enviando solicitudes GET a sus respectivas URLs:

```
curl -X GET http://localhost:9200/json
curl -X GET http://localhost:9200/_template/json*
curl -X GET http://localhost:9200/_ingest/pipeline/json*
```

Puede comparar las respuestas recibidas con las respuestas esperadas en la documentación de la API de Elasticsearch.

6. Compruebe si algún shard está en estado fallido; reiniciar Elasticsearch puede resolverlo.

7. Consulte Elasticsearch para ver si los datos recopilados por Zabbix están almacenados, por ejemplo:

```
curl 'http://localhost:9200/json/_search' \
 -H 'Content-Type: application/json' \
 -d '{
   "query": {
     "term": {
       "itemid": 42269
     }
   }
 }'
```

8. Si necesita restablecer su configuración de Elasticsearch y empezar de nuevo, puede eliminar todos los índices, templates e ingest pipelines:

```
curl -X DELETE "http://localhost:9200/_all"
curl -X DELETE "http://localhost:9200/_template/*"
curl -X DELETE "http://localhost:9200/_ingest/pipeline/*"
```

Configuración de ClickHouse

Zabbix puede almacenar los datos históricos en ClickHouse como alternativa a una base de datos relacional.

> **Note:**
> ClickHouse también se puede utilizar como fuente de datos de telemetría para la supervisión del rendimiento de las aplicaciones (APM); para obtener más información, consulte *Administración* > *Fuente de datos*.

Esta guía describe la configuración de las versiones compatibles de ClickHouse. Si utiliza una versión diferente, es posible que algunas funciones no funcionen según lo previsto.

ClickHouse puede almacenar los siguientes tipos de valores:

Tipo de valor del item	Tabla de la base de datos	Tipo de ClickHouse
Numérico (sin signo)	history_uint	uint
Numérico (flotante)	history	dbl
Carácter	history_str	str
Registro	history_log	log
Texto	history_text	text
Binario	history_bin	no compatible con Zabbix
JSON	history_json	json

> **Note:**
> ClickHouse no acepta matrices JSON. Un valor JSON debe ser un único objeto o un conjunto de objetos. Además, ClickHouse gestiona las claves JSON con NULL de la misma manera que las claves ausentes.

Notas importantes

- El housekeeper no elimina datos de ClickHouse. Para controlar durante cuánto tiempo se conservan los datos, configure el período de almacenamiento de datos de ClickHouse en ClickHouse.

- Zabbix **no** calcula ni almacena tendencias en ClickHouse. Considere ampliar el período de almacenamiento del historial para conservar los datos antiguos.
- Si desea migrar los datos del historial desde una base de datos de Zabbix existente (MySQL o PostgreSQL) a ClickHouse, consulte los scripts de migración del esquema y del historial de ClickHouse.
- ClickHouse no es compatible como almacenamiento del historial para los proxy de Zabbix, pero puede utilizarse como fuente de datos de telemetría para un proxy de Zabbix con APM habilitado.
- Si tiene un clúster de ClickHouse, consulte Configuración de clústeres de ClickHouse.

Configuración de ClickHouse

Debe crear y configurar una base de datos y un usuario de Zabbix, e importar el esquema de la base de datos.

Esta guía proporciona instrucciones para instalaciones de ClickHouse mediante Docker o paquetes.

Docker

1. Cree y configure la base de datos y el usuario de Zabbix al ejecutar el contenedor de ClickHouse:

```
sudo docker run -d \
  --name clickhouse \
  -e CLICKHOUSE_DB=zabbix \
  -e CLICKHOUSE_USER=zabbix \
  -e CLICKHOUSE_DEFAULT_ACCESS_MANAGEMENT=1 \
  -e CLICKHOUSE_PASSWORD=<password> \
  -p 8123:8123/tcp \
  -p 9000:9000/tcp \
  --cap-add=SYS_NICE \
  --cap-add=NET_ADMIN \
  --cap-add=IPC_LOCK \
  --ulimit nofile=262144:262144 \
  -v clickhouse_data:/var/lib/clickhouse \
  clickhouse/clickhouse-server:26.4
```

2. Confirme que ClickHouse funciona y que puede conectarse a él:

```
sudo docker exec -it clickhouse \
  clickhouse-client \
  --query "SELECT version()"

#### 26.4.4.38
```

3. Importe el esquema de la base de datos usando el script `history_all.sh` desde su directorio de Zabbix:

```
./usr/share/zabbix/sql-scripts/clickhouse/history_all.sh \
  --user zabbix \
  --password <password> \
  --db zabbix \
  --server http://localhost:8123
```

> **Note:**
> El período de almacenamiento de datos (Time-To-Live, o TTL) para ClickHouse es, de forma predeterminada, de 31 días. Para cambiarlo, use la opción `--ttl` al importar el esquema de la base de datos (por ejemplo, `--ttl 604800` para 7 días), o configúrelo más adelante.

4. Verifique que se hayan creado las tablas:

```
sudo docker exec -it clickhouse \
  clickhouse-client \
  --user zabbix \
  --password <password> \
  --query "SHOW TABLES FROM zabbix"

#### history
#### history_json
#### history_log
#### history_str
#### history_text
#### history_uint
```

Paquetes

1. Inicie ClickHouse:

```
sudo service clickhouse-server start
```

2. Cree y configure la base de datos y el usuario de Zabbix:

```
clickhouse-client --user default

localhost :) CREATE DATABASE IF NOT EXISTS zabbix
localhost :) CREATE USER IF NOT EXISTS zabbix IDENTIFIED WITH sha256_password BY '<password>'
localhost :) GRANT CREATE, ALTER, DROP, INSERT, SELECT, UPDATE, OPTIMIZE ON zabbix.* TO zabbix
localhost :) quit
```

3. Importe el esquema de la base de datos usando el script `history_all.sh` desde su directorio de Zabbix:

```
./usr/share/zabbix/sql-scripts/clickhouse/history_all.sh \
  --user zabbix \
  --password <password> \
  --db zabbix \
  --server http://localhost:8123
```

> **Note:**
> El período de almacenamiento de datos (Time-To-Live, o TTL) para ClickHouse es, de forma predeterminada, de 31 días. Para cambiarlo, use la opción `--ttl` al importar el esquema de la base de datos (por ejemplo, `--ttl 604800` para 7 días), o configúrelo más adelante.

4. Verifique que se hayan creado las tablas:

```
clickhouse-client \
  --user zabbix \
  --password <password> \
  --query "SHOW TABLES FROM zabbix"

#### history
#### history_json
#### history_log
#### history_str
#### history_text
#### history_uint
```

Configuración de Zabbix server

En el archivo de configuración de Zabbix server (`zabbix_server.conf`), establezca el parámetro `HistoryProvider`.

Por ejemplo, para almacenar todos los tipos de valores admitidos en ClickHouse:

```
HistoryProvider=clickhouse;value_types="uint,dbl,str,log,text,json",url=http://localhost:8123,db=zabbix,us
```

Después de realizar los cambios, reinicie Zabbix server:

```
systemctl restart zabbix-server
```

Configuración del frontend de Zabbix

En el archivo de configuración del frontend de Zabbix (`zabbix.conf.php`), establezca la variable `$HISTORY_PROVIDERS` para que coincida con la configuración del server:

```
$HISTORY_PROVIDERS[] = [
  'types' => ['uint','dbl','str','log','text','json'],
  'provider' => 'clickhouse',
  'url' => 'http://localhost:8123',
  'db' => 'zabbix',
  'username' => 'zabbix',
  'password' => '<password>'
];
```

Configuración adicional

Los pasos siguientes son opcionales. No los necesita para una configuración básica.

Configuración del período de almacenamiento de datos de ClickHouse

El período de almacenamiento de datos (Time-To-Live, o TTL) de ClickHouse es, de forma predeterminada, de 31 días. Para cambiarlo, ejecute los comandos siguientes.

Los ejemplos siguientes usan Docker. Si instaló ClickHouse mediante paquetes, ejecute las consultas directamente en el cliente de ClickHouse.

1. Modifique la tabla (reemplace `history_json` y 3600 por los valores que necesite):

```
sudo docker exec -it clickhouse \
  clickhouse-client \
  --user zabbix \
  --password <password> \
  --query "ALTER TABLE zabbix.history_json MODIFY TTL clock_ns + toIntervalSecond(3600)"
```

2. Aplique los cambios inmediatamente:

```
sudo docker exec -it clickhouse \
  clickhouse-client \
  --user zabbix \
  --password <password> \
  --query "OPTIMIZE TABLE zabbix.history_json MATERIALIZE TTL"
```

3. Verifique que el período de almacenamiento de datos haya cambiado:

```
sudo docker exec -it clickhouse \
  clickhouse-client \
  --user zabbix \
  --password <password> \
  --query "SHOW CREATE TABLE zabbix.history_json"

#### CREATE TABLE zabbix.history_json
#### (
####     `itemid` UInt64,
####     `clock_ns` DateTime64(9),
####     `value` JSON,
####     `value_str` String
#### )
#### ENGINE = MergeTree
#### PARTITION BY toDate(clock_ns)
#### PRIMARY KEY (itemid, clock_ns)
#### ORDER BY (itemid, clock_ns)
#### TTL clock_ns + toIntervalSecond(3600)
#### SETTINGS index_granularity = 8192
```

4. Reinicie Zabbix server para actualizar el período de almacenamiento de datos en *Administración* > *Housekeeping*:

```
systemctl restart zabbix-server
```

Configuración de clústeres de ClickHouse

Esta sección proporciona los pasos de configuración para clústeres de ClickHouse con replicación y particionado.

Los pasos siguientes usan Docker y el ClickHouse `cluster_2S_2R` de la documentación de ClickHouse (ejecutando ClickHouse 26.4.4.38 con ClickHouse Keeper).

Antes de empezar, se recomienda colocar un balanceador de carga delante de su clúster de ClickHouse para Zabbix. La configuración de Zabbix server y frontend solo admite una única URL de ClickHouse, por lo que solo puede apuntar a un nodo. Si ese nodo deja de funcionar, un balanceador de carga redirigirá Zabbix a un nodo sano en su lugar, a través de esa misma URL.

1. Cree la base de datos de Zabbix:

```
sudo docker exec -it clickhouse-01 \
  clickhouse-client \
  --user default \
  --query "CREATE DATABASE IF NOT EXISTS zabbix ON CLUSTER cluster_2S_2R ENGINE = Replicated('/clickhouse/
```

Attention:
La cláusula `ON CLUSTER` alcanza solo a los nodos que forman parte del clúster cuando ejecuta el comando. Si añade nodos a su clúster más adelante, ejecute este comando de nuevo.

2. Cree y configure el usuario de Zabbix:

```
sudo docker exec -it clickhouse-01 \
  clickhouse-client \
  --user default \
  --query "CREATE USER IF NOT EXISTS zabbix ON CLUSTER cluster_2S_2R IDENTIFIED WITH sha256_password BY '<

sudo docker exec -it clickhouse-01 \
  clickhouse-client \
  --user default \
  --query "GRANT ON CLUSTER cluster_2S_2R CREATE, ALTER, DROP, INSERT, SELECT, UPDATE, OPTIMIZE ON zabbix.
```

3. Importe el esquema de la base de datos usando el script `history_all.sh` de su directorio de Zabbix:

```
./usr/share/zabbix/sql-scripts/clickhouse/history_all.sh \
  --user zabbix \
  --password <password> \
  --db zabbix \
  --server http://localhost:8123 \
  --engine "ReplicatedMergeTree()"
```

Note:
El período de almacenamiento de datos (Time-To-Live, o TTL) para ClickHouse es, de forma predeterminada, de 31 días. Para cambiarlo, use la opción `--ttl` al importar el esquema de la base de datos (por ejemplo, `--ttl 604800` para 7 días), o configúrelo más adelante.

Note:
Si su clúster de ClickHouse requiere rutas explícitas para las réplicas (por ejemplo, si la configuración de su servidor ClickHouse no establece rutas predeterminadas para tablas replicadas), ejecute los scripts de esquema (`history_schema.sh`, `history_uint_schema.sh`, etc.) de forma individual y establezca una ruta diferente para cada tabla:

```
./usr/share/zabbix/sql-scripts/clickhouse/history_schema.sh \
  --user zabbix \
  --password <password> \
  --db zabbix \
  --server http://localhost:8123 \
  --engine "ReplicatedMergeTree('/clickhouse/tables/{shard}/history', '{replica}')"
```

Para obtener más detalles sobre la replicación en ClickHouse, consulte Replicating data y Replicated* table engines en la documentación de ClickHouse.

4. Configure el sharding para cada tipo de tabla. Cambie el nombre de las tablas y cree una tabla `Distributed` con el nombre original, apuntando a la tabla renombrada. Esta tabla `Distributed` es la que Zabbix realmente lee y escribe; la tabla divide los datos entre shards automáticamente. Por ejemplo, para la tabla `history`:

```
sudo docker exec -it clickhouse-01 \
  clickhouse-client \
  --user default \
  --query "RENAME TABLE zabbix.history TO zabbix.history_local"

sudo docker exec -it clickhouse-01 \
  clickhouse-client \
  --user default \
  --query "CREATE TABLE zabbix.history AS zabbix.history_local ENGINE = Distributed('cluster_2S_2R', 'zabb
```

Note:
Para obtener más detalles sobre el sharding en ClickHouse, consulte Distributed table engine en la documentación de ClickHouse.

5. Configure Zabbix server.

6. Configure Zabbix frontend.

7. Verifique que los datos de Zabbix estén distribuidos entre los shards. El recuento de filas de `zabbix.history_local` puede cambiar entre comprobaciones repetidas, ya que Zabbix server inserta continuamente nuevos datos. Ejecute este comando en cada nodo:

```
sudo docker exec -it clickhouse-01 \
  clickhouse-client \
  --user default \
  --query "SELECT count(*) FROM zabbix.history_local"

#### 2446 (clickhouse-01)
#### 2471 (clickhouse-02)
#### 2494 (clickhouse-03)
```

8. Compruebe que la replicación esté sincronizada (`active_replicas` debe coincidir con `total_replicas`):

```
sudo docker exec -it clickhouse-01 \
  clickhouse-client \
  --user default \
  --query "SELECT database, table, is_leader, total_replicas, active_replicas FROM system.replicas WHERE d

#### zabbix  history_json_local  1 2 2
#### zabbix  history_local       1 2 2
#### zabbix  history_log_local   1 2 2
#### zabbix  history_str_local   1 2 2
#### zabbix  history_text_local  1 2 2
#### zabbix  history_uint_local  1 2 2
```

Solución de problemas

Los siguientes pasos pueden ayudarle a solucionar problemas con su configuración de ClickHouse:

1. Revise los registros de ClickHouse o del server de Zabbix en busca de errores.
2. Para identificar consultas lentas, use la opción `log_slow_queries` en el parámetro de configuración del server de Zabbix `HistoryProvider`.
3. Verifique que ClickHouse permita el acceso desde el server de Zabbix y el frontend de Zabbix.
4. Consulte ClickHouse para ver si los datos recopilados por Zabbix se almacenan, por ejemplo:

```
sudo docker exec -it clickhouse \
  clickhouse-client \
  --user zabbix \
  --password <password> \
  --query "SELECT * FROM zabbix.history_uint WHERE itemid = 42269"
```

Notas específicas de la distribución para configurar NGINX para Zabbix

RHEL

En RHEL 8, 9 y 10, NGINX está incluido en AppStream; no se necesitan repositorios adicionales. Simplemente habilítelo e instálelo con:

```
dnf module list nginx
dnf -y install nginx
```

En RHEL 7, NGINX se puede instalar desde el repositorio CentOS 7 Extras (si utiliza CentOS) mediante:

```
yum -y install nginx
```

Si prefiere las compilaciones upstream más recientes, configure el repositorio oficial de NGINX en `/etc/yum.repos.d/nginx.repo`:

```
cat > /etc/yum.repos.d/nginx.repo << 'EOF'
[nginx-stable]
name=nginx stable repo
baseurl=https://nginx.org/packages/rhel/$releasever/$basearch/
gpgcheck=1
enabled=1
```

```
gpgkey=https://nginx.org/keys/nginx_signing.key
EOF
dnf -y install nginx
```

SLES 15

En SUSE Linux Enterprise Server 15 necesita configurar `php-fpm` (la ruta al archivo de configuración puede variar ligeramente dependiendo del service pack):

```
cp /etc/php8/fpm/php-fpm.conf{.default,}
cp /etc/php8/fpm/php-fpm.d/www.conf{.default,}
sed -i 's/user = nobody/user = wwwrun/; s/group = nobody/group = www/' /etc/php8/fpm/php-fpm.d/www.conf
```

Configuración de SAML con Microsoft Entra ID

Descripción general

Esta sección proporciona pautas para configurar el inicio de sesión único y el aprovisionamiento de usuarios en Zabbix desde Microsoft Entra ID (anteriormente Microsoft Azure Active Directory) utilizando la autenticación SAML 2.0.

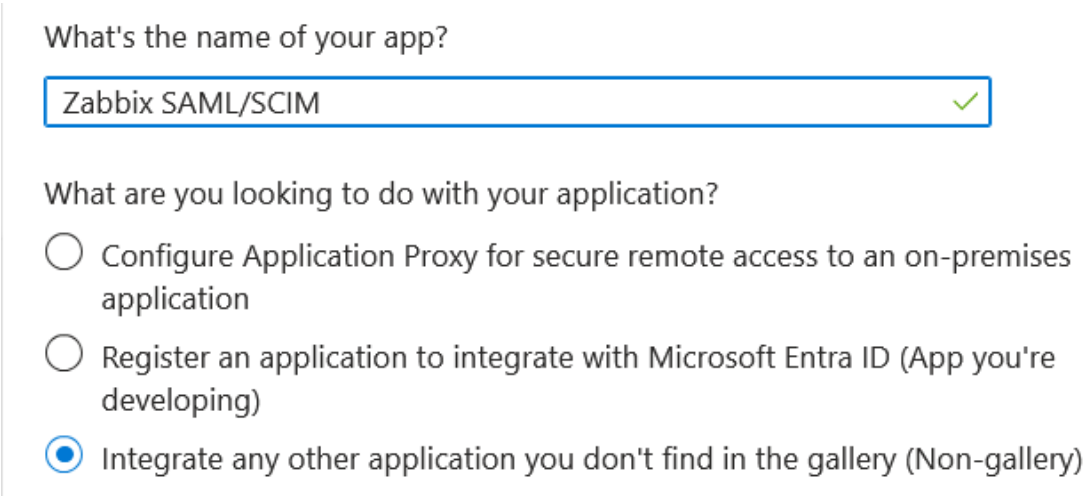
Configuración de Microsoft Entra ID

Creación de una aplicación

1. Inicie sesión en Microsoft Entra admin center en Microsoft Entra ID. Para fines de prueba, puede crear una cuenta de prueba gratuita en Microsoft Entra ID.

2. En Microsoft Entra admin center, seleccione *Applications* > *Enterprise applications* > *New application* > *Create your own application*.

3. Agregue el nombre de su aplicación y seleccione la opción *Integrate any other application...*. Después de eso, haga clic en *Create*.

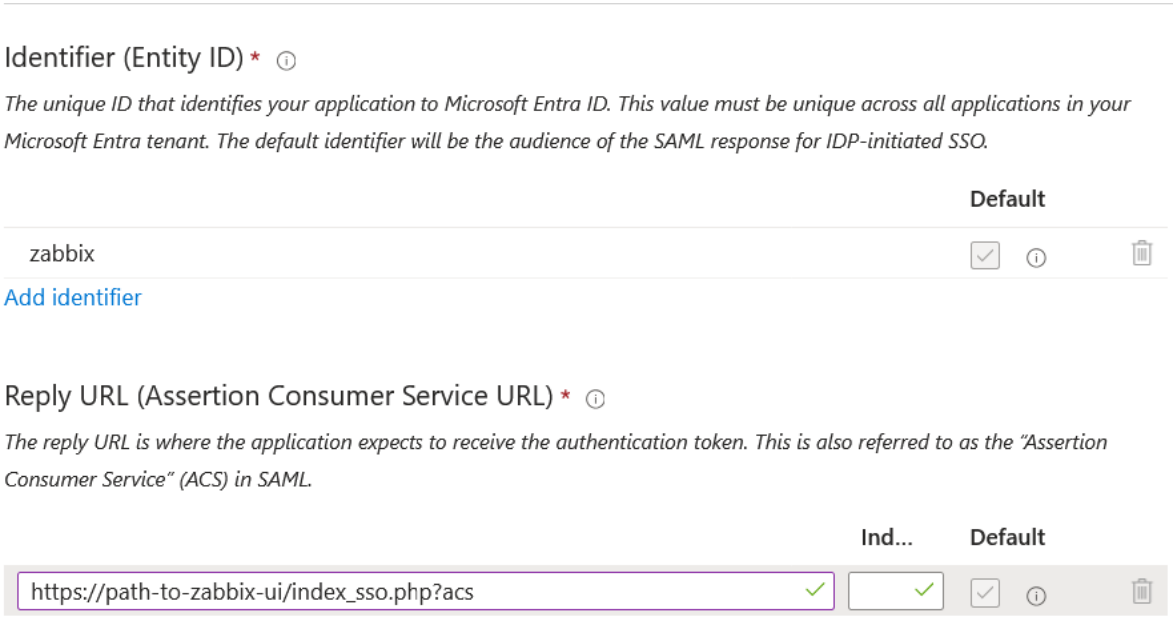


Configuración del inicio de sesión único

1. En la página de su aplicación, vaya a *Set up single sign on* y haga clic en *Get started*. A continuación, seleccione *SAML*.

2. Edite *Basic SAML Configuration*:

- En *Identifier (Entity ID)*, establezca un nombre único para identificar su aplicación en Microsoft Entra ID, por ejemplo, `zabbix`;
- En *Reply URL (Assertion Consumer Service URL)*, establezca el endpoint de inicio de sesión único de Zabbix: `https://<path-to-zabbi`



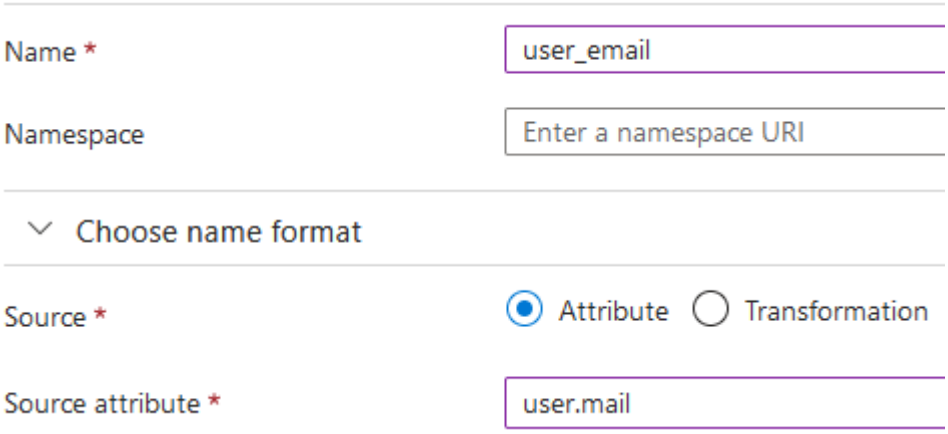
Tenga en cuenta que se requiere `https`. Para usar esto con Zabbix, añada la siguiente línea al archivo de configuración del frontend:

```
$SSO['SETTINGS'] = ['use_proxy_headers' => true];
```

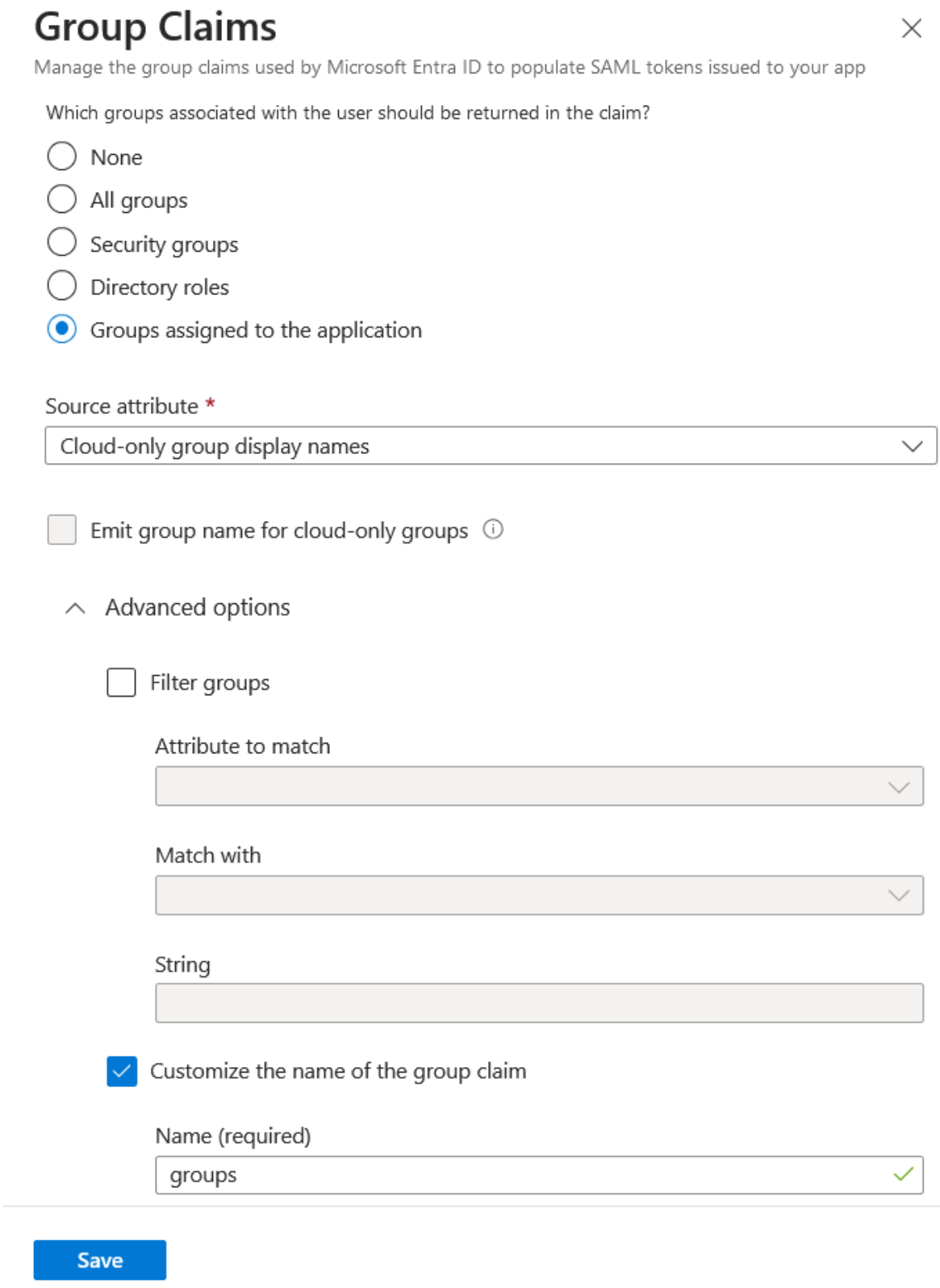
3. Edite *Attributes & Claims*. Debe añadir todos los atributos que desee pasar a Zabbix (user_name, user_lastname, user_email, user_mobile, groups).

Los nombres de los atributos son arbitrarios. Se pueden utilizar nombres de atributos diferentes; sin embargo, es necesario que coincidan con el valor del campo correspondiente en la configuración SAML de Zabbix.

- Haga clic en *Add new claim* para añadir un atributo:



- Haga clic en *Add a group claim* para añadir un atributo para pasar los grupos a Zabbix:



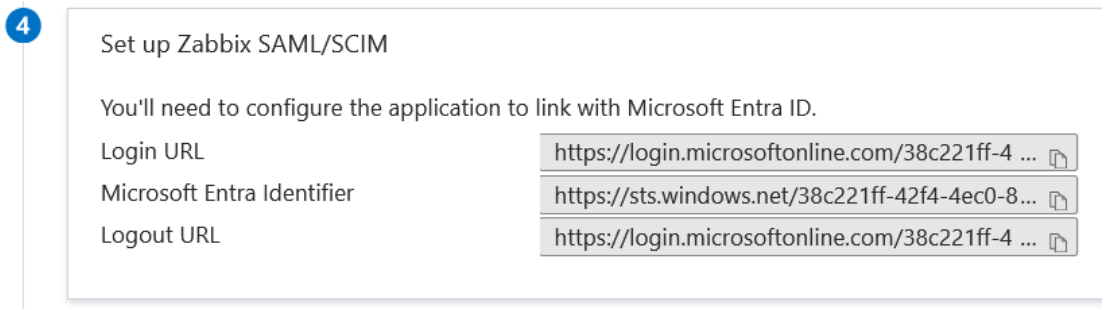
Es importante que, en esta declaración, los nombres de los grupos (en lugar de sus ID) se pasen a Zabbix mediante el *Source attribute* seleccionado. De lo contrario, el aprovisionamiento de usuarios JIT no funcionará correctamente.

4. En *SAML Certificates*, descargue el certificado Base64 proporcionado por Entra ID y colóquelo en `conf/certs` de la instalación del frontend de Zabbix (la ubicación puede variar según el sistema operativo o el origen de su instalación de Zabbix).

Establezca los permisos 644 ejecutando:

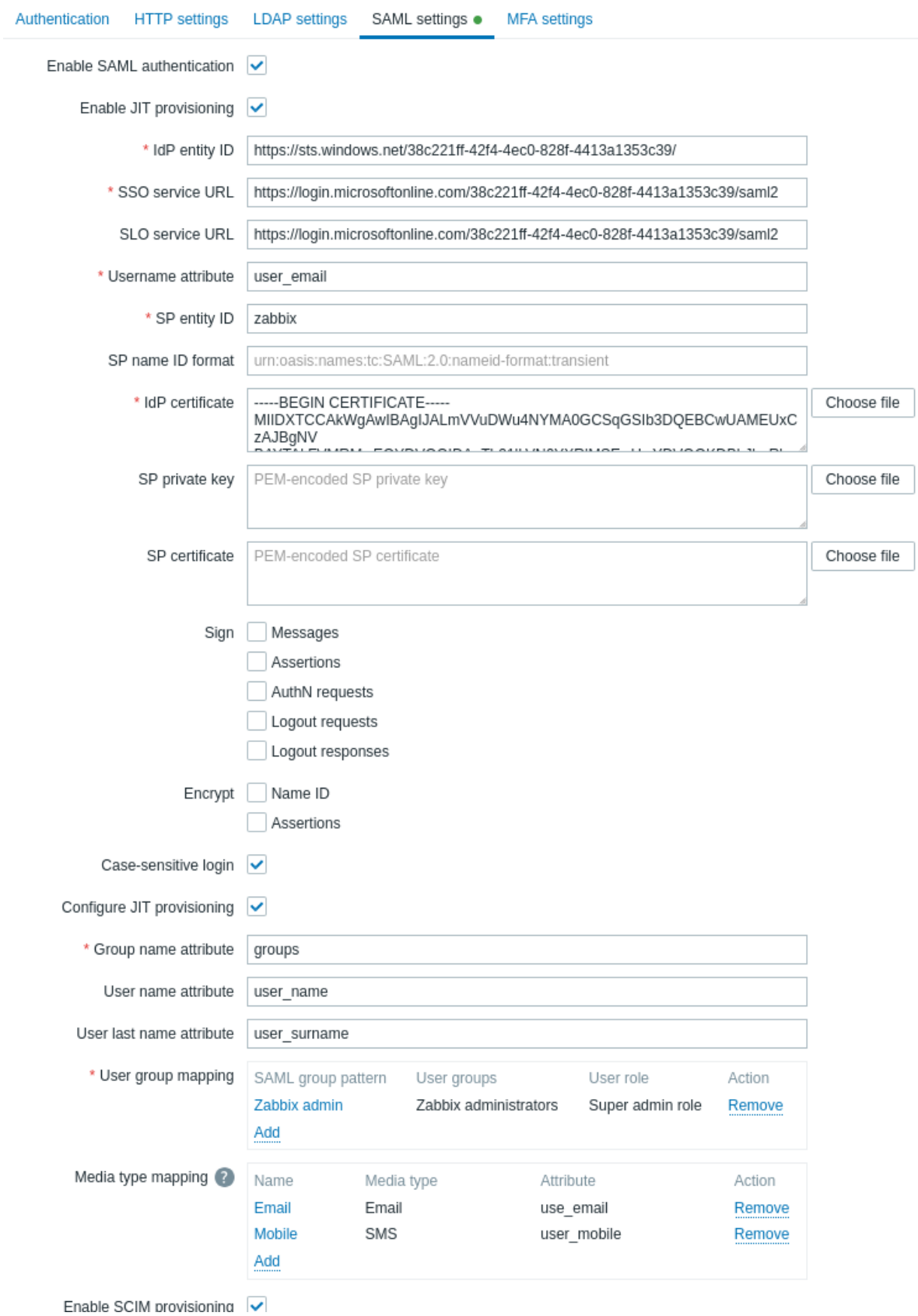
```
chmod 644 entra.cer
```

5. Utilice los valores de *Set up <your app name>* en Entra ID para configurar la autenticación SAML de Zabbix (consulte la siguiente sección):



Configuración de Zabbix

1. En Zabbix, vaya a Configuración de SAML y complete las opciones de configuración según la configuración de Entra ID:



Campo de Zabbix	Campo de configuración en Entra ID	Valor de ejemplo
IdP entity ID	Identificador de Microsoft Entra	
SSO service URL	URL de inicio de sesión	
SLO service URL	URL de cierre de sesión	
SP entity ID	Identificador (Entity ID)	
Username attribute	Atributo personalizado (claim)	`user_email`
Group name attribute	Atributo personalizado (claim)	`groups`
User name attribute	Atributo personalizado (claim)	`user_name`

Campo de Zabbix	Campo de configuración en Entra ID	Valor de ejemplo
User last name attribute	Atributo personalizado (claim)	`user_lastname`

2. Añada el certificado Base64 proporcionado por Entra ID.

Si `$SSO['CERT_STORAGE'] = 'database'` está establecido en el archivo de configuración del frontend (`zabbix.conf.php`), puede pegar el texto del certificado o cargar el archivo del certificado en el frontend durante la configuración de SAML; no se requieren archivos en el sistema de archivos.

Si `$SSO['CERT_STORAGE'] = 'file'` está establecido en el archivo de configuración, el certificado debe estar disponible en el sistema de archivos (de forma predeterminada, en `ui/conf/certs` o en la ruta configurada en el archivo de configuración), y el frontend no almacenará los certificados en la base de datos. Tenga en cuenta que, si `$SSO['CERT_STORAGE']` no está establecido o está comentado, se asume el almacenamiento en archivos y los certificados se leen desde `ui/conf/certs`.

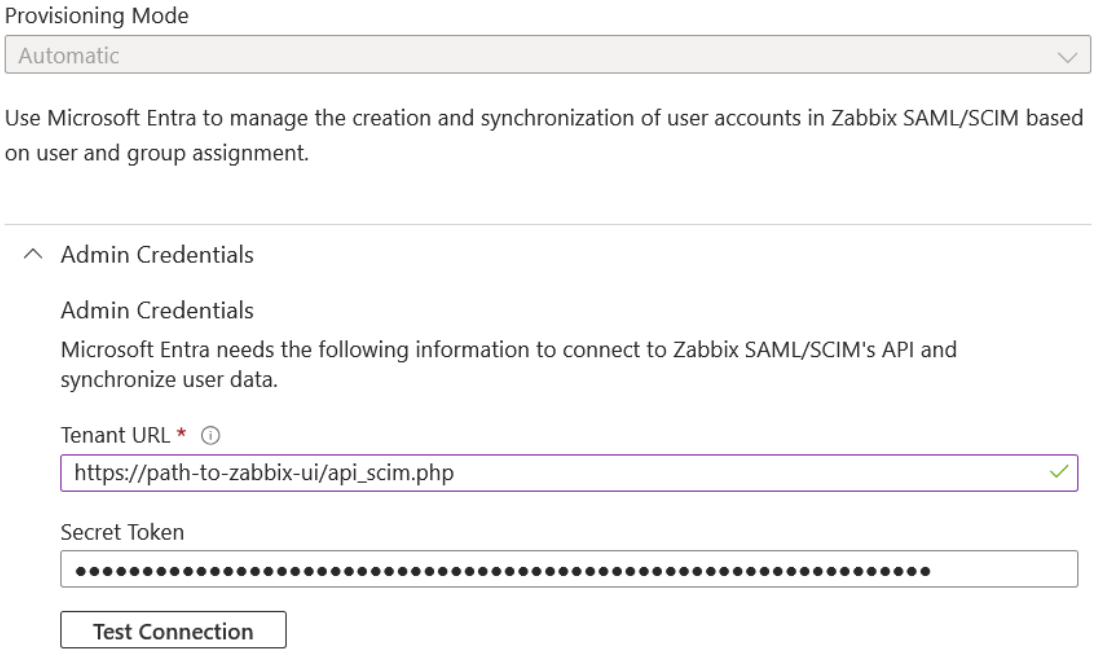
También es necesario configurar la asignación de grupos de usuarios. La asignación de medios es opcional.

3. Haga clic en el botón *Actualizar* para guardar esta configuración.

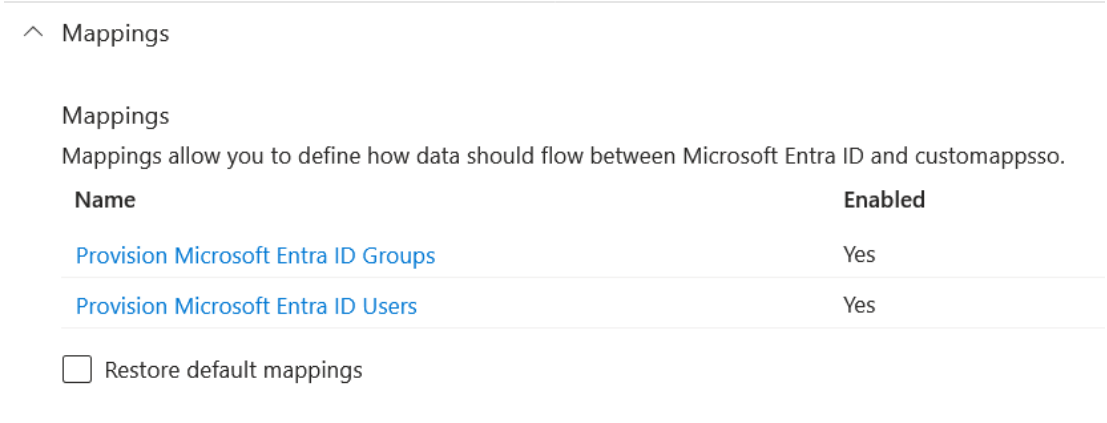
Aprovisionamiento de usuarios SCIM

1. En la página de tu aplicación de Entra ID, desde el menú principal abre la página *Provisioning*. Haz clic en *Get started* y luego selecciona el modo de aprovisionamiento automático:

- En *Tenant URL*, establece el siguiente valor: `https://<path-to-zabbix-ui>/api_scim.php`
- En *Secret token*, introduce un token de API de Zabbix con permisos de Super admin.
- Haz clic en *Test connection* para comprobar si la conexión se establece.

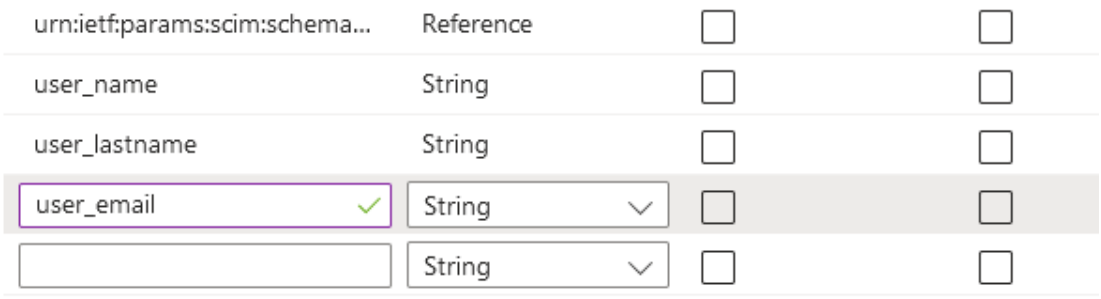


2. Ahora puedes añadir todos los atributos que se pasarán con SCIM a Zabbix. Para ello, haz clic en *Mappings* y luego en *Provision Microsoft Entra ID Users*.



En la parte inferior de la lista *Attribute Mapping*, habilita *Show advanced options* y luego haz clic en *Edit attribute list for custom-appsso*.

En la parte inferior de la lista de atributos, añade tus propios atributos con tipo 'String':



Guarda la lista.

3. Ahora puedes añadir asignaciones para los atributos añadidos. En la parte inferior de la lista *Attribute Mapping*, haz clic en *Add New Mapping* y crea las asignaciones como se muestra a continuación:



Cuando se hayan añadido todas las asignaciones, guarda la lista de asignaciones.



4. Como requisito previo para el aprovisionamiento de usuarios en Zabbix, debes tener usuarios y grupos configurados en Entra ID.

Para ello, ve a *Microsoft Entra admin center* y luego añade usuarios/grupos en las páginas correspondientes de Users y Groups.

5. Cuando se hayan creado usuarios y grupos en Entra ID, puedes ir al menú *Users and groups* de tu aplicación y añadirlos a la app.

6. Ve al menú *Provisioning* de tu app y haz clic en *Start provisioning* para aprovisionar usuarios en Zabbix.

Ten en cuenta que la solicitud PATCH de Users en Entra ID no admite cambios en media.

Firma de solicitudes de autenticación

Es posible configurar Entra ID para validar la firma de las solicitudes de autenticación firmadas.

Para que esto funcione, cree claves públicas/privadas:

```
openssl req -x509 -newkey rsa:4096 -keyout /usr/share/zabbix/conf/certs/request-sign.key -out /usr/share/z
```

Asigne los permisos:

```
chown apache /usr/share/zabbix/conf/certs/request-sign.key
chmod 400 /usr/share/zabbix/conf/certs/request-sign.key
```

Actualice la configuración del frontend de Zabbix añadiendo:

```
$SSO['SP_KEY'] = 'conf/certs/request-sign.key';
$SSO['SP_CERT'] = 'conf/certs/request-sign.crt';
```

Solución de problemas

Pueden producirse problemas de autenticación con los navegadores Microsoft Edge cuando un usuario que intenta iniciar sesión en Zabbix mediante SAML ya ha iniciado sesión con el perfil de Microsoft Edge. Como indicio de este problema, el usuario puede iniciar sesión en Zabbix usando Microsoft Edge en modo privado.

Para evitar problemas de autenticación en este caso, puede ser necesario establecer `requestedAuthnContext` en `false` en el archivo de configuración del frontend de Zabbix.

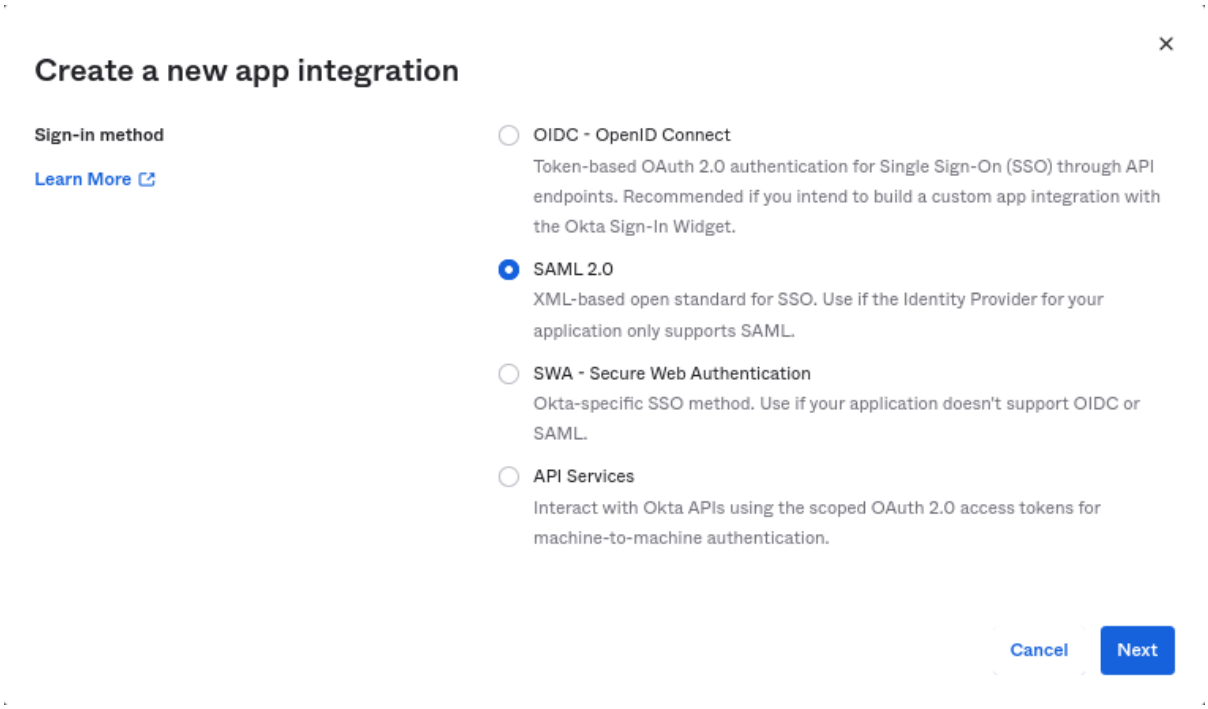
```
$SSO['SETTINGS'] = [
    'security' => [
        'requestedAuthnContext' => false
    ]
];
```

Configuración de SAML con Okta

Esta sección proporciona directrices para configurar Okta para habilitar la autenticación SAML 2.0 y el aprovisionamiento de usuarios para Zabbix.

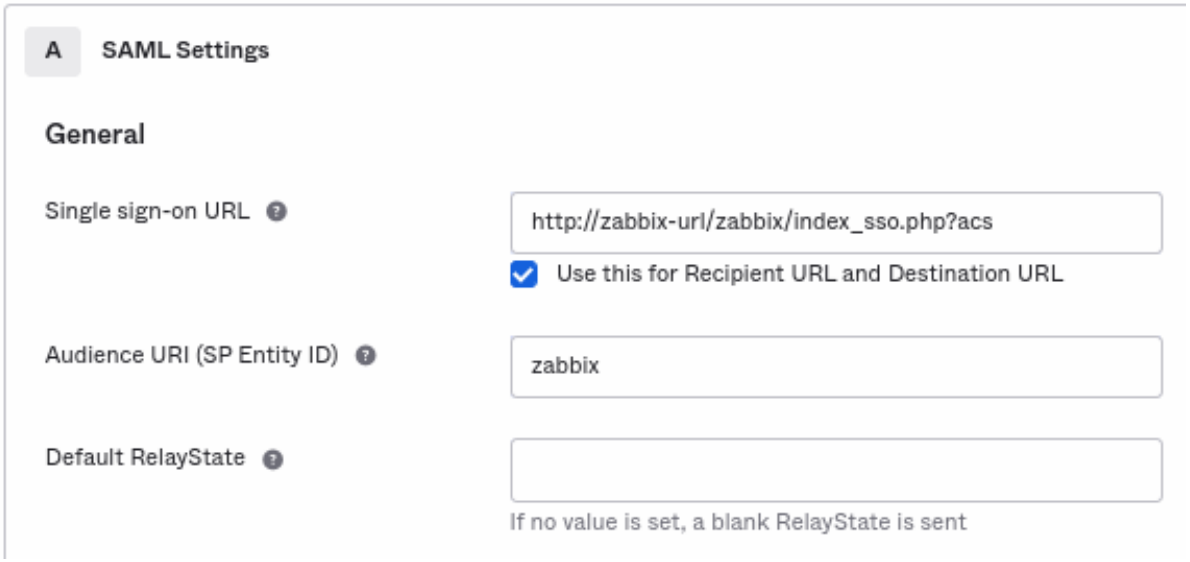
Configuración de Okta

1. Vaya a https://developer.okta.com/signup/ y regístrese/inicie sesión en su cuenta.
2. En la interfaz web de Okta, navegue a *Applications* > *Applications*.
3. Haga clic en *Create App Integration*.

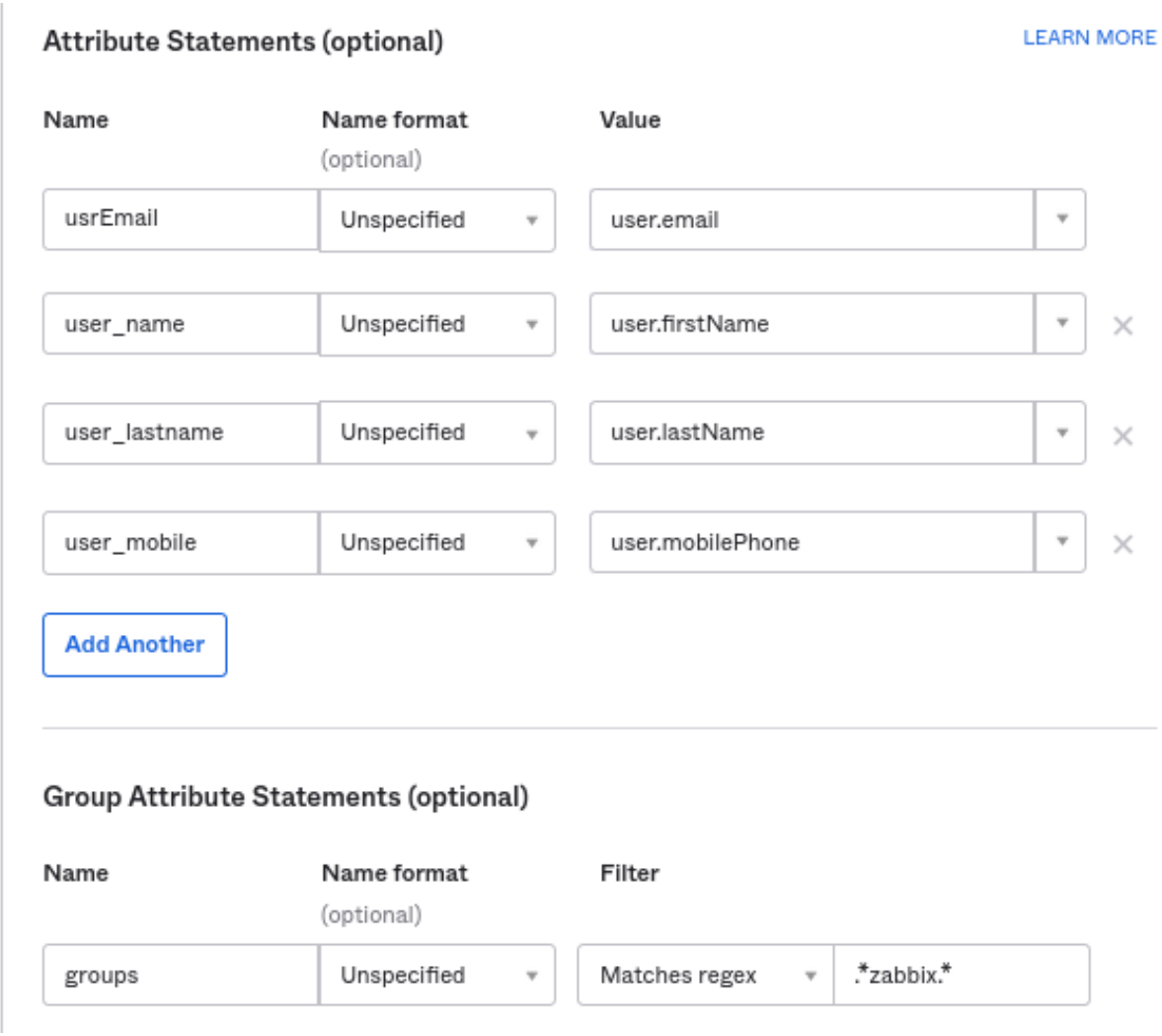


Seleccione "SAML 2.0" como método de inicio de sesión y haga clic en *Next*.

4. En la configuración general, complete el nombre de la aplicación y haga clic en *Next*.
5. En la configuración SAML, introduzca los valores que se indican a continuación y, a continuación, haga clic en *Next*.



- En **General** añada:
  - *Single sign-on URL*: `http://<your-zabbix-url>/zabbix/index_sso.php?acs<br>` Tenga en cuenta el uso de "http" y no de "https", para que el parámetro `acs` no se elimine de la solicitud. La casilla *Use this for Recipient URL and Destination URL* también debe estar marcada.
  - *Audience URI (SP Entity ID)*: `zabbix<br>` Tenga en cuenta que este valor se utilizará dentro de la aserción SAML como identificador único del proveedor de servicios (si no coincide, la operación será rechazada). Es posible especificar una URL o cualquier cadena de datos en este campo.
  - *Default RelayState*:
 Deje este campo en blanco; si se requiere una redirección personalizada, puede añadirse en Zabbix en la configuración *Users* > *Users*.
  - Complete los demás campos según sus preferencias.
- En **Attribute Statements/Group Attribute Statements** añada:



Estas declaraciones de atributos se insertan en las aserciones SAML compartidas con Zabbix.

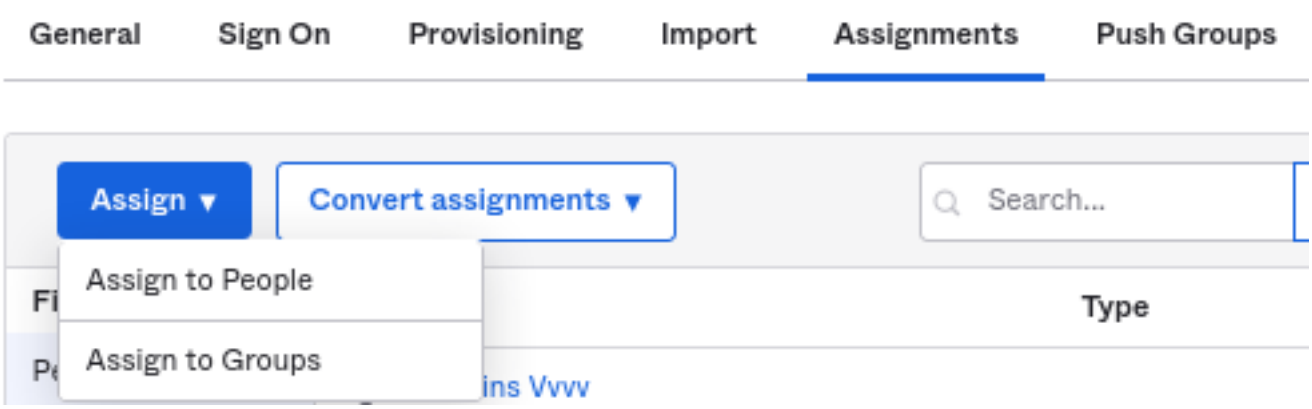
Los nombres de atributos utilizados aquí son ejemplos arbitrarios. Puede usar nombres de atributos diferentes; sin embargo, es necesario que coincidan con el valor del campo correspondiente en la configuración SAML de Zabbix.

Si desea configurar el inicio de sesión SAML en Zabbix *sin* aprovisionamiento de usuarios JIT, entonces solo se requiere el atributo de correo electrónico.

Note:
Si planea usar una conexión cifrada, genere los certificados de cifrado privado y público y, a continuación, cargue el certificado público en Okta. El formulario de carga del certificado aparece cuando *Assertion Encryption* está configurado como "Encrypted" (haga clic en *Show Advanced Settings* para encontrar este parámetro).

6. En la siguiente pestaña, seleccione "I'm a software vendor. I'd like to integrate my app with Okta" y haga clic en *Finish*.

7. Navegue a la pestaña "Assignments" de la aplicación recién creada y haga clic en el botón *Assign*, luego seleccione "Assign to People" en el menú desplegable.



8. En la ventana emergente que aparece, asigne la aplicación a las personas que usarán SAML 2.0 para autenticarse en Zabbix y, a continuación, haga clic en *Save and go back*.

9. Descargue el certificado IdP. Asigne permisos 644 ejecutando:

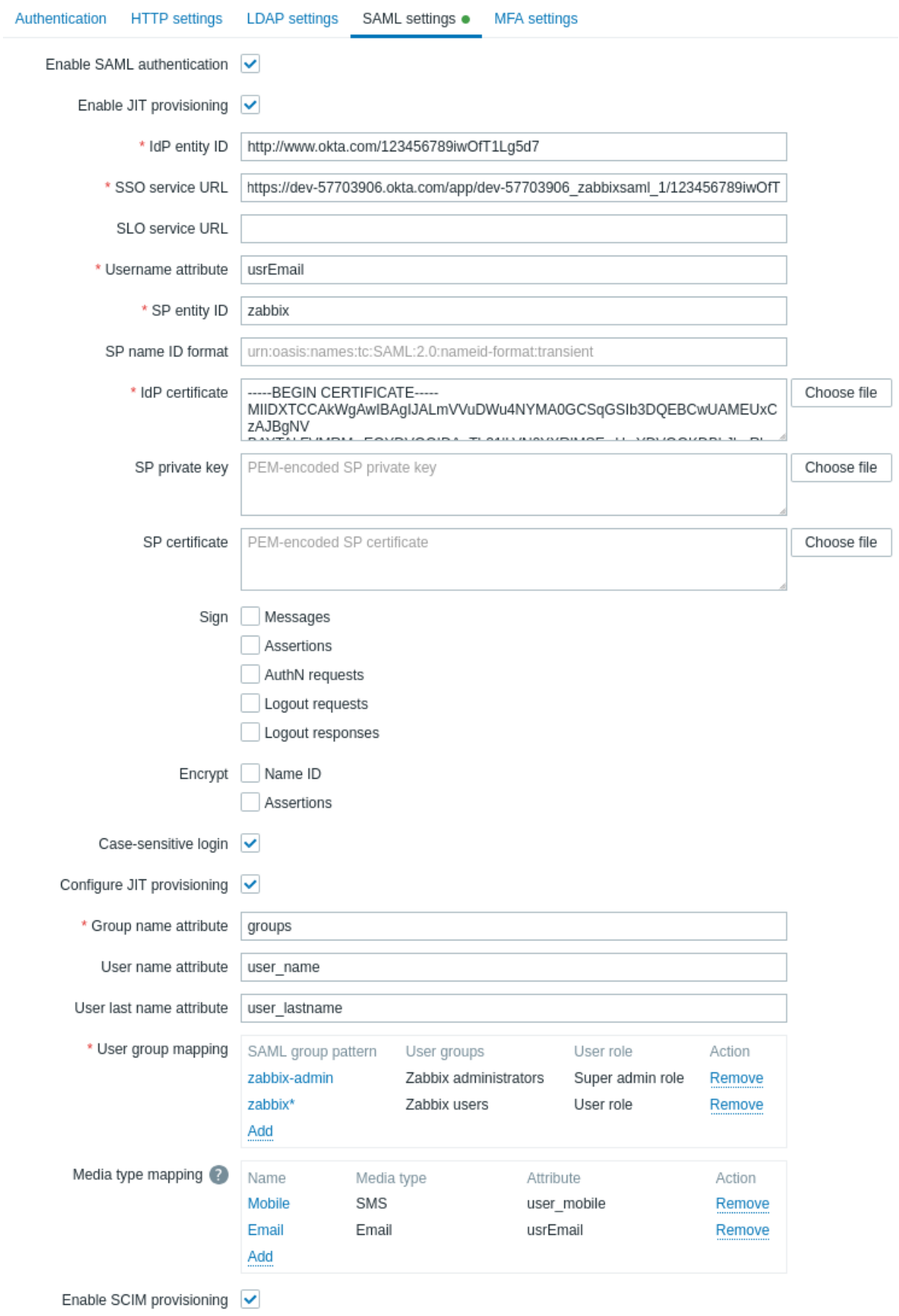
```
chmod 644 idp.crt
```

10. Navegue a la pestaña "Sign On" y haga clic en el botón *View Setup Instructions*.

Las **instrucciones** de configuración se abrirán en una nueva pestaña; mantenga esta pestaña abierta mientras configura Zabbix.

Configuración de Zabbix

1. En Zabbix, vaya a Configuración de SAML y complete las opciones de configuración según las instrucciones de configuración de Okta:



Campo de Zabbix	Campo de configuración en Okta	Valor de ejemplo
IdP entity ID	Identity Provider Issuer	
SSO service URL	Identity Provider Single Sign-On URL	
Username attribute	Attribute name	`usrEmail`
SP entity ID	Audience URI	`zabbix`
Group name attribute	Attribute name	`groups`

Campo de Zabbix	Campo de configuración en Okta	Valor de ejemplo
User name attribute	Attribute name	`user_name`
User last name attribute	Attribute name	`user_lastname`

También es necesario configurar la asignación de grupos de usuarios y medios.

2. Añada el certificado Base64 proporcionado en las instrucciones de configuración de SAML de Okta.

Si `$SSO['CERT_STORAGE'] = 'database'` está establecido en el archivo de configuración del frontend (`zabbix.conf.php`), puede pegar el texto del certificado o cargar el archivo de certificado en el frontend durante la configuración de SAML; no se requieren archivos en el sistema de archivos.

Si `$SSO['CERT_STORAGE'] = 'file'` está establecido en el archivo de configuración, el certificado debe estar disponible en el sistema de archivos (de forma predeterminada, en `ui/conf/certs` o en la ruta configurada en el archivo de configuración), y el frontend no almacenará los certificados en la base de datos.
Tenga en cuenta que, si `$SSO['CERT_STORAGE']` no está establecido o está comentado, se asumirá el almacenamiento en archivos y los certificados se leerán desde `ui/conf/certs`.

3. Si *Assertion Encryption* está establecido en "Encrypted" en Okta, la casilla "Assertions" del parámetro *Encrypt* también debe marcarse en Zabbix.

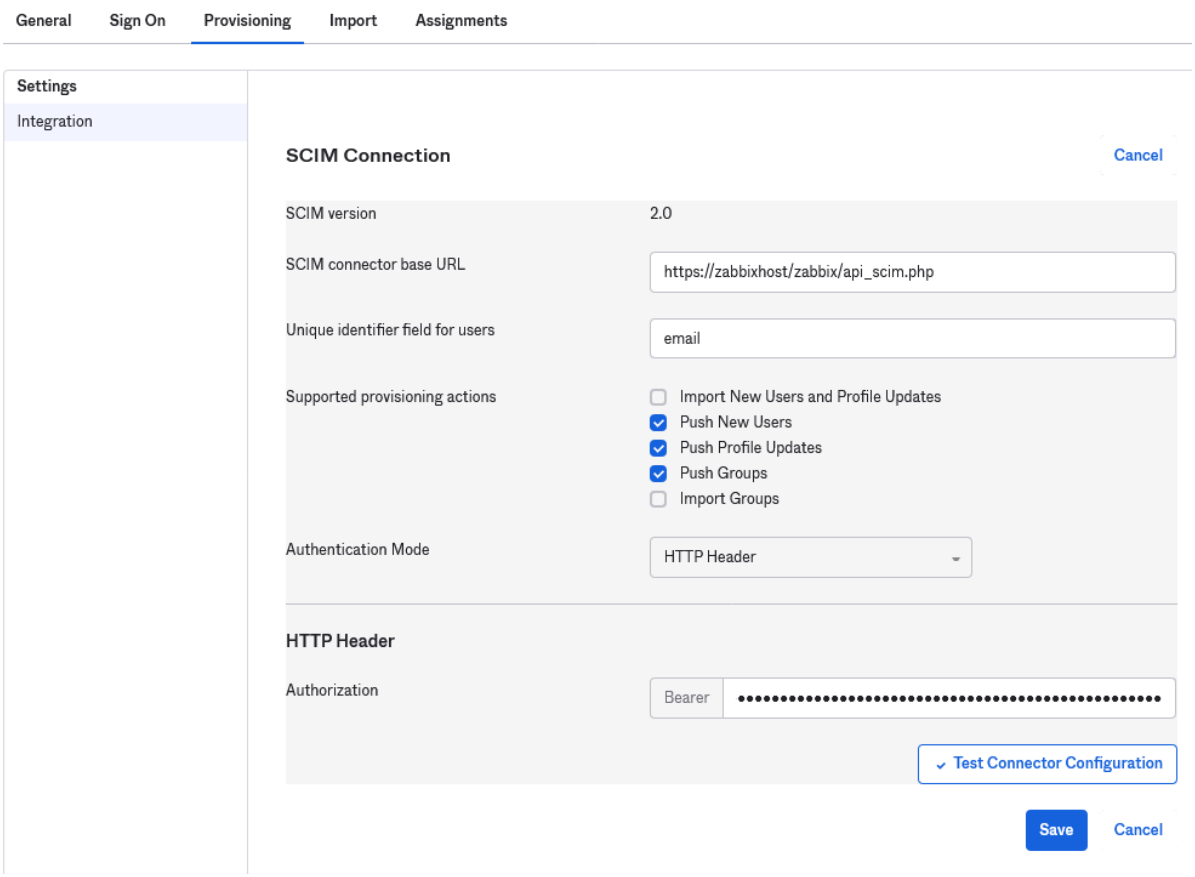
4. Haga clic en el botón *Update* para guardar esta configuración.

Aprovisionamiento SCIM

1. Para activar el aprovisionamiento SCIM, vaya a *General* > *App Settings* de la aplicación en Okta.

Marque la casilla *Enable SCIM provisioning*. Como resultado, aparecerá una nueva pestaña *Provisioning*.

2. Vaya a la pestaña "Provisioning" para configurar una conexión SCIM: - En *SCIM connector base URL*, especifique la ruta al frontend de Zabbix y añada `api_scim.php`, es decir:
 `https://<your-zabbix-url>/zabbix/api_scim.php` - *Unique identifier field for users*: `email` - *Authentication mode*: `HTTP header` - En *Authorization*, introduzca un token de API válido con derechos de Super admin.



Attention:
Si experimenta problemas de autenticación, consulte Authorization header forwarding.

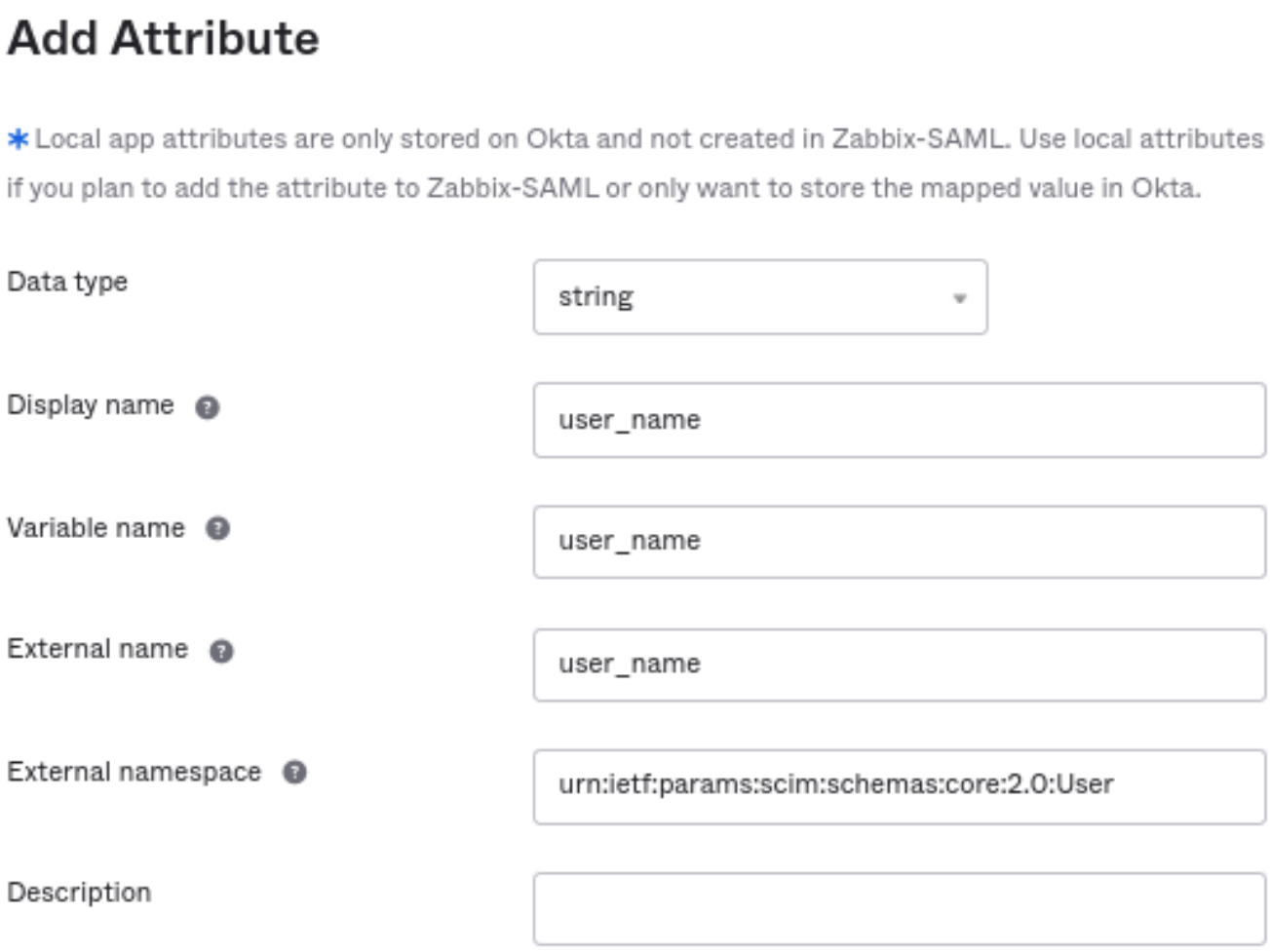
3. Haga clic en *Test Connector Configuration* para probar la conexión. Si todo es correcto, se mostrará un mensaje de confirmación.

4. En *Provisioning* > *To App*, asegúrese de marcar las siguientes casillas: - Create Users - Update User Attributes - Deactivate Users

Esto garantizará que estos tipos de solicitudes se envíen a Zabbix.

5. Asegúrese de que todos los atributos definidos en SAML estén definidos en SCIM. Puede acceder al editor de perfiles de su aplicación en *Provisioning* > *To App*, haciendo clic en *Go to Profile Editor*.

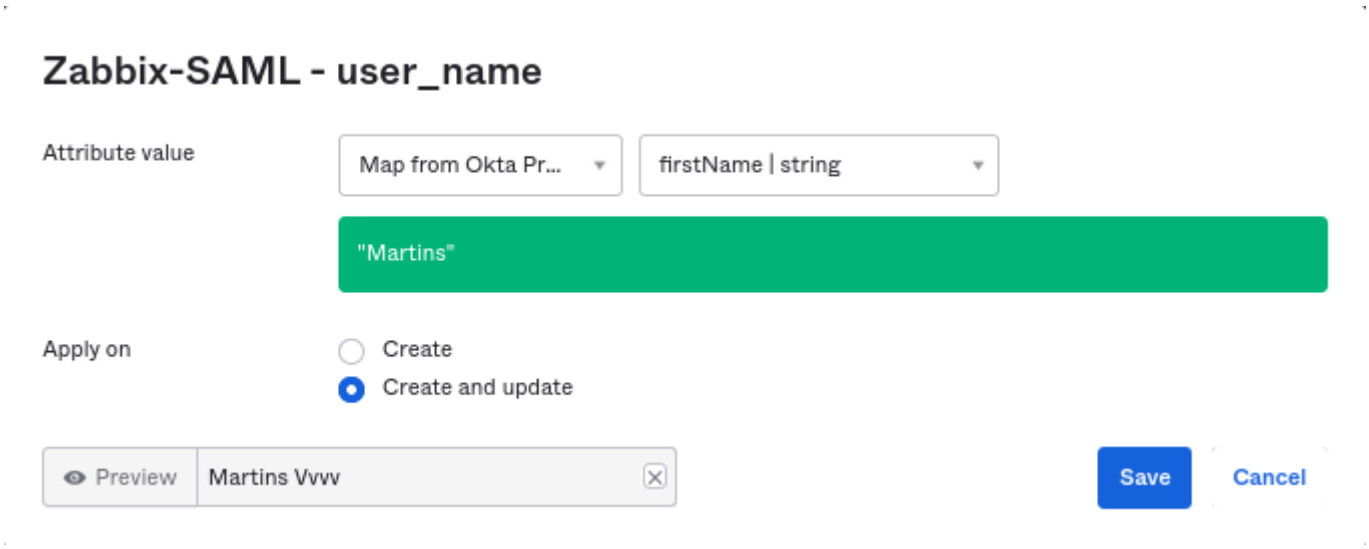
Haga clic en *Add Attribute*. Complete los valores de *Display name*, *Variable name* y *External name* con el nombre del atributo SAML, por ejemplo, `user_name`.



External namespace debe ser el mismo que el esquema de usuario: `urn:ietf:params:scim:schemas:core:2.0:User`

6. Vaya a *Provisioning* > *To App* > *Attribute Mappings* de su aplicación. Haga clic en *Show Unmapped Attributes* en la parte inferior. Aparecerán los atributos añadidos recientemente.

7. Asigne cada atributo añadido.



8. Añada usuarios en la pestaña "Assignments". Antes, los usuarios deben añadirse en *Directory* > *People*. Todas estas asignaciones se enviarán como solicitudes a Zabbix.

9. Añada grupos en la pestaña "Push Groups". El patrón de asignación de grupos de usuarios en la configuración SAML de Zabbix debe coincidir con un grupo especificado aquí. Si no hay ninguna coincidencia, el usuario no podrá crearse en Zabbix.

La información sobre los miembros de los grupos se envía cada vez que se realiza algún cambio.

Configuración de SAML con OneLogin

Descripción general

Esta sección proporciona directrices para configurar el inicio de sesión único y el aprovisionamiento de usuarios en Zabbix desde OneLogin utilizando la autenticación SAML 2.0.

Configuración de OneLogin

Creación de la aplicación

1. Inicie sesión en su cuenta en OneLogin. Para fines de prueba, puede crear una cuenta gratuita de desarrollador en OneLogin.

2. En la interfaz web de OneLogin, vaya a *Applications* > *Applications*.

3. Haga clic en "Add App" y busque la aplicación adecuada. Las directrices de esta página se basan en el ejemplo de la aplicación *SCIM Provisioner with SAML (SCIM v2 Enterprise, full SAML)*.

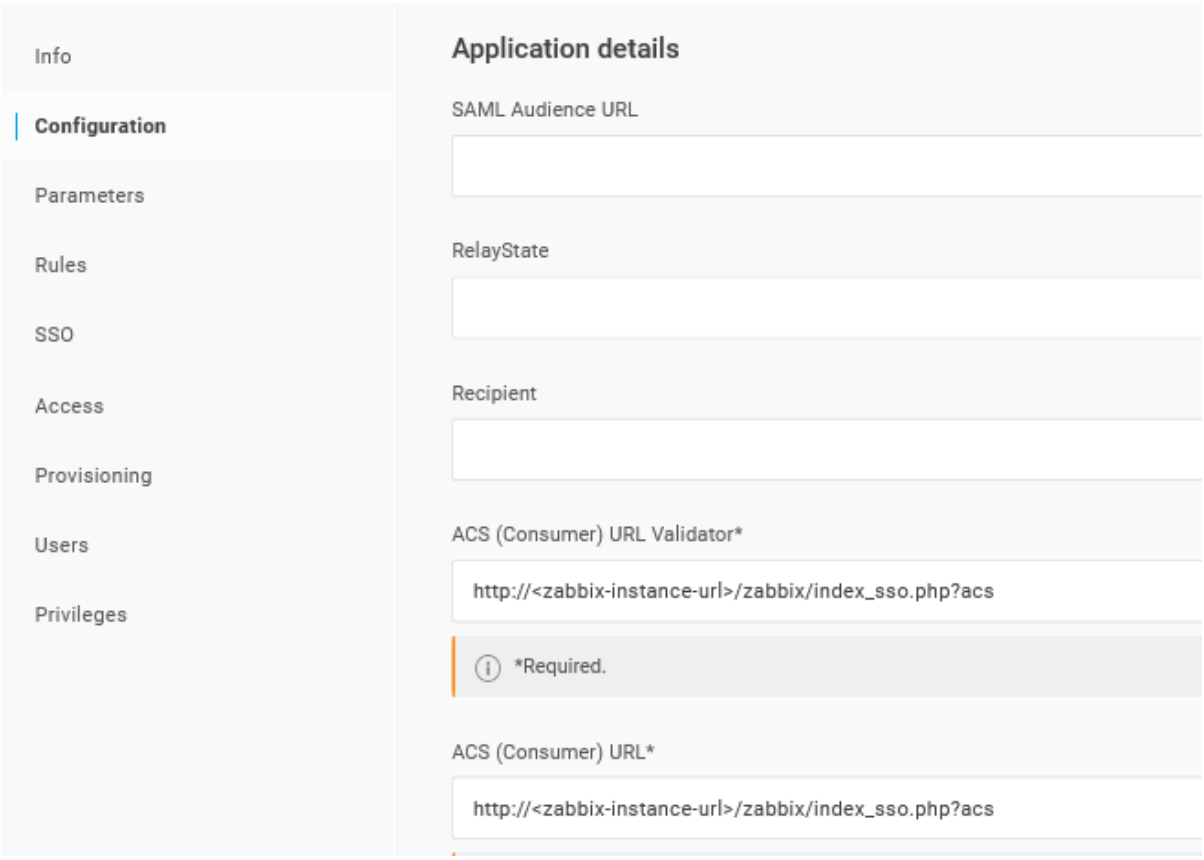
4. Para empezar, es posible que desee personalizar el nombre visible de su aplicación. También puede querer añadir el icono y los detalles de la aplicación. Después de eso, haga clic en *Save*.

Configuración del aprovisionamiento SSO/SCIM

1. En *Configuration* > *Application details*, establezca el endpoint de inicio de sesión único de Zabbix `http://<zabbix-instance-url>/zab` como valor de estos campos:

- *ACS (Consumer) URL Validator*
- *ACS (Consumer) URL*

Tenga en cuenta el uso de "http", y no de "https", para que el parámetro `acs` no se elimine de la solicitud.



También es posible utilizar "https". Para usarlo con Zabbix, añada la siguiente línea al archivo de configuración del frontend (`zabbix.conf.php`):

```
$SSO['SETTINGS'] = ['use_proxy_headers' => true];
```

Deje las demás opciones con sus valores predeterminados.

2. En *Configuration* > *API connection*, establezca los siguientes valores:

- *SCIM Base URL*: `https://<zabbix-instance-url>/zabbix/api_scim.php`
- *SCIM JSON Template*: debe contener todos los atributos personalizados que desee pasar a Zabbix mediante SCIM, como `user_name`, `user_lastname`, `user_email` y `user_mobile`:

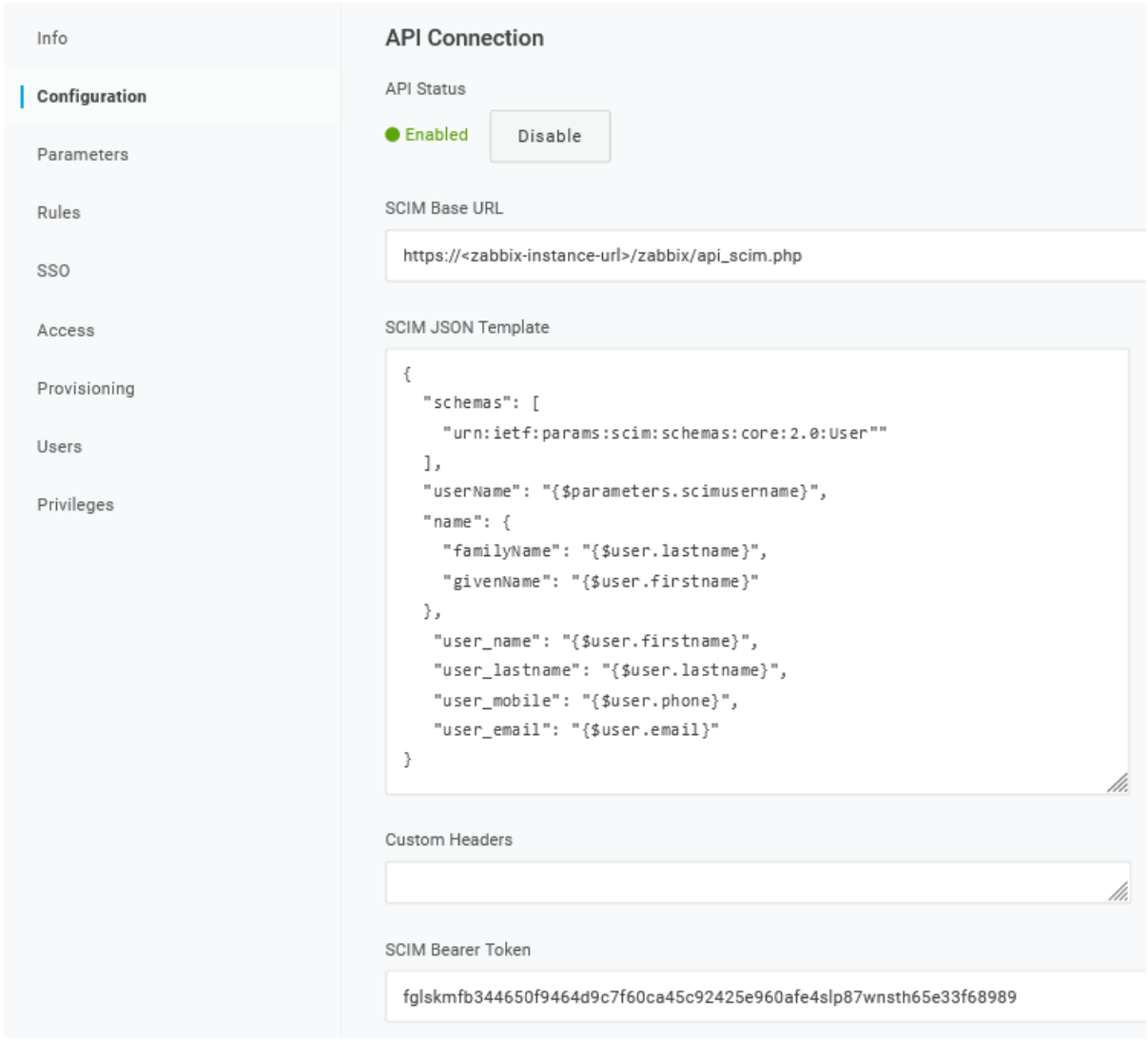
```
{
  "schemas": [
    "urn:ietf:params:scim:schemas:core:2.0:User"
  ],
  "userName": "{$parameters.scimusername}",
  "name": {
    "familyName": "{$user.lastname}",
    "givenName": "{$user.firstname}"
  },
   "user_name": "{$user.firstname}",
   "user_lastname": "{$user.lastname}",
   "user_mobile": "{$user.phone}",
   "user_email": "{$user.email}"
}
```

Los nombres de los atributos son arbitrarios. Se pueden utilizar nombres de atributos diferentes; sin embargo, es necesario que coincidan con el valor del campo correspondiente en la configuración SAML de Zabbix.

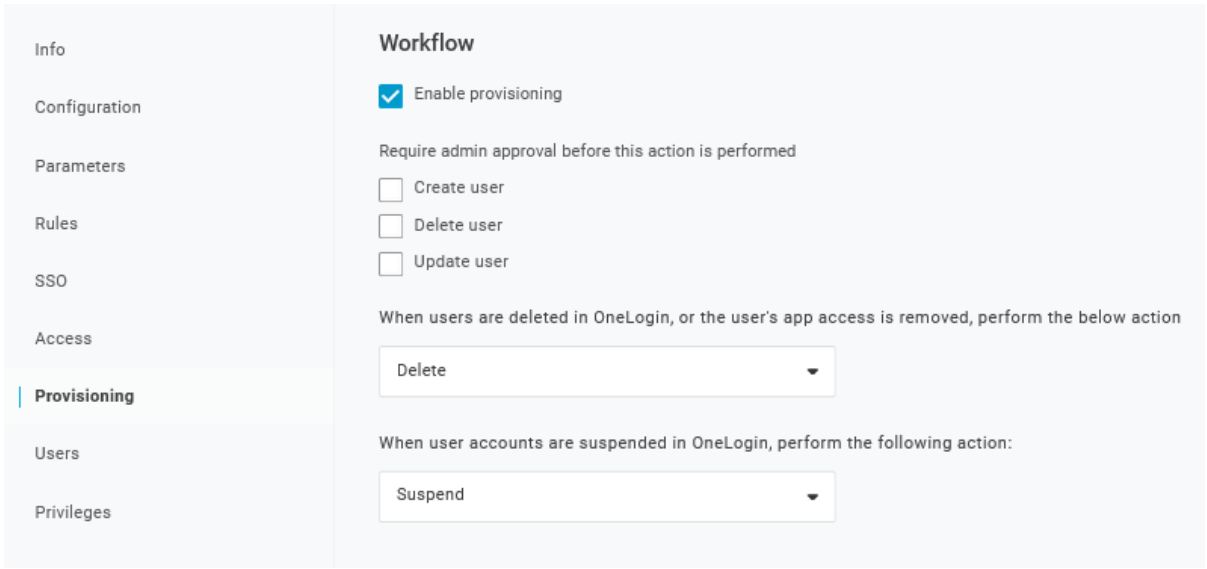
Tenga en cuenta que, para que el aprovisionamiento de usuarios funcione, OneLogin debe recibir en la respuesta un atributo 'name' con 'givenName' y 'familyName', aunque el proveedor de servicios no lo haya solicitado. Por lo tanto, es necesario especificarlo en el esquema, en la sección de configuración de la aplicación.

- *SCIM Bearer Token*: introduzca un token de API de Zabbix con permisos de Super admin.

Haga clic en *Enable* para activar la conexión.

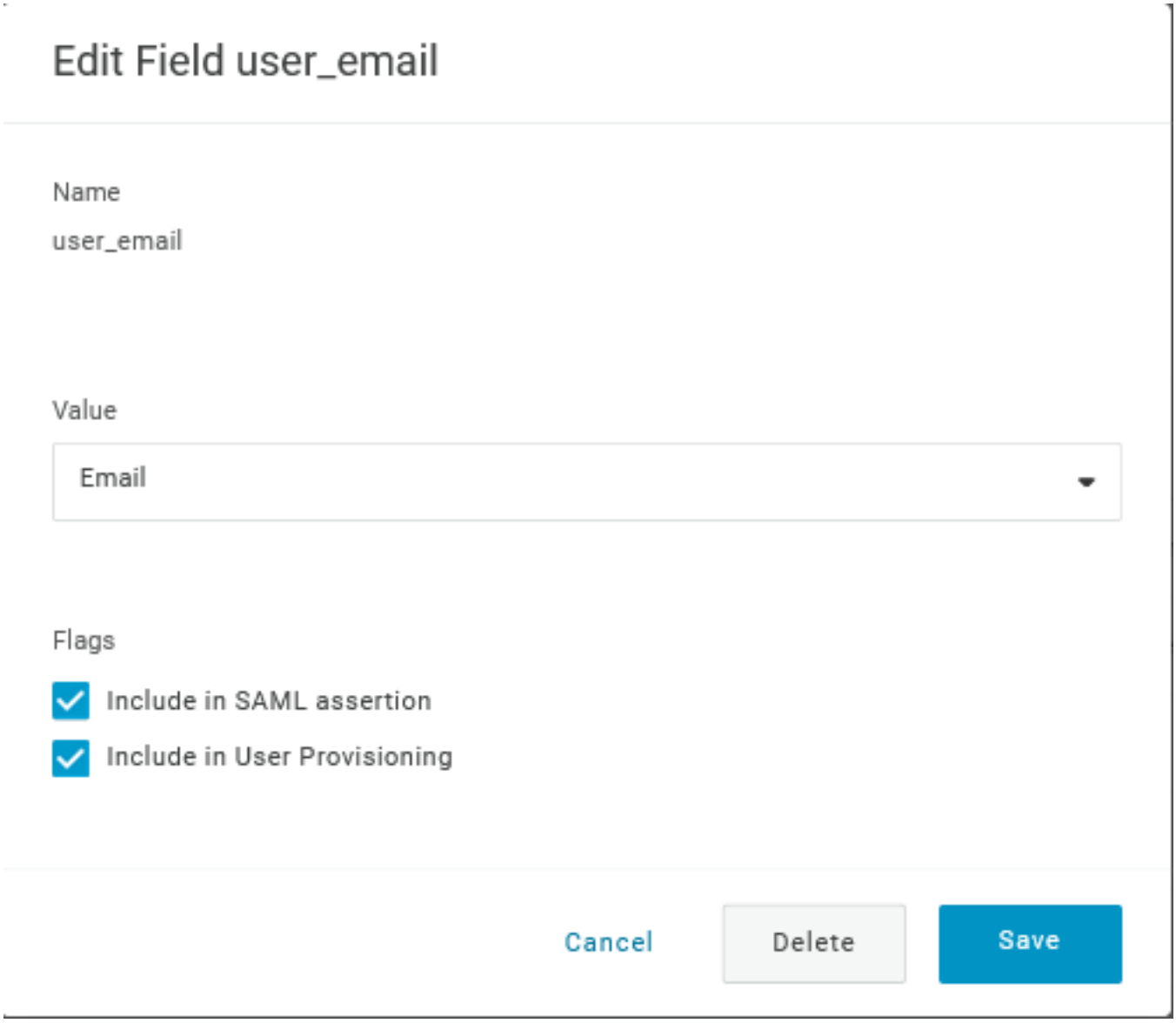


3. En la página *Provisioning*, habilite la opción Provisioning:



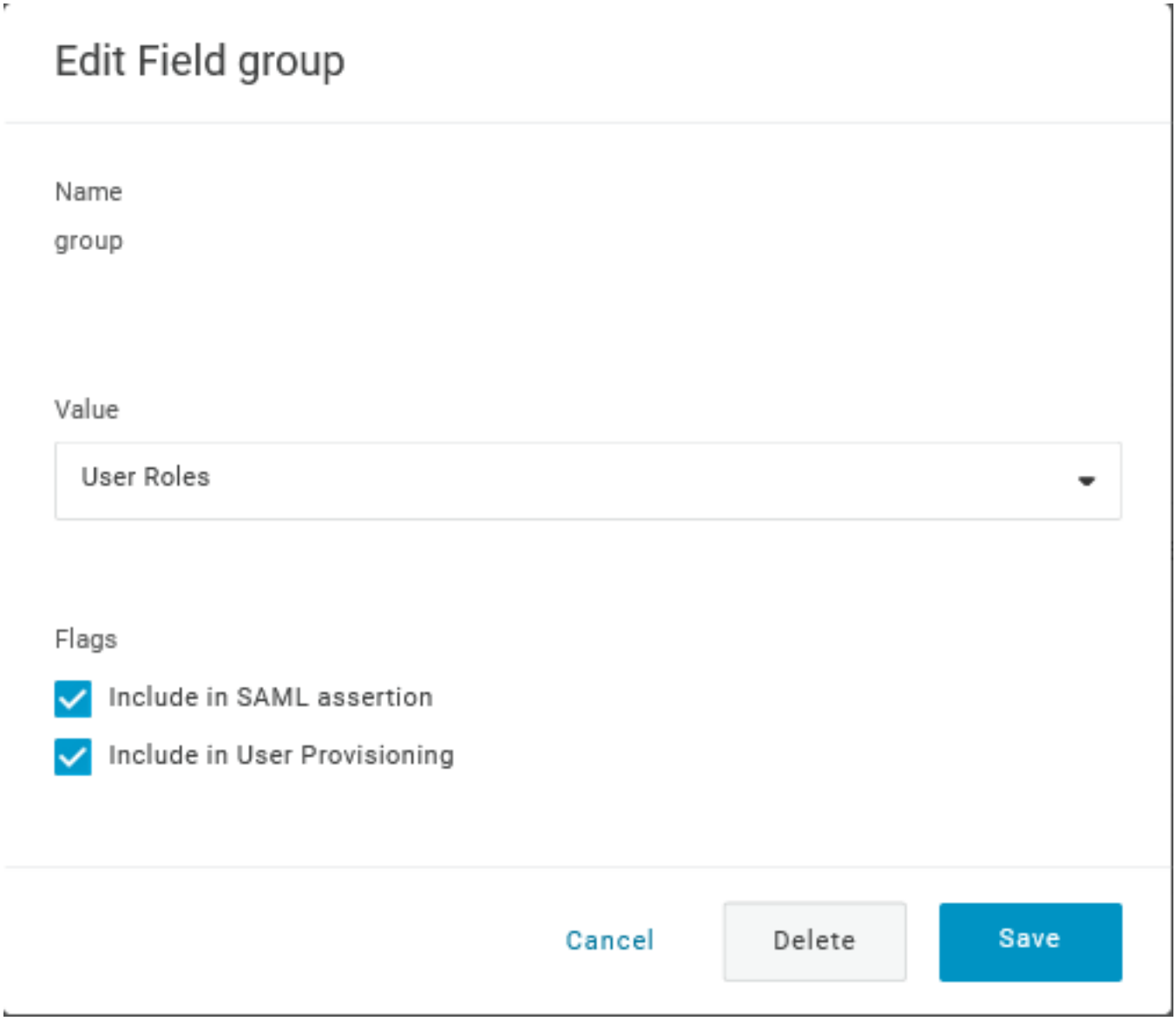
4. La página *Parameters* contiene una lista de parámetros predeterminados:

- Asegúrese de que 'scimusername' coincida con el valor de inicio de sesión del usuario en OneLogin (por ejemplo, el correo electrónico);
- Marque la opción *Include in User Provisioning* para el parámetro 'Groups';
- Haga clic en "+" para crear los parámetros personalizados necesarios para las aserciones SAML y el aprovisionamiento de usuarios, como `user_name`, `user_lastname`, `user_email` y `user_mobile`:

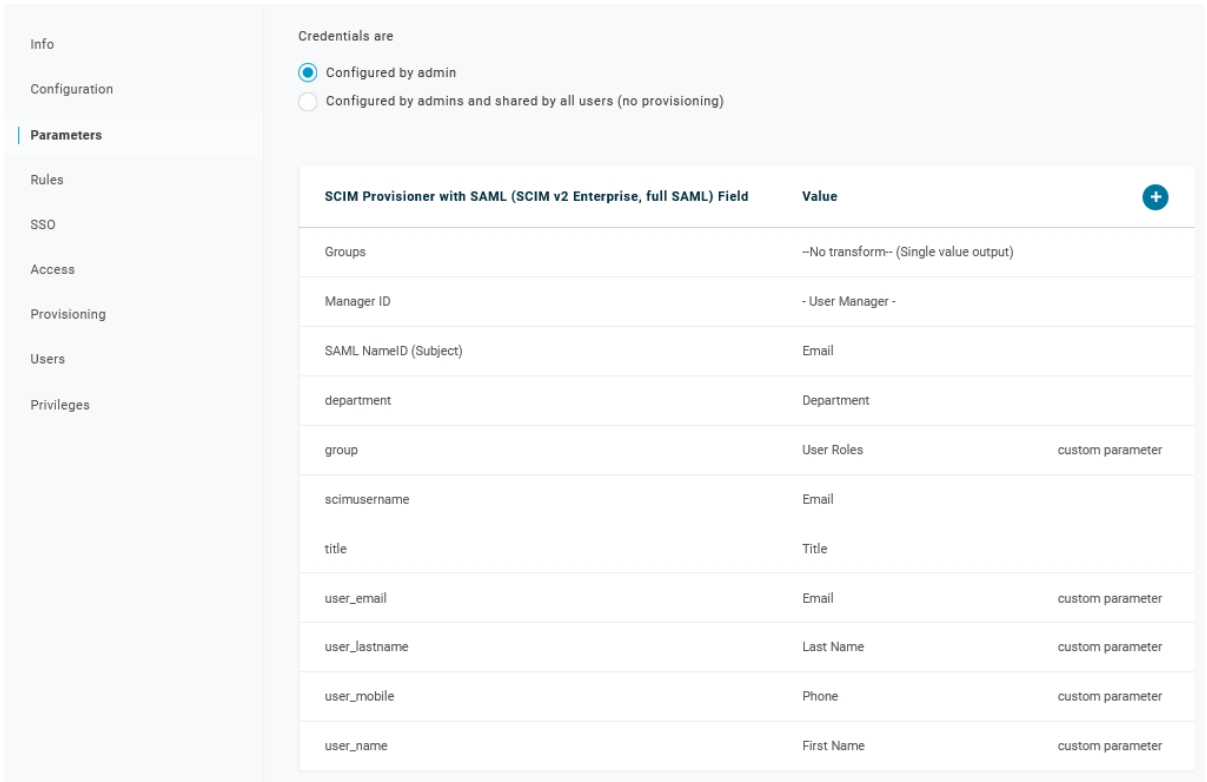


Al añadir un parámetro, asegúrese de marcar las opciones *Include in SAML assertion* y *Include in User Provisioning*.

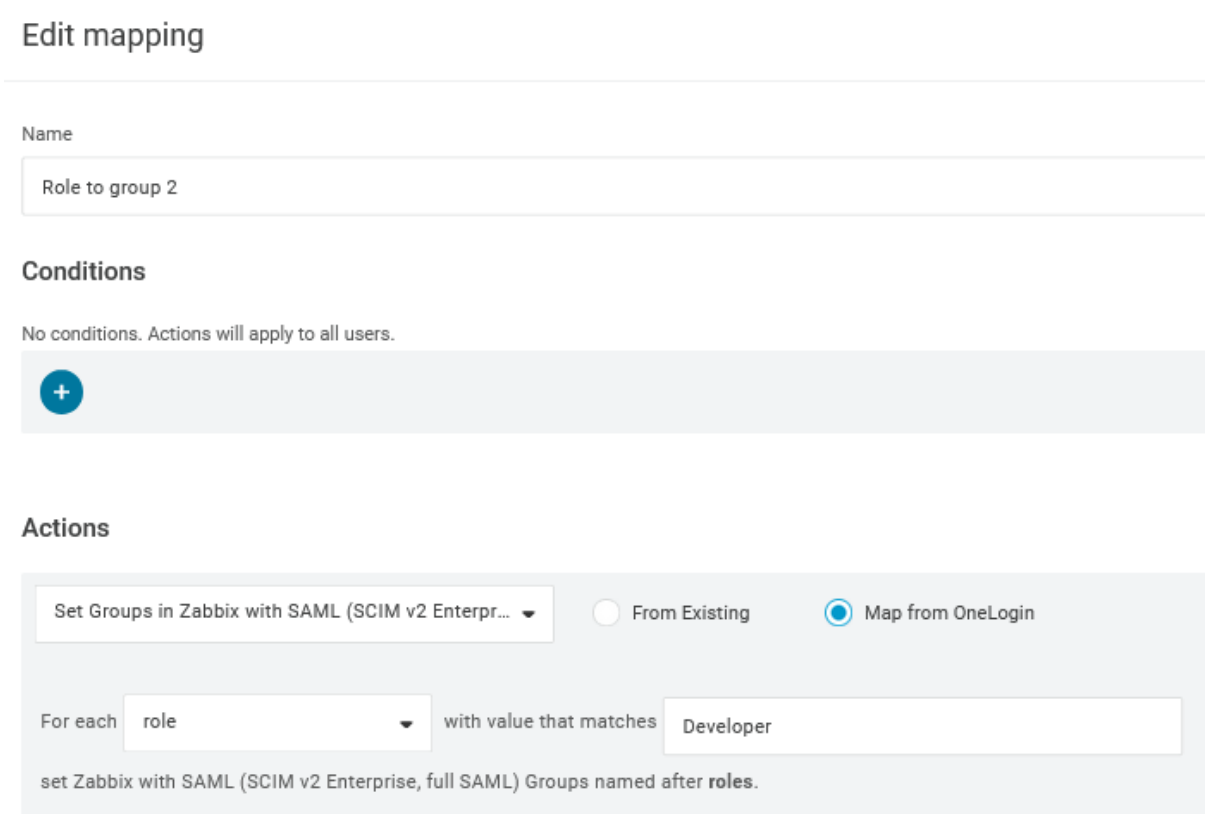
- Añada un parámetro 'group' que coincida con los roles de usuario en OneLogin. Los roles de usuario se pasarán como una cadena, separados por un punto y coma `;`. Los roles de usuario de OneLogin se utilizarán para crear grupos de usuarios en Zabbix:



Verifique la lista de parámetros:



5. En la página *Rules*, cree asignaciones de roles de usuario al parámetro predeterminado Groups.



Puede utilizar una expresión regular para pasar roles específicos como grupos. Los nombres de los roles no deben contener `;`, ya que OneLogin lo utiliza como separador al enviar un atributo con varios roles.

6. Descargue el certificado del IdP. Establezca los permisos 644 ejecutando:

```
chmod 644 idp.crt
```

Configuración de Zabbix

1. En Zabbix, vaya a Configuración de SAML y complete las opciones de configuración según la configuración de OneLogin:



Campo de Zabbix	Campo de configuración en OneLogin	Valor de ejemplo
IdP entity ID	Issuer URL (consulte la pestaña SSO de su aplicación en OneLogin)	

Campo de Zabbix	Campo de configuración en OneLogin	Valor de ejemplo
SSO service URL	SAML 2.0 Endpoint (HTTP) (consulte la pestaña SSO de su aplicación en OneLogin)	
SLO service URL	SLO Endpoint (HTTP) (consulte la pestaña SSO de su aplicación en OneLogin)	
Username attribute	Custom parameter	`user_email`
Group name attribute	Custom parameter	`group`
User name attribute	Custom parameter	`user_name`
User last name attribute	Custom parameter	`user_lastname`

También es necesario configurar la asignación de grupos de usuarios.
La asignación de medios es opcional.
Haga clic en *Update* para guardar esta configuración.

2. Añada el certificado Base64 proporcionado por OneLogin.

Si `$SSO['CERT_STORAGE'] = 'database'` está establecido en el archivo de configuración del frontend (`zabbix.conf.php`), puede pegar el texto del certificado o cargar el archivo de certificado en el frontend durante la configuración de SAML; no se requieren archivos en el sistema de archivos.

Si `$SSO['CERT_STORAGE'] = 'file'` está establecido en el archivo de configuración, el certificado debe estar disponible en el sistema de archivos (de forma predeterminada, en `ui/conf/certs` o en la ruta configurada en el archivo de configuración), y el frontend no almacenará los certificados en la base de datos.
Tenga en cuenta que, si `$SSO['CERT_STORAGE']` no está establecido o está comentado, se asumirá el almacenamiento en archivos y los certificados se leerán desde `ui/conf/certs`.

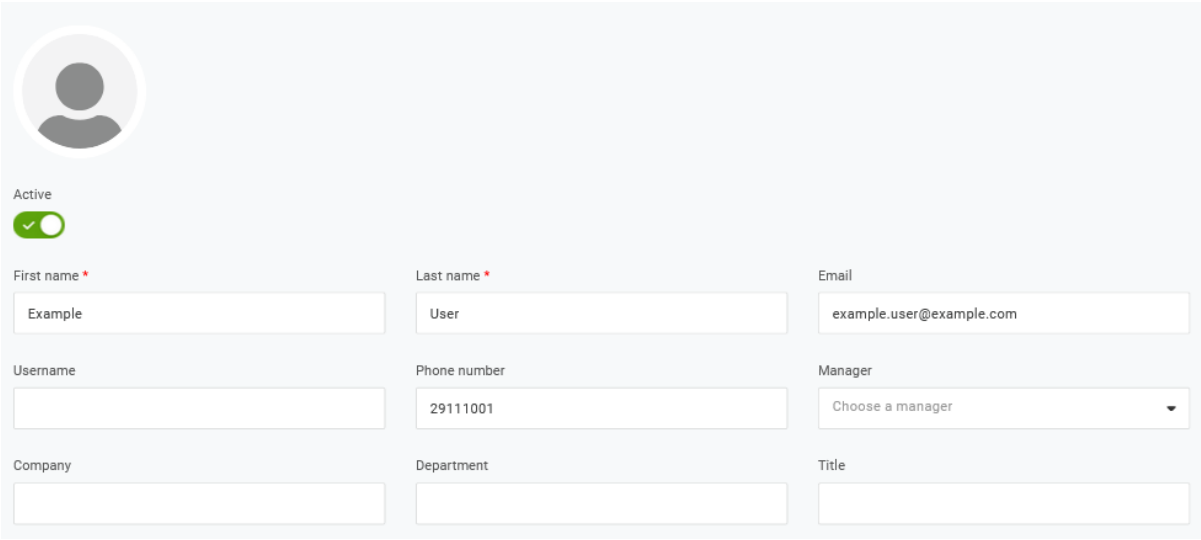
Puede acceder a la descarga del certificado en OneLogin desde *Applications* > *SSO* > haga clic en *View details* debajo del certificado actual.

3. Haga clic en el botón *Update* para guardar esta configuración.

Aprovisionamiento de usuarios SCIM

Con el aprovisionamiento de usuarios habilitado, ahora es posible agregar/actualizar usuarios y sus roles en OneLogin y hacer que se aprovisionen inmediatamente en Zabbix.

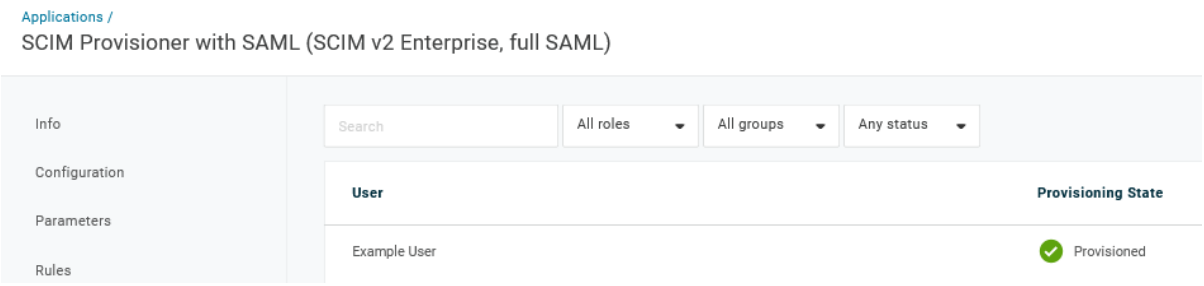
Por ejemplo, puede crear un nuevo usuario:



Agrégelo a un rol de usuario y a la aplicación que aprovisionará al usuario:



Al guardar el usuario, se aprovisionará en Zabbix. En *Application* > *Users* puede comprobar el estado de aprovisionamiento de los usuarios actuales de la aplicación:



Si se aprovisiona correctamente, el usuario podrá verse en la lista de usuarios de Zabbix.



Username ▲	Name	Last name	User role	Groups	Is online?	Login	Frontend access
Admin	Zabbix	Administrator	Super admin role	Zabbix administrators	Yes (2023-04-18 21:11:43)	Ok	System default
example.user@example.com	Example	User	Admin role	Zabbix administrators	No	Ok	SAML

Configuración de informes programados

Descripción general

Esta sección proporciona instrucciones sobre cómo instalar el servicio web de Zabbix y configurar Zabbix para habilitar la generación de informes programados.

Instalación

Se debe instalar un nuevo proceso de servicio web de Zabbix y el navegador Google Chrome para habilitar la generación de informes programados. El servicio web puede instalarse en la misma máquina donde está instalado el servidor Zabbix o en una máquina diferente. El navegador Google Chrome debe instalarse en la misma máquina donde se instala el servicio web.

El paquete oficial zabbix-web-service está disponible en el repositorio de Zabbix. El navegador Google Chrome no está incluido en estos paquetes y debe instalarse por separado.

Para compilar Zabbix web service desde el código fuente, consulte Installing Zabbix web service.

Después de la instalación, ejecute `zabbix_web_service` en la máquina donde está instalado el web service:

```
zabbix_web_service
```

Configuración

Para garantizar una comunicación adecuada entre todos los elementos involucrados, asegúrese de que los parámetros de configuración del server y configuración del frontend estén configurados correctamente.

Zabbix server

Es necesario actualizar los siguientes parámetros en el archivo de configuración de Zabbix server: `WebServiceURL` y `StartReportWriters`.

WebServiceURL

Este parámetro es necesario para habilitar la comunicación con el servicio web.
La URL debe tener el formato `http[s]://host:port/report`.

- De forma predeterminada, el servicio web escucha en el puerto 10053.
  Se puede especificar un puerto diferente en el archivo de configuración del servicio web.
- Es obligatorio especificar la ruta `/report` (la ruta está codificada y no se puede cambiar).

Ejemplo:

```
WebServiceURL=http://localhost:10053/report
```

Attention:
Se recomienda encarecidamente configurar el cifrado entre Zabbix server y el servicio web de Zabbix mediante certificados.
De forma predeterminada, los datos transmitidos entre Zabbix server y el servicio web de Zabbix no están cifrados, lo que puede provocar accesos no autorizados.

StartReportWriters

Este parámetro determina cuántos procesos de generación de informes deben iniciarse.
Si no está definido o es igual a 0, la generación de informes está deshabilitada.
Según el número y la frecuencia de los informes necesarios, es posible habilitar entre 1 y 100 procesos de generación de informes.

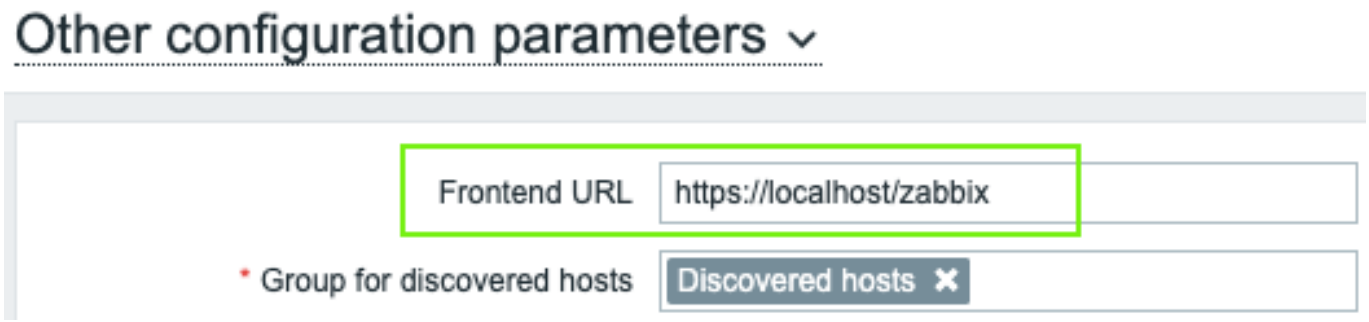
Ejemplo:

```
StartReportWriters=3
```

Frontend de Zabbix

Se debe establecer un parámetro *Frontend URL* para habilitar la comunicación entre el frontend de Zabbix y el servicio web de Zabbix:

- Vaya a la sección de menú del frontend *Administración > General > Otros*
- Especifique la URL completa de la interfaz web de Zabbix en el parámetro *Frontend URL*



Note:
Una vez completado el procedimiento de configuración, puede configurar y enviar un informe de prueba para asegurarse de que todo funciona correctamente.

Items

Esta sección contiene información de referencia y notas técnicas sobre items, incluidos detalles de parámetros para claves de item específicas, detalles específicos de la plataforma y notas sobre configuraciones especiales de item.

Utilice la barra lateral para encontrar la información relevante para su caso de uso.

Parámetros de vm.memory.size

Descripción general

Esta sección proporciona algunos detalles de los parámetros para la métrica de agente vm.memory.size[<mode>].

Parámetros

Los siguientes parámetros están disponibles para este item:

- **active** - memoria actualmente en uso o usada muy recientemente, por lo que está en RAM
- **anon** - memoria no asociada a un archivo (no se puede volver a leer desde él)
- **available** - memoria disponible, calculada de manera diferente dependiendo de la plataforma (ver la tabla a continuación)
- **buffers** - caché para cosas como metadatos del sistema de archivos
- **cached** - caché para varias cosas
- **exec** - código ejecutable, típicamente de un archivo (programa)
- **file** - caché para el contenido de archivos accedidos recientemente
- **free** - memoria que está fácilmente disponible para cualquier entidad que la solicite
- **inactive** - memoria que está marcada como no utilizada
- **pavailable** - memoria 'disponible' como porcentaje del 'total' (calculado como `available/total*100`)
- **pinned** - igual que 'wired'
- **pused** - memoria 'usada' como porcentaje del 'total' (calculado como `used/total*100`)
- **shared** - memoria que puede ser accedida simultáneamente por múltiples procesos
- **slab** - cantidad total de memoria utilizada por el kernel para almacenar en caché estructuras de datos para su propio uso
- **total** - memoria física total disponible

- **used** - memoria usada, calculada de manera diferente dependiendo de la plataforma (ver la tabla a continuación)
- **wired** - memoria que está marcada para permanecer siempre en RAM. Nunca se mueve al disco.

Warning:
Algunos de estos parámetros son específicos de la plataforma y pueden no estar disponibles en su plataforma. Consulte items de Zabbix agent para más detalles.

Cálculo específico de la plataforma de **available** y **used**:

Plataforma	**"available"**	**"used"**
AIX	free + cached	memoria real en uso
FreeBSD	inactive + cached + free	active + wired + cached
HP UX	free	total - free
Linux<3.14	free + buffers + cached	total - free
Linux 3.14+ (también retroportado a 3.10 en RHEL 7)	/proc/meminfo, consulte "MemAvailable" en la documentación del kernel de Linux para más detalles. Tenga en cuenta que free + buffers + cached ya no es igual a 'available' debido a que no toda la caché de páginas se puede liberar y se utiliza un valor mínimo en el cálculo.	total - free
NetBSD	inactive + execpages + file + free	total - free
OpenBSD	inactive + free + cached	active + wired
OSX	inactive + free	active + wired
Solaris	free	total - free
Win32	free	total - free

Attention:
La suma de *vm.memory.size[used]* y *vm.memory.size[available]* no necesariamente es igual al total. Por ejemplo, en FreeBSD:
 * Las memorias activas, inactivas, cableadas y en caché se consideran usadas, porque almacenan información útil.
 * Al mismo tiempo, las memorias inactivas, en caché y libres se consideran disponibles, porque este tipo de memorias pueden ser entregadas instantáneamente a los procesos que solicitan más memoria.
Por lo tanto, la memoria inactiva es tanto usada como disponible simultáneamente. Debido a esto, el item *vm.memory.size[used]* está diseñado solo con fines informativos, mientras que *vm.memory.size[available]* está diseñado para ser utilizado en triggers.

Véase también

1. Detalles adicionales sobre el cálculo de memoria en diferentes SO

Nivel mínimo de permisos para items del agent de Windows

Descripción general

Al monitorizar sistemas utilizando un agente, una buena práctica es obtener métricas del equipo en el que está instalado el agente. Para aplicar el principio de menor privilegio, es necesario determinar qué métricas se obtienen del agente.

La tabla en este documento le permite seleccionar los derechos mínimos para el funcionamiento correcto garantizado del agente Zabbix.

Si se selecciona un usuario diferente para que el agente funcione, en lugar de 'LocalSystem', entonces para el funcionamiento del agente como un servicio de Windows, el nuevo usuario debe tener los derechos "Iniciar sesión como un servicio" desde "Política local→Asignación de derechos de usuario" y el derecho a crear, escribir y eliminar el archivo de registro del agente Zabbix. Un usuario de Active Directory debe ser añadido al grupo *Usuarios de Monitor de Rendimiento*.

Note:
Al trabajar con los derechos de un agente basados en el grupo "mínimamente aceptable técnicamente", se requiere la provisión previa de derechos a los objetos para la monitorización.

Items comunes de agent compatibles en Windows

Item key	Grupo de usuarios	
	Recomendado	Mínimo técnicamente aceptable (la funcionalidad es limitada)
agent.hostname	Invitados	Invitados
agent.ping	Invitados	Invitados
agent.variant	Invitados	Invitados
agent.version	Invitados	Invitados
log	Administradores	Invitados
log.count	Administradores	Invitados
logrt	Administradores	Invitados
logrt.count	Administradores	Invitados
net.dns	Invitados	Invitados
net.dns.perf	Invitados	Invitados
net.dns.record	Invitados	Invitados
net.if.discovery	Invitados	Invitados
net.if.in	Invitados	Invitados
net.if.out	Invitados	Invitados
net.if.total	Invitados	Invitados
net.tcp.listen	Invitados	Invitados
net.tcp.port	Invitados	Invitados
net.tcp.service	Invitados	Invitados
net.tcp.service.perf	Invitados	Invitados
net.udp.service	Invitados	Invitados
net.udp.service.perf	Invitados	Invitados
proc.num	Administradores	Invitados
system.cpu.discovery	Usuarios del Monitor de rendimiento	Usuarios del Monitor de rendimiento
system.cpu.load	Usuarios del Monitor de rendimiento	Usuarios del Monitor de rendimiento
system.cpu.num	Invitados	Invitados
system.cpu.util	Usuarios del Monitor de rendimiento	Usuarios del Monitor de rendimiento
system.hostname	Invitados	Invitados
system.localtime	Invitados	Invitados
system.run	Administradores	Invitados
system.sw.arch	Invitados	Invitados
system.swap.size	Invitados	Invitados
system.uname	Invitados	Invitados
system.uptime	Usuarios del Monitor de rendimiento	Usuarios del Monitor de rendimiento
vfs.dir.count	Administradores	Invitados
vfs.dir.get	Administradores	Invitados
vfs.dir.size	Administradores	Invitados
vfs.file.cksum	Administradores	Invitados
vfs.file.contents	Administradores	Invitados
vfs.file.exists	Administradores	Invitados
vfs.file.md5sum	Administradores	Invitados
vfs.file.regexp	Administradores	Invitados
vfs.file.regmatch	Administradores	Invitados
vfs.file.size	Administradores	Invitados
vfs.file.time	Administradores	Invitados
vfs.fs.discovery	Administradores	Invitados
vfs.fs.size	Administradores	Invitados
vfs.fs.get	Administradores	Invitados
vm.memory.size	Invitados	Invitados
web.page.get	Invitados	Invitados
web.page.perf	Invitados	Invitados
web.page.regexp	Invitados	Invitados
zabbix.stats	Invitados	Invitados

Claves de item específicas de Windows

Item key	User group	
	Recomendado	Mínimo técnicamente aceptable (la funcionalidad es limitada)

Item key	User group	
eventlog	Lectores del registro de eventos	Invitados
net.if.list	Invitados	Invitados
perf_counter	Usuarios del Monitor de rendimiento	Usuarios del Monitor de rendimiento
proc_info	Administradores	Invitados
service.discovery	Invitados	Invitados
service.info	Invitados	Invitados
services	Invitados	Invitados
wmi.get	Administradores	Invitados
vm.vmemory.size	Invitados	Invitados

Codificación de los valores devueltos

El server de Zabbix espera que cada valor de texto devuelto esté codificado en UTF8. Esto se aplica a cualquier tipo de comprobación: Zabbix agent, SSH, Telnet, etc.

Distintos sistemas/dispositivos supervisados y comprobaciones pueden devolver caracteres no ASCII en el valor. En esos casos, casi todas las claves posibles de **zabbix** contienen un parámetro adicional de clave de item: **<encoding>**. Este parámetro de clave es opcional, pero debe especificarse si el valor devuelto no está codificado en UTF8 y contiene caracteres no ASCII. De lo contrario, el resultado puede ser inesperado e impredecible.

A continuación se describe el comportamiento con distintos backends de base de datos en estos casos.

MySQL

Si un valor contiene un carácter no ASCII en una codificación que no sea UTF8, este carácter y los siguientes se descartarán cuando la base de datos almacene este valor. No se escribirán mensajes de advertencia en el `zabbix_server.log`.
 Aplicable al menos a la versión 5.1.61 de MySQL.

PostgreSQL

Si un valor contiene un carácter no ASCII en una codificación que no sea UTF8, esto provocará que la consulta SQL falle (PGRES_FATAL_ERROR:ERROR invalid byte sequence for encoding) y los datos no se almacenarán. Se escribirá un mensaje de advertencia apropiado en `zabbix_server.log`.
 Aplicable al menos a la versión 9.1.3 de PostgreSQL.

Compatibilidad con archivos grandes

La compatibilidad con archivos grandes, a menudo abreviada como LFS, es el término aplicado a la capacidad de trabajar con archivos de más de 2 GB en sistemas operativos de 32 bits. La compatibilidad con archivos grandes afecta al menos a la supervisión de archivos de registro y a todos los items vfs.file.*. La compatibilidad con archivos grandes depende de las capacidades de un sistema en el momento de la compilación de Zabbix, pero está completamente deshabilitada en Solaris de 32 bits debido a su incompatibilidad con procfs y swapctl.

Sensor

Descripción general

Cada chip de sensor obtiene su propio directorio en el árbol sysfs `/sys/devices`. Para encontrar todos los chips de sensores, es más fácil seguir los enlaces simbólicos de dispositivos desde `/sys/class/hwmon/hwmon*`, donde `*` es un número real (0,1,2,...).

Las lecturas de los sensores se encuentran en el directorio `/sys/class/hwmon/hwmon*/` para dispositivos virtuales, o en el directorio `/sys/class/hwmon/hwmon*/device` para dispositivos no virtuales. Un archivo, llamado `name`, ubicado dentro de los directorios `hwmon*` o `hwmon*/device` contiene el nombre del chip, que corresponde al nombre del controlador del kernel utilizado por el chip del sensor.

Solo hay un valor de lectura del sensor por archivo. El esquema común para nombrar los archivos que contienen lecturas de sensores dentro de cualquiera de los directorios mencionados anteriormente es: `<type><number>_<item>`, donde

- **type** - para chips de sensores es ”in” (voltaje), ”temp” (temperatura), ”fan” (ventilador), etc.
- **item** - ”input” (valor medido), ”max” (umbral alto), ”min” (umbral bajo), etc.
- **number** - siempre se usa para elementos que pueden estar presentes más de una vez (generalmente comienza desde 1, excepto para voltajes que comienzan desde 0). Si los archivos no se refieren a un elemento específico, tienen un nombre simple sin número.

La información sobre los sensores disponibles en el host se puede obtener utilizando las herramientas **sensors-detect** y **sensors** (paquete lm-sensors). **Sensors-detect** ayuda a determinar qué módulos son necesarios para los sensores disponibles. Cuando los módulos están cargados, el programa **sensors** se puede usar para mostrar las lecturas de todos los chips de sensores. El etiquetado de las lecturas de los sensores, utilizado por este programa, puede ser diferente del esquema de nombres común (`<type><number>_<item>`):

- si hay un archivo llamado `<type><number>_label`, entonces la etiqueta dentro de este archivo se usará en lugar del nombre `<type><number><item>`;
- si no hay un archivo `<type><number>_label`, entonces el programa busca dentro de `/etc/sensors.conf` (también podría ser `/etc/sensors3.conf`, o diferente) la sustitución del nombre.

Este etiquetado permite al usuario determinar qué tipo de hardware se utiliza. Si no hay ni un archivo `<type><number>_label` ni una etiqueta dentro del archivo de configuración, el tipo de hardware se puede determinar por el atributo name (`hwmon*/device/name`). Los nombres reales de los sensores, que acepta `zabbix_agent`, se pueden obtener ejecutando el programa **sensors** con el parámetro `-u` (`sensors -u`).

En el programa **sensors** los sensores disponibles se separan por tipo de bus (adaptador ISA, adaptador PCI, adaptador SPI, dispositivo virtual, interfaz ACPI, adaptador HID).

Obtención de identificadores de sensores

Las etiquetas de los sensores, tal como las imprime el comando `sensors`, no siempre pueden usarse directamente porque el etiquetado varía entre los fabricantes de chips. Por ejemplo, la salida de `sensors` podría contener las siguientes líneas:

```
sensors
in0:          +2.24 V  (min =  +0.00 V, max =  +3.32 V)
Vcore:        +1.15 V  (min =  +0.00 V, max =  +2.99 V)
+3.3V:        +3.30 V  (min =  +2.97 V, max =  +3.63 V)
+12V:        +13.00 V  (min =  +0.00 V, max = +15.94 V)
M/B Temp:     +30.0°C  (low  = -127.0°C, high = +127.0°C)
```

De estos, solo un nombre de visualización corresponde directamente a un identificador interno:

```
zabbix_get -s 127.0.0.1 -k sensor[lm85-i2c-0-2e,in0]
2.240000
```

Intentar usar otras etiquetas de visualización (como `Vcore` o `+12V`) no funcionará:

```
zabbix_get -s 127.0.0.1 -k sensor[lm85-i2c-0-2e,Vcore]
ZBX_NOTSUPPORTED
```

Para encontrar el identificador interno del sensor que acepta Zabbix, ejecute `sensors -u`. En la salida, se puede observar lo siguiente:

```
sensors -u
...
Vcore:
  in1_input: 1.15
  in1_min: 0.00
  in1_max: 2.99
  in1_alarm: 0.00
...
+12V:
  in4_input: 13.00
  in4_min: 0.00
  in4_max: 15.94
  in4_alarm: 0.00
...
```

Así, la etiqueta `Vcore` corresponde al identificador `in1`, y `+12V` corresponde a `in4`. De acuerdo con la especificación de hwmon sysfs del kernel, estos son voltajes en los pines del chip y, en general, pueden necesitar escalado.

```
zabbix_get -s 127.0.0.1 -k sensor[lm85-i2c-0-2e,in1]
1.301000
```

Zabbix puede recuperar no solo voltaje (`in`), sino también lecturas de corriente (`curr`), temperatura (`temp`) y velocidad del ventilador (`fan`).

Compatibilidad

Distribuciones antiguas

Las lecturas de los sensores se obtienen del directorio `/proc/sys/dev/sensors`:

- **device** - nombre del dispositivo (si se utiliza <mode>, es una expresión regular)
- **sensor** - nombre del sensor (si se utiliza <mode>, es una expresión regular)
- **mode** - valores posibles: `avg`, `max`, `min` (si se omite este parámetro, el dispositivo y el sensor se tratan literalmente)

Ejemplo de clave: `sensor[w83781d-i2c-0-2d,temp1]`.

Distribuciones actuales

Las lecturas de los sensores se obtienen del directorio `/sys/class/hwmon`:

- **device** - nombre del dispositivo (no es una expresión regular). El nombre del dispositivo puede ser el nombre real del dispositivo (por ejemplo, `0000:00:18.3`) o el nombre adquirido utilizando el programa sensors (por ejemplo, `k8temp-pci-00c3`). Depende del usuario elegir qué nombre usar.
- **sensor** - nombre del sensor (no es una expresión regular).
- **mode** - valores posibles: `avg`, `max`, `min` (si se omite este parámetro, el dispositivo y el sensor se tratan literalmente).

Ejemplos de claves:

`sensor[k8temp-pci-00c3,temp,max]` o `sensor[0000:00:18.3,temp1]`.

`sensor[smsc47b397-isa-0880,in,avg]` o `sensor[smsc47b397.2176,in1]`.

Notas sobre el parámetro memtype en items proc.mem

Descripción general

El parámetro **memtype** es compatible con las plataformas Linux, AIX, FreeBSD y Solaris.

En todas estas plataformas se admiten tres valores comunes de `memtype`: `pmem`, `rss` y `vsize`. Además, algunas plataformas admiten valores de `memtype` específicos de la plataforma.

AIX

Consulte en la tabla los valores compatibles con el parámetro `memtype` en AIX.

Valor compatible	Descripción	Origen en la estructura procentry64	Intenta ser compatible con
`vsize` [1]	Tamaño de la memoria virtual	`pi_size`	
`pmem`	Porcentaje de memoria real	`pi_prm`	`ps -o pmem`
`rss`	Tamaño del conjunto residente	`pi_trss + pi_drss`	`ps -o rssize`
`size`	Tamaño del proceso (código + datos)	`pi_dvm`	Columna SIZE de `ps gvw`
`dsize`	Tamaño de los datos	`pi_dsize`	
`tsize`	Tamaño del texto (código)	`pi_tsize`	Columna TSIZ de `ps gvw`
`sdsize`	Tamaño de los datos de la biblioteca compartida	`pi_sdsize`	
`drss`	Tamaño del conjunto de datos residente	`pi_drss`	
`trss`	Tamaño del conjunto de texto residente	`pi_trss`	

Notas para AIX:

1. Al elegir los parámetros para la clave del item `proc.mem[]` en AIX, intente especificar criterios de selección de procesos precisos. De lo contrario, existe el riesgo de que se contabilicen procesos no deseados en el resultado de `proc.mem[]`.

Ejemplo:

```
$ zabbix_agentd -t proc.mem[,,,NonExistingProcess,rss]
proc.mem[,,,NonExistingProcess,rss]           [u|2879488]
```

Este ejemplo muestra que especificar únicamente el parámetro de la línea de comandos (expresión regular con la que se debe comparar) hace que el agent de Zabbix se contabilice a sí mismo, lo que probablemente no es lo que desea.

2. No utilice `ps -ef` para consultar los procesos, ya que solo muestra los procesos que no pertenecen al kernel. Utilice `ps -Af` para ver todos los procesos que podrá ver el agent de Zabbix.
3. Veamos el ejemplo de `topasrec` para comprender cómo `proc.mem[]` del agent de Zabbix selecciona los procesos.

```
$ ps -Af | grep topasrec
root 10747984        1   0   Mar 16      -  0:00 /usr/bin/topasrec  -L -s 300 -R 1 -r 6 -o /var/perf daily
```

proc.mem[] tiene los siguientes argumentos:

proc.mem[<name>,<user>,<mode>,<cmdline>,<memtype>]

donde: - El primer criterio es un nombre de proceso (argumento <name>). En nuestro ejemplo, Zabbix agent lo verá como topasrec. Para que coincida, debe especificar topasrec o dejarlo vacío. - El segundo criterio es un nombre de usuario (argumento <user>). Para que coincida, debe especificar root o dejarlo vacío. - El tercer criterio utilizado en la selección de procesos es un argumento <cmdline>. Zabbix agent verá su valor como /usr/bin/topasrec -L -s 300 -R 1 -r 6 -o /var/perf/daily/ -ypersistent=1 -O type=bin -ystart_time=04:08:54,Mar16,2023. Para que coincida, debe especificar una expresión regular que coincida con esta cadena o dejarla vacía.

Los argumentos <mode> y <memtype> se aplican después de utilizar los tres criterios mencionados anteriormente.

FreeBSD

Consulte en la tabla los valores compatibles con el parámetro memtype en FreeBSD.

Valor compatible	Descripción	Origen en la estructura kinfo_proc	Intenta ser compatible con
vsize	Tamaño de la memoria virtual	kp_eproc.e_vm.vm_map_size o ki_size	
pmem	Porcentaje de memoria real	calculado a partir de rss	ps -o pmem
rss	Tamaño del conjunto residente	kp_eproc.e_vm.vm_rssize o ki_rssize	
size [1]	Tamaño del proceso (código + datos + pila)	tsize + dsize + ssize	
tsize	Tamaño del texto (código)	kp_eproc.e_vm.vm_tsize o ki_tsize	
dsize	Tamaño de los datos	kp_eproc.e_vm.vm_dsize o ki_dsize	
ssize	Tamaño de la pila	kp_eproc.e_vm.vm_ssize o ki_ssize	

Linux

Consulta en la tabla los valores compatibles con el parámetro memtype en Linux.

Valor compatible	Descripción	Origen en el archivo /proc/<pid>/status
vsize [1]	Tamaño de la memoria virtual	VmSize
pmem	Porcentaje de memoria real	(VmRSS/total_memory) * 100
rss	Tamaño del conjunto residente	VmRSS
data	Tamaño del segmento de datos	VmData
exe	Tamaño del segmento de código	VmExe
hwm	Tamaño máximo del conjunto residente	VmHWM
lck	Tamaño de la memoria bloqueada	VmLck
lib	Tamaño de las bibliotecas compartidas	VmLib
peak	Tamaño máximo de la memoria virtual	VmPeak
pin	Tamaño de las páginas fijadas	VmPin
pte	Tamaño de las entradas de la tabla de páginas	VmPTE
size	Tamaño de los segmentos de código + datos + pila del proceso	VmExe + VmData + VmStk
stk	Tamaño del segmento de pila	VmStk
swap	Tamaño del espacio de intercambio utilizado	VmSwap

Notas para Linux:

1. No todos los valores de memtype son compatibles con los kernels de Linux antiguos. Por ejemplo, los kernels de Linux 2.4 no son compatibles con los valores hwm, pin, peak, pte y swap.

2. Hemos observado que la supervisión propia del proceso de comprobación activa del agent de Zabbix con `proc.mem[...,...,...,...` muestra un valor 4 kB mayor que el indicado en la línea `VmData` del archivo `/proc/<pid>/status` del agent. En el momento de la medición propia, el segmento de datos del agent aumenta en 4 kB y luego vuelve a su tamaño anterior.

Solaris

Consulte en la tabla los valores compatibles con el parámetro `memtype` en Solaris.

Valor compatible	Descripción	Origen en la estructura psinfo	Intenta ser compatible con
`vsize` [1]	Tamaño de la imagen del proceso	`pr_size`	`ps -o vsz`
`pmem`	Porcentaje de memoria real	`pr_pctmem`	`ps -o pmem`
`rss`	Tamaño del conjunto residente Puede estar subestimado (consulte la descripción de `rss` en `man ps`).	`pr_rssize`	`ps -o rss`

Notas al pie

[1] Valor por defecto.

Notas sobre la selección de procesos en los items proc.mem y proc.num

Procesos que modifican su línea de comandos

Algunos programas utilizan la modificación de su línea de comandos como método para mostrar su actividad actual. Un usuario puede ver la actividad ejecutando los comandos `ps` y `top`. Ejemplos de tales programas incluyen *PostgreSQL*, *Sendmail*, *Zabbix*.

Veamos un ejemplo en Linux. Supongamos que queremos monitorizar un número de procesos del agent de Zabbix.

El comando `ps` muestra los procesos de interés como

```
$ ps -fu zabbix
UID        PID  PPID  C STIME TTY          TIME CMD
...
zabbix    6318     1  0 12:01 ?        00:00:00 sbin/zabbix_agentd -c /home/zabbix/zabbix_agentd.conf
zabbix    6319  6318  0 12:01 ?        00:00:01 sbin/zabbix_agentd: collector [idle 1 sec]
zabbix    6320  6318  0 12:01 ?        00:00:00 sbin/zabbix_agentd: listener #1 [waiting for connection]
zabbix    6321  6318  0 12:01 ?        00:00:00 sbin/zabbix_agentd: listener #2 [waiting for connection]
zabbix    6322  6318  0 12:01 ?        00:00:00 sbin/zabbix_agentd: listener #3 [waiting for connection]
zabbix    6323  6318  0 12:01 ?        00:00:00 sbin/zabbix_agentd: active checks #1 [idle 1 sec]
...
```

Seleccionar procesos por nombre y usuario hace el trabajo:

```
$ zabbix_get -s localhost -k 'proc.num[zabbix_agentd,zabbix]'
6
```

Ahora renombremos el ejecutable `zabbix_agentd` a `zabbix_agentd_30` y reiniciémoslo.

`ps` ahora muestra

```
$ ps -fu zabbix
UID        PID  PPID  C STIME TTY          TIME CMD
...
zabbix    6715     1  0 12:53 ?        00:00:00 sbin/zabbix_agentd_30 -c /home/zabbix/zabbix_agentd.conf
zabbix    6716  6715  0 12:53 ?        00:00:00 sbin/zabbix_agentd_30: collector [idle 1 sec]
zabbix    6717  6715  0 12:53 ?        00:00:00 sbin/zabbix_agentd_30: listener #1 [waiting for connection
zabbix    6718  6715  0 12:53 ?        00:00:00 sbin/zabbix_agentd_30: listener #2 [waiting for connection
zabbix    6719  6715  0 12:53 ?        00:00:00 sbin/zabbix_agentd_30: listener #3 [waiting for connection
zabbix    6720  6715  0 12:53 ?        00:00:00 sbin/zabbix_agentd_30: active checks #1 [idle 1 sec]
...
```

Ahora, seleccionar procesos por nombre y usuario produce un resultado incorrecto:

```
$ zabbix_get -s localhost -k 'proc.num[zabbix_agentd_30,zabbix]'
1
```

¿Por qué un simple cambio de nombre del ejecutable a uno más largo conduce a un resultado tan diferente?

El agent de Zabbix comienza comprobando el nombre del proceso. Se abre el archivo /proc/<pid>/status y se comprueba la línea Name. En nuestro caso, las líneas Name son:

```
$ grep Name /proc/{6715,6716,6717,6718,6719,6720}/status
/proc/6715/status:Name:   zabbix_agentd_3
/proc/6716/status:Name:   zabbix_agentd_3
/proc/6717/status:Name:   zabbix_agentd_3
/proc/6718/status:Name:   zabbix_agentd_3
/proc/6719/status:Name:   zabbix_agentd_3
/proc/6720/status:Name:   zabbix_agentd_3
```

El nombre del proceso en el archivo status se trunca a 15 caracteres.

Un resultado similar puede verse con el comando ps:

```
$ ps -u zabbix
  PID TTY          TIME CMD
...
 6715 ?        00:00:00 zabbix_agentd_3
 6716 ?        00:00:01 zabbix_agentd_3
 6717 ?        00:00:00 zabbix_agentd_3
 6718 ?        00:00:00 zabbix_agentd_3
 6719 ?        00:00:00 zabbix_agentd_3
 6720 ?        00:00:00 zabbix_agentd_3
 ...
```

Obviamente, eso no es igual al valor de nuestro parámetro name de proc.num[] zabbix_agentd_30. Al no poder hacer coincidir el nombre del proceso del archivo status, el agent de Zabbix recurre al archivo /proc/<pid>/cmdline.

Cómo el agent ve el archivo "cmdline" puede ilustrarse ejecutando un comando

```
$ for i in 6715 6716 6717 6718 6719 6720; do cat /proc/$i/cmdline | awk '{gsub(/\x0/,"<NUL>"); print};'; d
sbin/zabbix_agentd_30<NUL>-c<NUL>/home/zabbix/zabbix_agentd.conf<NUL>
sbin/zabbix_agentd_30: collector [idle 1 sec]<NUL><NUL><NUL><NUL><NUL><NUL><NUL><NUL><NUL><NUL><NUL><NUL><
sbin/zabbix_agentd_30: listener #1 [waiting for connection]<NUL><NUL><NUL><NUL><NUL><NUL><NUL><NUL><NUL><N
sbin/zabbix_agentd_30: listener #2 [waiting for connection]<NUL><NUL><NUL><NUL><NUL><NUL><NUL><NUL><NUL><N
sbin/zabbix_agentd_30: listener #3 [waiting for connection]<NUL><NUL><NUL><NUL><NUL><NUL><NUL><NUL><NUL><N
sbin/zabbix_agentd_30: active checks #1 [idle 1 sec]<NUL><NUL><NUL><NUL><NUL><NUL><NUL><NUL><NUL><NUL><NUL
```

Los archivos /proc/<pid>/cmdline en nuestro caso contienen bytes nulos invisibles y no imprimibles, utilizados para terminar cadenas en el lenguaje *C*. Los bytes nulos se muestran como "<NUL>" en este ejemplo.

El agent de Zabbix comprueba "cmdline" para el proceso principal y toma un zabbix_agentd_30, que coincide con el valor de nuestro parámetro name zabbix_agentd_30. Por lo tanto, el proceso principal es contado por el item proc.num[zabbix_agentd_30,zabbix].

Al comprobar el siguiente proceso, el agent toma zabbix_agentd_30: collector [idle 1 sec] del archivo cmdline y no coincide con nuestro parámetro name zabbix_agentd_30. Por lo tanto, sólo el proceso principal que no modifica su línea de comandos es contado. Otros procesos del agent modifican su línea de comandos y son ignorados.

Este ejemplo muestra que el parámetro name no puede utilizarse en proc.mem[] y proc.num[] para seleccionar procesos en este caso.

Note:
Para el item proc.get[], cuando el agent de Zabbix comprueba "cmdline" para el nombre del proceso, sólo utilizará la parte del nombre que comienza desde la última barra y hasta el primer espacio o signo de dos puntos. El nombre del proceso recibido del archivo cmdline sólo se utilizará si su inicio coincide completamente con el nombre acortado del proceso en el archivo status. El algoritmo es el mismo tanto para el nombre del proceso en el filtro como en la salida JSON.

El uso del parámetro cmdline con una expresión regular adecuada produce un resultado correcto:

```
$ zabbix_get -s localhost -k 'proc.num[,zabbix,,zabbix_agentd_30[ :]]'
6
```

Tenga cuidado al utilizar los items proc.get[], proc.mem[] y proc.num[] para monitorizar programas que modifican sus líneas de comandos.

Antes de poner los parámetros name y cmdline en los items proc.get[], proc.mem[] y proc.num[], puede que desee probar los parámetros utilizando el item proc.num[] y el comando ps.

Hilos del kernel de Linux

Los hilos no se pueden seleccionar con el parámetro cmdline en las métricas proc.get[], proc.mem[] y proc.num[]

Tomemos como ejemplo uno de los hilos del kernel:

```
$ ps -ef| grep kthreadd
root         2     0  0 09:33 ?        00:00:00 [kthreadd]
```

Se puede seleccionar con el parámetro name del proceso:

```
$ zabbix_get -s localhost -k 'proc.num[kthreadd,root]'
1
```

Pero la selección por el parámetro cmdline del proceso no funciona:

```
$ zabbix_get -s localhost -k 'proc.num[,root,,kthreadd]'
0
```

La razón es que el agente Zabbix toma la expresión regular especificada en el parámetro cmdline y la aplica al contenido del archivo /proc/<pid>/cmdline del proceso. Para los hilos del kernel, sus archivos /proc/<pid>/cmdline están vacíos. Por lo tanto, el parámetro cmdline nunca coincide.

Conteo de hilos en las métricas proc.mem[] y proc.num[]

Los hilos del kernel de Linux son contados por la métrica proc.num[] pero no informan memoria en la métrica proc.mem[]. Por ejemplo:

```
$ ps -ef | grep kthreadd
root         2     0  0 09:51 ?        00:00:00 [kthreadd]

$ zabbix_get -s localhost -k 'proc.num[kthreadd]'
1

$ zabbix_get -s localhost -k 'proc.mem[kthreadd]'
ZBX_NOTSUPPORTED: Cannot get amount of "VmSize" memory.
```

Pero, ¿qué sucede si hay un proceso de usuario con el mismo nombre que un hilo del kernel? Entonces podría verse así:

```
$ ps -ef | grep kthreadd
root         2     0  0 09:51 ?        00:00:00 [kthreadd]
zabbix    9611  6133  0 17:58 pts/1    00:00:00 ./kthreadd

$ zabbix_get -s localhost -k 'proc.num[kthreadd]'
2

$ zabbix_get -s localhost -k 'proc.mem[kthreadd]'
4157440
```

proc.num[] contó tanto el hilo del kernel como el proceso de usuario. proc.mem[] informa la memoria solo para el proceso de usuario y cuenta la memoria del hilo del kernel como si fuera 0. Esto es diferente del caso anterior cuando se informó ZBX_NOTSUPPORTED.

Tenga cuidado al usar las métricas proc.mem[] y proc.num[] si el nombre del programa coincide con el de algún hilo.

Antes de poner parámetros en las métricas proc.mem[] y proc.num[], puede probar los parámetros usando la métrica proc.num[] y el comando ps.

Detalles de implementación de las comprobaciones net.tcp.service y net.udp.service

La implementación de las comprobaciones net.tcp.service y net.udp.service se detalla en esta página para varios servicios especificados en el parámetro service.

Parámetros de la métrica net.tcp.service

ftp

Crea una conexión TCP y espera que los primeros 4 caracteres de la respuesta sean ”220 ”, luego envía ”QUIT\r\n”. Se utiliza el puerto 21 por defecto si no se especifica.

http

Crea una conexión TCP sin esperar ni enviar nada. Se utiliza el puerto 80 por defecto si no se especifica.

https

Utiliza (y solo funciona con) libcurl, no verifica la autenticidad del certificado, no verifica el nombre del equipo en el certificado SSL, solo obtiene la cabecera de la respuesta (petición HEAD). Se utiliza el puerto 443 por defecto si no se especifica.

imap

Crea una conexión TCP y espera que los primeros 4 caracteres de la respuesta sean ”* OK”, luego envía ”a1 LOGOUT\r\n”. Se utiliza el puerto 143 por defecto si no se especifica.

ldap

Abre una conexión a un servidor LDAP y realiza una operación de búsqueda LDAP con el filtro establecido en (objectClass=*). Espera la recuperación exitosa del primer atributo de la primera entrada. Se utiliza el puerto 389 por defecto si no se especifica.

nntp

Crea una conexión TCP y espera que los primeros 3 caracteres de la respuesta sean ”200” o ”201”, luego envía ”QUIT\r\n”. Se utiliza el puerto 119 por defecto si no se especifica.

pop

Crea una conexión TCP y espera que los primeros 3 caracteres de la respuesta sean ”+OK”, luego envía ”QUIT\r\n”. Se utiliza el puerto 110 por defecto si no se especifica.

smtp

Crea una conexión TCP y espera que los primeros 3 caracteres de la respuesta sean ”220”, seguidos de un espacio, el final de línea o un guion. Las líneas que contienen un guion pertenecen a una respuesta multilínea y la respuesta se volverá a leer hasta que se reciba una línea sin el guion. Luego envía ”QUIT\r\n”. Se utiliza el puerto 25 por defecto si no se especifica.

ssh

Crea una conexión TCP. Si la conexión se ha establecido, ambas partes intercambian una cadena de identificación (SSH-major.minor-XXXX), donde major y minor son las versiones del protocolo y XXXX es una cadena. Zabbix comprueba si se encuentra la cadena que coincide con la especificación y luego envía de vuelta la cadena ”SSH-major.minor-zabbix_agent\r\n” o ”0\n” en caso de no coincidir. Se utiliza el puerto 22 por defecto si no se especifica.

tcp

Crea una conexión TCP sin esperar ni enviar nada. A diferencia de las otras comprobaciones, requiere que se especifique el parámetro de puerto.

telnet

Crea una conexión TCP y espera un aviso de inicio de sesión (’:’ al final). Se utiliza el puerto 23 por defecto si no se especifica.

Parámetros de la métrica net.udp.service

ntp

Envía un paquete SNTP por UDP y valida la respuesta de acuerdo con RFC 4330, sección 5. Se utiliza el puerto 123 por defecto si no se especifica.

Parámetros de proc.get

Descripción general

El item **proc.get**[<name>,<user>,<cmdline>,<mode>] es compatible con Linux, Windows, FreeBSD, OpenBSD y NetBSD.

La lista de parámetros de proceso devueltos por el item varía según el sistema operativo y el valor del argumento ’mode’.

Linux

Los siguientes parámetros del proceso se devuelven en Linux para cada modo:

mode=process	mode=thread	mode=summary
pid: PID	pid: PID	name: nombre del proceso
ppid: PID del proceso padre	ppid: PID del proceso padre	processes: número de procesos
name: nombre del proceso	name: nombre del proceso	vsize: tamaño de la memoria virtual

mode=process	mode=thread	mode=summary
cmdline: comando con argumentos	user: usuario (real) con el que se ejecuta el proceso	pmem: porcentaje de memoria real
user: usuario (real) con el que se ejecuta el proceso	group: grupo (real) con el que se ejecuta el proceso	rss: tamaño del conjunto residente
group: grupo (real) con el que se ejecuta el proceso	uid: ID de usuario	data: tamaño del segmento de datos
uid: ID de usuario	gid: ID del grupo con el que se ejecuta el proceso	exe: tamaño del segmento de código
gid: ID del grupo con el que se ejecuta el proceso	tid: ID del hilo	lib: tamaño de las bibliotecas compartidas
vsize: tamaño de la memoria virtual	tname: nombre del hilo	lck: tamaño de la memoria bloqueada
pmem: porcentaje de memoria real	cputime_user: tiempo total de CPU empleado en modo usuario (el valor se expresa en ciclos de reloj; divídalo entre `getconf CLK_TCK` o `sysconf(_SC_CLK_TCK)` para obtener los segundos)	pin: tamaño de las páginas fijadas
rss: tamaño del conjunto residente	cputime_system: tiempo total de CPU empleado en modo sistema (el valor se expresa en ciclos de reloj; divídalo entre `getconf CLK_TCK` o `sysconf(_SC_CLK_TCK)` para obtener los segundos)	pte: tamaño de las entradas de la tabla de páginas
data: tamaño del segmento de datos	state: estado del hilo	size: tamaño de los segmentos de código + datos + pila del proceso
exe: tamaño del segmento de código	ctx_switches: número de cambios de contexto	stk: tamaño del segmento de pila
hwm: tamaño máximo del conjunto residente	page_faults: número de fallos de página	swap: tamaño del espacio de intercambio utilizado
lck: tamaño de la memoria bloqueada		cputime_user: tiempo total de CPU empleado en modo usuario (el valor se expresa en ciclos de reloj; divídalo entre `getconf CLK_TCK` o `sysconf(_SC_CLK_TCK)` para obtener los segundos)
lib: tamaño de las bibliotecas compartidas		cputime_system: tiempo total de CPU empleado en modo sistema (el valor se expresa en ciclos de reloj; divídalo entre `getconf CLK_TCK` o `sysconf(_SC_CLK_TCK)` para obtener los segundos)
peak: tamaño máximo de la memoria virtual		ctx_switches: número de cambios de contexto
pin: tamaño de las páginas fijadas		threads: número de hilos
pte: tamaño de las entradas de la tabla de páginas		page_faults: número de fallos de página
size: tamaño de los segmentos de código + datos + pila del proceso		pss: memoria del tamaño proporcional del conjunto
stk: tamaño del segmento de pila		
swap: tamaño del espacio de intercambio utilizado		
cputime_user: tiempo total de CPU empleado en modo usuario (el valor se expresa en ciclos de reloj; divídalo entre `getconf CLK_TCK` o `sysconf(_SC_CLK_TCK)` para obtener los segundos)		

mode=process	mode=thread	mode=summary
cputime_system: tiempo total de CPU empleado en modo sistema (el valor se expresa en ciclos de reloj; divídalo entre `getconf CLK_TCK` o `sysconf(_SC_CLK_TCK)` para obtener los segundos) state: estado del proceso (obtenido de forma transparente desde procfs, formato largo) ctx_switches: número de cambios de contexto threads: número de hilos page_faults: número de fallos de página pss: memoria del tamaño proporcional del conjunto		

Sistemas operativos basados en BSD

Los siguientes parámetros de proceso se devuelven en FreeBSD, OpenBSD y NetBSD para cada modo:

mode=process	mode=thread	mode=summary
pid: PID	pid: PID	name: nombre del proceso
ppid: PID padre	ppid: PID padre	processes: número de procesos
jid: ID de jail (solo FreeBSD)	jid: ID de jail (solo FreeBSD)	vsize: tamaño de la memoria virtual
jname: nombre de jail (solo FreeBSD)	jname: nombre de jail (solo FreeBSD)	pmem: porcentaje de memoria real (solo FreeBSD)
name: nombre del proceso	name: nombre del proceso	rss: tamaño del conjunto residente
cmdline: comando con argumentos	user: usuario (real) bajo el que se ejecuta el proceso	size: tamaño del proceso (código + datos + pila)
user: usuario (real) bajo el que se ejecuta el proceso	group: grupo (real) bajo el que se ejecuta el proceso	tsize: tamaño del texto (código)
group: grupo (real) bajo el que se ejecuta el proceso	uid: ID de usuario	dsize: tamaño de los datos
uid: ID de usuario	gid: ID del grupo bajo el que se ejecuta el proceso	ssize: tamaño de la pila
gid: ID del grupo bajo el que se ejecuta el proceso	tid: ID del hilo	cputime_user: tiempo total de CPU en modo usuario (informado en segundos)
vsize: tamaño de la memoria virtual	tname: nombre del hilo	cputime_system: tiempo total de CPU en modo sistema (informado en segundos)
pmem: porcentaje de memoria real (solo FreeBSD)	cputime_user: tiempo total de CPU en modo usuario (informado en segundos)	ctx_switches: número de cambios de contexto
rss: tamaño del conjunto residente	cputime_system: tiempo total de CPU en modo sistema (informado en segundos)	threads: número de hilos (no soportado para NetBSD)
size: tamaño del proceso (código + datos + pila)	state: estado del hilo	stk: tamaño del segmento de pila
tsize: tamaño del texto (código)	ctx_switches: número de cambios de contexto	page_faults: número de fallos de página
dsize: tamaño de los datos	io_read_op: número de veces que el sistema tuvo que realizar una entrada	fds: número de descriptores de archivo (solo OpenBSD)
ssize: tamaño de la pila	io_write_op: número de veces que el sistema tuvo que realizar una salida	swap: tamaño del espacio de swap utilizado
cputime_user: tiempo total de CPU en modo usuario (informado en segundos)		io_read_op: número de veces que el sistema tuvo que realizar una entrada

mode=process	mode=thread	mode=summary
cputime_system: tiempo total de CPU en modo sistema (informado en segundos) state: estado del proceso (espera de disco/ejecutando/durmiendo/parada de rastreo/zombi/otro) ctx_switches: número de cambios de contexto threads: número de hilos (no soportado para NetBSD) page_faults: número de fallos de página fds: número de descriptores de archivo (solo OpenBSD) swap: tamaño del espacio de swap utilizado io_read_op: número de veces que el sistema tuvo que realizar una entrada io_write_op: número de veces que el sistema tuvo que realizar una salida		io_write_op: número de veces que el sistema tuvo que realizar una salida

Windows

Los siguientes parámetros de proceso se devuelven en Windows para cada modo:

mode=process	mode=thread	mode=summary
pid: PID ppid: PID padre name: nombre del proceso user: usuario bajo el que se ejecuta el proceso sid: SID de usuario vmsize: tamaño de la memoria virtual wkset: tamaño del conjunto de trabajo del proceso cputime_user: tiempo total de CPU en modo usuario (reportado en milisegundos) cputime_system: tiempo total de CPU en modo sistema (reportado en milisegundos) threads: número de hilos page_faults: número de fallos de página handles: número de manejadores io_read_b: bytes de IO leídos io_write_b: bytes de IO escritos io_read_op: operaciones de lectura de IO io_write_op: operaciones de escritura de IO	pid: PID ppid: PID padre name: nombre del proceso user: usuario bajo el que se ejecuta el proceso sid: SID de usuario tid: ID del hilo	name: nombre del proceso processes: número de procesos vmsize: tamaño de la memoria virtual wkset: tamaño del conjunto de trabajo del proceso cputime_user: tiempo total de CPU en modo usuario (reportado en milisegundos) cputime_system: tiempo total de CPU en modo sistema (reportado en milisegundos) threads: número de hilos page_faults: número de fallos de página handles: número de manejadores io_read_b: bytes de IO leídos io_write_b: bytes de IO escritos io_read_op: operaciones de lectura de IO io_write_op: operaciones de escritura de IO io_other_b: bytes de IO transferidos, distintos de operaciones de lectura y escritura io_other_op: operaciones de IO, distintas de operaciones de lectura y escritura

mode=process	mode=thread	mode=summary
io_other_b: bytes de IO transferidos, distintos de operaciones de lectura y escritura io_other_op: operaciones de IO, distintas de operaciones de lectura y escritura		

Configuración de la interfaz del host inalcanzable/no disponible

Resumen

Varios parámetros de configuración definen cómo debe comportarse el server de Zabbix cuando falla una comprobación de agent (Zabbix, SNMP, IPMI, JMX) y una interfaz de host se vuelve inaccesible.

Interfaz inaccesible

Una interfaz de host se considera inaccesible después de una comprobación fallida (error de red, tiempo de espera agotado) por parte de Zabbix, SNMP, IPMI o los agentes JMX. Desde Zabbix 6.2.0, las comprobaciones activas de Zabbix agent también afectan a la disponibilidad de la interfaz. Cuando las comprobaciones activas dejan de estar disponibles, contribuyen al estado general de disponibilidad de la interfaz del agent.

A partir del momento en que una interfaz se vuelve inaccesible, **UnreachableDelay** define con qué frecuencia se vuelve a comprobar usando uno de los items (incluidas las reglas LLD). Estas nuevas comprobaciones las realizan los pollers de inaccesibilidad (o los pollers IPMI para comprobaciones IPMI). De forma predeterminada, el intervalo entre comprobaciones consecutivas de accesibilidad es de 15 segundos.

Attention:
Las comprobaciones realizadas por los pollers asíncronos no se trasladan a los pollers de inaccesibilidad.

En el registro del server de Zabbix, la inaccesibilidad se indica con mensajes como:

```
Zabbix agent item "system.cpu.load[percpu,avg1]" on host "New host" failed: first network error, wait for
Zabbix agent item "system.cpu.load[percpu,avg15]" on host "New host" failed: another network error, wait f
```

Los mensajes del registro especifican el item exacto que falló y su tipo (Zabbix agent).

Note:
El parámetro *Timeout* también afectará a cuán pronto se vuelve a comprobar una interfaz durante el estado de inaccesibilidad. Si Timeout está establecido en 20 segundos y UnreachableDelay en 30 segundos, la siguiente comprobación se realizará 50 segundos después del primer intento.

El parámetro **UnreachablePeriod** define la duración total del período de inaccesibilidad. De forma predeterminada, UnreachablePeriod es de 45 segundos. Este valor debe ser varias veces mayor que UnreachableDelay para garantizar que una interfaz se vuelva a comprobar varias veces antes de marcarla como no disponible.

Un item interno, zabbix[host,active_agent,available], permite supervisar la disponibilidad de las comprobaciones activas en escenarios de inaccesibilidad.

Cambiando la interfaz de nuevo a disponible

Cuando finaliza el período de inaccesibilidad, la interfaz se sondea nuevamente, disminuyendo la prioridad para la métrica que puso la interfaz en estado de inaccesibilidad. Si la interfaz inaccesible reaparece, la monitorización vuelve a la normalidad automáticamente:

```
reanudando comprobaciones del agente Zabbix en el equipo "Nuevo equipo": conexión restaurada
```

Note:
Una vez que la interfaz se vuelve disponible, el equipo no sondea todas sus métricas inmediatamente por dos razones:
- Podría sobrecargar el equipo.
- El tiempo de restauración de la interfaz no siempre coincide con el tiempo programado de sondeo de la métrica.

Por lo tanto, después de que la interfaz se vuelve disponible, las métricas no se sondean inmediatamente, sino que se reprograman para su próxima ronda de sondeo.

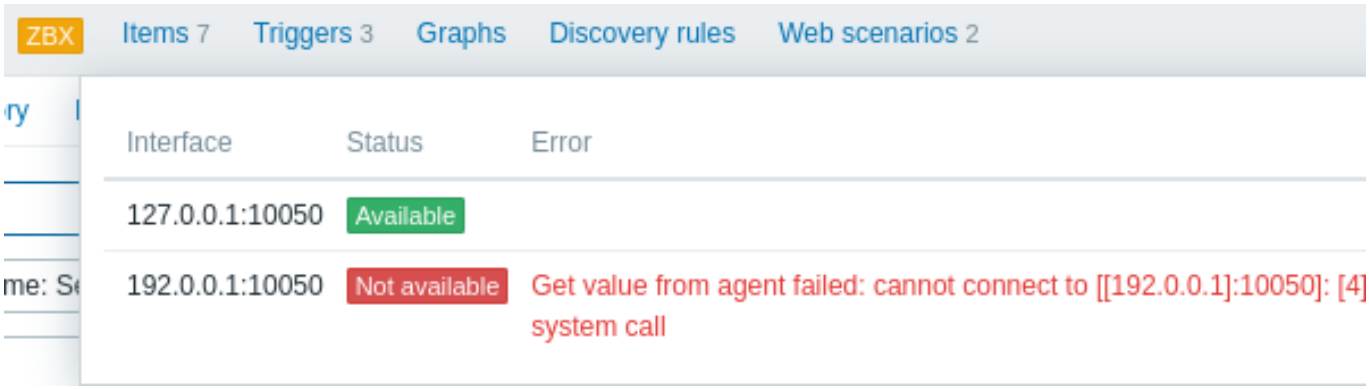
Interfaz no disponible

Después de que finaliza el UnreachablePeriod y la interfaz no ha reaparecido, la interfaz se trata como no disponible.

En el registro del servidor se indica mediante mensajes como estos:

```
deshabilitando temporalmente las comprobaciones del agente Zabbix en el equipo "Nuevo equipo": interfaz no
```

y en el frontend el icono de disponibilidad del equipo pasa de verde/gris a amarillo/rojo (los detalles de la interfaz no disponible pueden verse en el cuadro de información que se muestra cuando se posiciona el ratón sobre el icono de disponibilidad del equipo):



El parámetro **UnavailableDelay** define con qué frecuencia se comprueba una interfaz durante la indisponibilidad de la interfaz.

Por defecto es de 60 segundos (por lo que en este caso "deshabilitando temporalmente", según el mensaje del registro anterior, significará deshabilitar las comprobaciones durante un minuto).

Cuando se restablece la conexión con la interfaz, la monitorización vuelve a la normalidad automáticamente también:

```
habilitando las comprobaciones del agente Zabbix en el equipo "Nuevo equipo": la interfaz volvió a estar d
```

Monitoreo remoto de estadísticas de Zabbix

Resumen

Es posible hacer que algunas métricas internas de Zabbix server y proxy sean accesibles de forma remota por otra instancia de Zabbix o por una herramienta de terceros. Esto puede ser útil para que los proveedores de soporte/servicios puedan supervisar de forma remota los Zabbix server/proxies de sus clientes, o para que, en organizaciones donde Zabbix no es la herramienta principal de supervisión, las métricas internas de Zabbix puedan ser supervisadas por un sistema de terceros en una configuración de supervisión jerárquica.

Las estadísticas internas de Zabbix se exponen a un conjunto configurable de direcciones incluidas en el nuevo parámetro 'StatsAllowedIP' server/proxy. Las solicitudes solo se aceptarán desde estas direcciones.

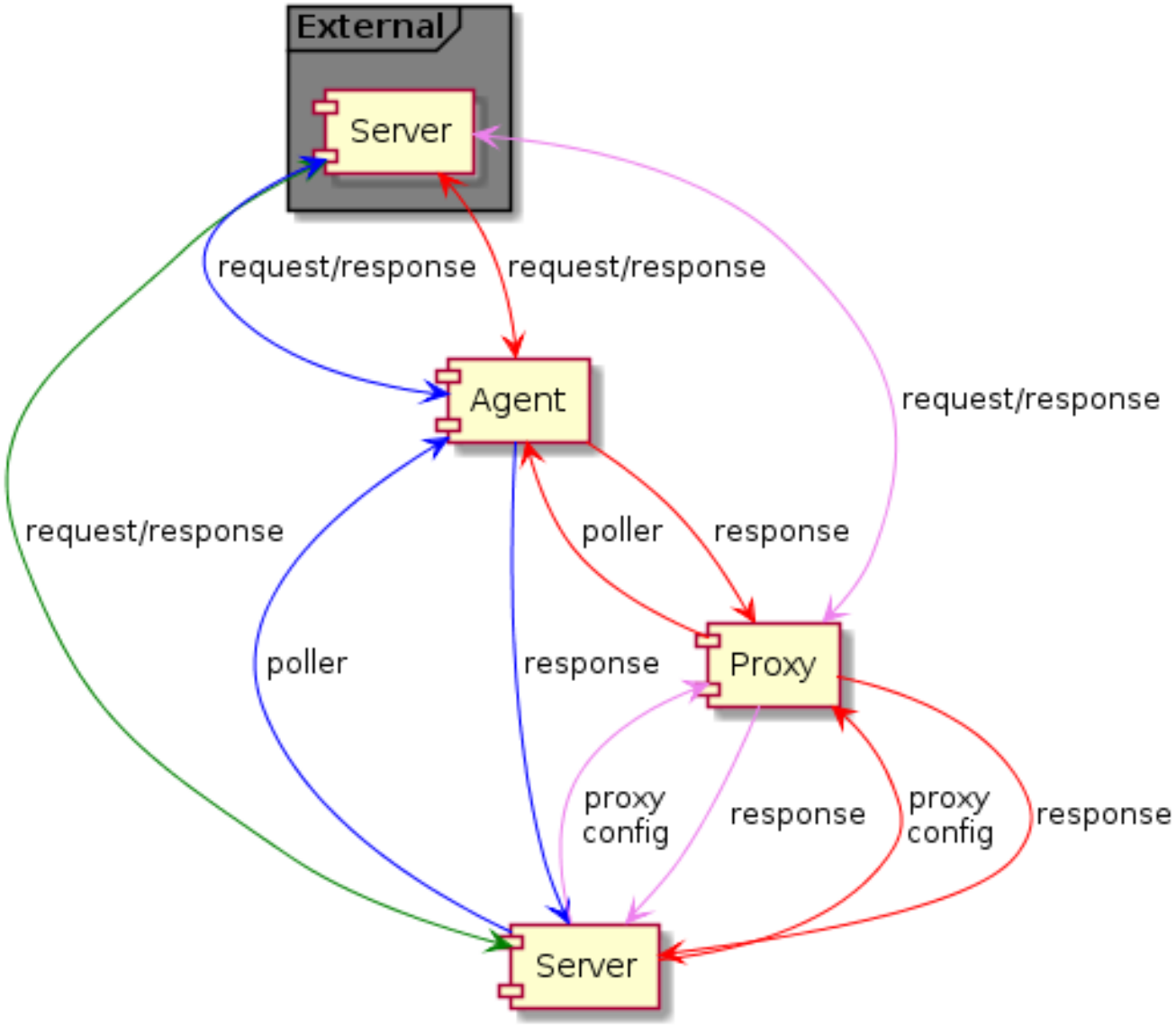
Métricas

Para configurar la consulta de estadísticas internas en otra instancia de Zabbix, puede utilizar dos métricas:

- Métrica interna `zabbix[stats,<ip>,<port>]` - para consultas remotas directas al servidor/proxy de Zabbix. <ip> y <port> se utilizan para identificar la instancia de destino.
- Métrica de agente `zabbix.stats[<ip>,<port>]` - para consultas remotas basadas en agente al servidor/proxy de Zabbix. <ip> y <port> se utilizan para identificar la instancia de destino.

Consulte también: Métricas internas, Métricas de agente Zabbix

El siguiente diagrama ilustra el uso de cada métrica dependiendo del contexto.



- - Servidor → instancia externa de Zabbix (`zabbix[stats,<ip>,<port>]`)
- - Servidor → proxy → instancia externa de Zabbix (`zabbix[stats,<ip>,<port>]`)
- - Servidor → agente → instancia externa de Zabbix (`zabbix.stats[<ip>,<port>]`)
- - Servidor → proxy → agente → instancia externa de Zabbix (`zabbix.stats[<ip>,<port>]`)

Para asegurarse de que la instancia de destino permita la consulta por parte de la instancia externa, incluya la dirección de la instancia externa en el parámetro 'StatsAllowedIP' en la instancia de destino.

Métricas expuestas

Las métricas de estadísticas recopilan las estadísticas en bloque y devuelven un JSON, que es la base para que las métricas dependientes obtengan sus datos. Las siguientes métricas internas son devueltas por cualquiera de las dos métricas:

- `zabbix[boottime]`
- `zabbix[hosts]`
- `zabbix[items]`
- `zabbix[items_unsupported]`
- `zabbix[preprocessing]` (solo servidor)
- `zabbix[preprocessing_queue]` (solo servidor)
- `zabbix[process,<type>,<mode>,<state>]` (solo estadísticas basadas en el tipo de proceso)
- `zabbix[rcache,<cache>,<mode>]`
- `zabbix[requiredperformance]`
- `zabbix[triggers]` (solo servidor)
- `zabbix[uptime]`
- `zabbix[vcache,buffer,<mode>]` (solo servidor)

- zabbix[vcache,cache,<parameter>]
- zabbix[version]
- zabbix[vmware,buffer,<mode>]
- zabbix[wcache,<cache>,<mode>] (tipo de caché 'trends' solo en servidor)

Plantillas

Las plantillas están disponibles para la monitorización remota de las métricas internas del servidor Zabbix o del proxy desde una instancia externa:

- Estado remoto del servidor Zabbix
- Estado remoto del proxy Zabbix

Tenga en cuenta que, para utilizar una plantilla para la monitorización remota de varias instancias externas, se requiere un equipo separado para la monitorización de cada instancia externa.

Proceso trapper

La recepción de solicitudes de métricas internas desde una instancia externa de Zabbix es gestionada por el proceso trapper, que valida la solicitud, recopila las métricas, crea el búfer de datos JSON y envía el JSON preparado de vuelta, por ejemplo, desde el servidor:

```
{
  "response": "success",
  "data": {
    "boottime": N,
    "uptime": N,
    "hosts": N,
    "items": N,
    "items_unsupported": N,
    "preprocessing": {
        "queued": {
          "count": N,
          "size": N
        },
        "direct": {
          "count": N,
          "size": N
        },
        "queue": N
    },
    "preprocessing_queue": N,
    "process": {
      "alert manager": {
        "busy": {
          "avg": N,
          "max": N,
          "min": N
        },
        "idle": {
          "avg": N,
          "max": N,
          "min": N
        },
        "count": N
      },
...
    },
    "queue": N,
    "rcache": {
      "total": N,
      "free": N,
      "pfree": N,
      "used": N,
      "pused": N
    },
```

```
    "requiredperformance": N,
    "triggers": N,
    "uptime": N,
    "vcache": {
      "buffer": {
        "total": N,
        "free": N,
        "pfree": N,
        "used": N,
        "pused": N
      },
      "cache": {
        "requests": N,
        "hits": N,
        "misses": N,
        "mode": N
      }
    },
    "vmware": {
      "total": N,
      "free": N,
      "pfree": N,
      "used": N,
      "pused": N
    },
    "version": "N",
    "wcache": {
      "values": {
        "all": N,
        "float": N,
        "uint": N,
        "str": N,
        "log": N,
        "text": N,
        "not supported": N
      },
      "history": {
        "pfree": N,
        "free": N,
        "total": N,
        "used": N,
        "pused": N
      },
      "index": {
        "pfree": N,
        "free": N,
        "total": N,
        "used": N,
        "pused": N
      },
      "trend": {
        "pfree": N,
        "free": N,
        "total": N,
        "used": N,
        "pused": N
      }
    }
  }
}
```

Items de cola internos

También hay otros dos items que permiten específicamente consultar de forma remota las estadísticas de la cola interna en otra instancia de Zabbix:

- item interno `zabbix[stats,<ip>,<port>,queue,<from>,<to>]` - para consultas directas de la cola interna al server/proxy remoto de Zabbix
- item de agent `zabbix.stats[<ip>,<port>,queue,<from>,<to>]` - para consultas de la cola interna basadas en agent al server/proxy remoto de Zabbix

Consulte también: Items internos, Items de Zabbix agent

Configuración de Kerberos con Zabbix

Descripción general

La autenticación Kerberos puede utilizarse en la monitorización web y en los items HTTP en Zabbix.

Esta página describe un ejemplo de configuración de Kerberos para que el servidor Zabbix realice la monitorización web de `www.example.com` con un principal Kerberos para el proceso Zabbix en Debian/Ubuntu.

Configuración

1. Instale KDC y las utilidades del cliente:

```
sudo apt update
sudo apt install krb5-kdc krb5-admin-server krb5-user
```

Durante la configuración del paquete, responda a las indicaciones, por ejemplo:

```
Default Kerberos version 5 realm: EXAMPLE.COM
Kerberos servers for your realm: localhost (or your FQDN)
Administrative server for your Kerberos realm: localhost (or your FQDN)
```

2. Asigne un nombre de host descriptivo (opcional, para pruebas locales).

Edite /etc/hosts y añada una entrada para su DC y webserver si no dispone de DNS:

```
sudo vi /etc/hosts
```

Ejemplo de línea que puede añadir:

```
192.168.1.100  dc01.example.com dc01
```

3. Configure el cliente Kerberos y el realm de KDC:

```
sudo vi /etc/krb5.conf
```

Ejemplo de configuración:

```
[libdefaults]
    default_realm = EXAMPLE.COM
    dns_lookup_realm = false
    dns_lookup_kdc = false
    rdns = false
    ticket_lifetime = 24h
    renew_lifetime = 7d
    forwardable = true

[realms]
    EXAMPLE.COM = {
        kdc = dc01.example.com
        admin_server = dc01.example.com
    }

[domain_realm]
    .example.com = EXAMPLE.COM
    example.com = EXAMPLE.COM
```

Si planea utilizar `.localdomain` u otros nombres no públicos, añada asignaciones explícitas de dominio→realm para que funcione la asignación de nombre de host→realm.
Las discrepancias aquí provocan errores `Server not found in Kerberos database`.

4. Inicialice la base de datos de Kerberos (una sola vez, en el host de KDC).
Establezca una contraseña maestra segura cuando se le solicite:

```
sudo krb5_newrealm
```

5. Cree el principal `HTTP/host.fqdn@REALM` utilizando el nombre de host exacto que usarán los clientes; se recomienda utilizar minúsculas (por ejemplo, `HTTP/web.example.com@EXAMPLE.COM`).
Una discrepancia en las mayúsculas/minúsculas o en el nombre provoca el error `Server not found in Kerberos database`.

```
sudo kadmin.local
```

Dentro de `kadmin.local`:

```
addprinc kerb-admin@EXAMPLE.COM     # principal administrativo
addprinc -randkey HTTP/dc01.example.com@EXAMPLE.COM
ktadd -k /etc/apache2/http.keytab HTTP/dc01.example.com@EXAMPLE.COM
quit
```

Mueva el keytab al web host (o manténgalo local si se trata de la misma máquina) y establezca permisos que Apache pueda utilizar:

```
chown www-data:www-data /etc/apache2/http.keytab
chmod 600 /etc/apache2/http.keytab
#### verificar
sudo -u www-data -k /etc/apache2/http.keytab
```

6. Instale y habilite el módulo GSSAPI de Apache:

```
sudo apt install libapache2-mod-auth-gssapi
sudo a2enmod auth_gssapi
sudo a2enmod headers
sudo systemctl restart apache2
```

> **Attention:**
> No todas las versiones de mod_auth_gssapi admiten todas las directivas `Gssapi*`. Si Apache falla con `Invalid command 'GssapiCredStore'`, elimine la directiva no compatible o actualice el módulo.

7. Configure un VirtualHost (ajuste `DocumentRoot` / la ruta a su interfaz de Zabbix):

```
sudo vi /etc/apache2/sites-available/zabbix.conf
```

Dentro de `zabbix.conf`:

```
<VirtualHost *:80>
    ServerName dc01.example.com
    DocumentRoot /usr/share/zabbix/ui
    <Directory /usr/share/zabbix/ui>
        Options FollowSymLinks
        AllowOverride None
        Require all granted
        AuthType GSSAPI
        AuthName "Kerberos Login"
        GssapiCredStore keytab:/etc/apache2/http.keytab
        GssapiLocalName On
        Require valid-user
    </Directory>
    RequestHeader set X-Remote-User %{REMOTE_USER}s env=REMOTE_USER
    RequestHeader unset Authorization
</VirtualHost>
```

Reinicie Apache:

```
sudo systemctl restart apache2
```

8. Habilite/inicie los servicios de KDC y verifique los puertos en escucha (host de KDC):

```
sudo systemctl enable --now krb5-kdc krb5-admin-server
ss -tnlp | grep :80    # o: sudo netstat -tnlp | grep :80
```

9. Obtenga un TGT para realizar pruebas (ejecútelo como el usuario que utilizará el ticket).

Debería aparecer `krbtgt/EXAMPLE.COM@EXAMPLE.COM` en la lista de tickets.
Ejecute `kinit` como el mismo usuario del sistema operativo que necesita el ticket (por ejemplo, `zabbix` para las comprobaciones web o `www-data`/Apache para las pruebas interactivas de SSO en el navegador).
Los tickets emitidos para otro usuario del sistema operativo no serán visibles a menos que se ajusten `KRB5CCNAME` y los permisos.

```
kinit kerb-admin@EXAMPLE.COM
klist
```

10. Pruebe el intercambio SPNEGO con curl (desde un cliente con un TGT válido).
Un `200 OK` (o una redirección a la aplicación) indica que SPNEGO se realizó correctamente:

```
curl -v --negotiate -u : http://dc01.example.com/
```

11. Opcionalmente, si la interfaz de Zabbix debe aceptar inicios de sesión autenticados mediante HTTP, habilite la autenticación HTTP en el frontend de Zabbix.

En el archivo de configuración del frontend (`zabbix.conf.php`), establezca:

```
$ZBX_FEATURE_FLAGS['http_auth_enabled'] = true;
```

A continuación, configure la autenticación HTTP en la interfaz web.

12. Configuración del navegador (Firefox se utiliza como ejemplo): establezca `network.negotiate-auth.trusted-uris` en los hosts que realizan Negotiate (`dc01.example.com`) para que el navegador envíe automáticamente los tokens de Kerberos.

Dentro de `about:config`:

```
network.negotiate-auth.trusted-uris = dc01.example.com
```

Ahora, al visitar `http://dc01.example.com`, debería iniciar sesión directamente en Zabbix sin mostrar el formulario.

13. Mantenga actualizadas las claves y los tickets.
La duración predeterminada de un ticket de Kerberos es de aproximadamente 10 horas.
Añada un temporizador de cron/systemd para evitar que caduquen:

```
####para el servicio web
kinit -kt /etc/apache2/http.keytab HTTP/dc01.example.com@EXAMPLE.COM
####para el usuario de monitorización
kinit -kt /var/lib/zabbix/kerb.keytab kerb-admin@EXAMPLE.COM
```

14. Comprobaciones de mantenimiento:

- `klist -k /etc/apache2/http.keytab` - verifique que el principal del servicio esté presente en el keytab.
- `sudo tail -f /var/log/apache2/error.log` - supervise los errores de GSSAPI (`gss_acquire_cred[_from]() failed to get server creds` significa que hay un problema con el keytab o los permisos, o que falta el principal).
- Si `curl --negotiate` devuelve 401/403, normalmente significa que el principal es incorrecto, que no hay ningún ticket, que el encabezado de host no coincide o que existe un problema con los permisos del sistema de archivos; compruebe los registros y las asignaciones de dominio de `/etc/krb5.conf`.

Notas sobre seguridad y permisos de archivos

Los archivos keytab solo deben ser legibles por la cuenta que los necesita. Ejemplos de permisos: 0400, con propietario `zabbix:zabbix`, para un keytab del usuario zabbix; o 0440, con propietario `root:www-data`, para un keytab de Apache.

Evite almacenar contraseñas de texto plano de larga duración en el host. Utilice keytabs o principales de máquina unidos al dominio siempre que sea posible.

Al ejecutar pruebas o scripts que establezcan `KRB5CCNAME` o copien keytabs, vuelva a comprobar el propietario y los permisos después de la operación: que un servidor web rechace las credenciales suele deberse a un problema de permisos de archivo.

parámetros de modbus.get

Descripción general

La siguiente tabla presenta los detalles de los parámetros de la métrica `modbus.get`.

Parámetros

Parámetro	Descripción	Por defecto	Ejemplo
endpoint	Protocolo y dirección del endpoint, definido como `protocol://connection_string` Valores posibles para el protocolo: *rtu*, *ascii* (solo Agente 2), *tcp* Formato de la cadena de conexión: con *tcp* - `dirección:puerto` con línea serie: *rtu*, *ascii* - `nombre_puerto:velocidad:parámetros` donde 'velocidad' - 1200, 9600, etc. 'parámetros' - bits de datos (5,6,7 u 8), paridad (n,e u o para ninguna/par/e impar), bits de parada (1 o 2)	protocolo: ninguno protocolo *rtu/ascii*: nombre_puerto: ninguno velocidad: 115200 parámetros: 8n1 protocolo *tcp*: dirección: ninguno puerto: 502	tcp://192.168.6.1:511 tcp://192.168.6.2 tcp://[::1]:511 tcp://::1 tcp://localhost:511 tcp://localhost rtu://COM1:9600:8n ascii://COM2:1200:7o2 rtu://ttyS0:9600 ascii://ttyS1
slave id	Dirección Modbus del dispositivo al que está destinado (1 a 247), ver MODBUS Messaging Implementation Guide (página 23) el dispositivo tcp (no GW) ignorará el campo	serie: 1 tcp: 255 (0xFF)	2
function	Vacío o valor de una función soportada: 1 - Leer bobina, 2 - Leer entrada discreta, 3 - Leer registros de retención, 4 - Leer registros de entrada	vacío	3
address	Dirección del primer registro, bobina o entrada. Si 'function' está vacío, entonces 'address' debe estar en el rango para: Bobina - 00001 - 09999 Entrada discreta - 10001 - 19999 Registro de entrada - 30001 - 39999 Registro de retención - 40001 - 49999 Si 'function' no está vacío, el campo 'address' será de 0 a 65535 y se usará sin modificación (PDU)	función vacía: 00001 función no vacía: 0	9999
count	Cantidad de 'type' secuenciales que se leerán del dispositivo, donde: para Bobina o Entrada discreta el 'type' = 1 bit para otros casos: (count*sizeof(type))/2 = cantidad real de registros a leer Si 'offset' no es 0, el valor se sumará a la 'cantidad real' El rango aceptable para la 'cantidad real' es 1:65535	1	2
type	Tipo de dato: para Leer bobina y Leer entrada discreta - *bit* para Leer registros de retención y Leer registros de entrada: *int8* - 8bit *uint8* - 8bit (sin signo) *int16* - 16bit *uint16* - 16bit (sin signo) *int32* - 32bit *uint32* - 32bit (sin signo) *float* - 32bit *uint64* - 64bit (sin signo) *double* - 64bit	bit uint16	uint64

Parámetro	Descripción	Por defecto	Ejemplo
endianness	Tipo de endianness: *be* - Big Endian *le* - Little Endian *mbe* - Mid-Big Endian *mle* - Mid-Little Endian Limitaciones: para 1 bit - be para 8 bits - be,le para 16 bits - be,le	be	le
offset	Número de registros, comenzando desde 'address', cuyo resultado será descartado. El tamaño de cada registro es de 16bit (necesario para soportar equipos que no soportan acceso de lectura aleatoria).	0	4

Creación de nombres personalizados de contadores de rendimiento para VMware

Descripción general

La ruta del contador de rendimiento de VMware tiene el formato `group/counter[rollup]`, donde:

- `group` - el grupo del contador de rendimiento, por ejemplo, *cpu*
- `counter` - el nombre del contador de rendimiento, por ejemplo, *usagemhz*
- `rollup` - el tipo de acumulación del contador de rendimiento, por ejemplo, *average*

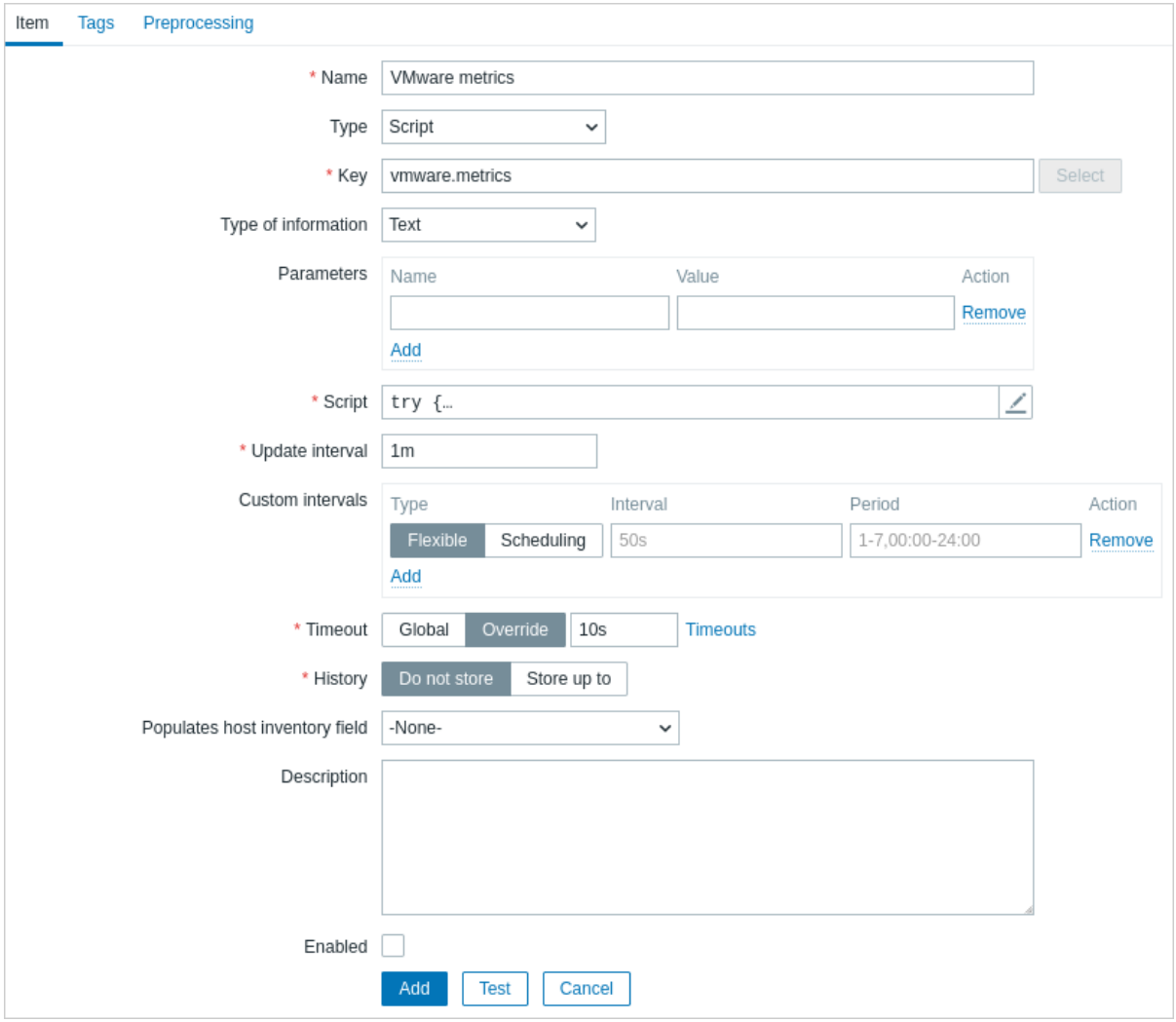
Por lo tanto, el ejemplo anterior daría la siguiente ruta del contador: `cpu/usagemhz[average]`

Las descripciones de los grupos de contadores de rendimiento, los nombres de los contadores y los tipos de acumulación se pueden encontrar en la documentación de VMware.

Es posible obtener nombres internos y crear nombres personalizados de contadores de rendimiento mediante el uso de un item de script en Zabbix.

Configuración

1. Cree un item de tipo Script deshabilitado en el host principal de VMware (donde está presente el item **eventlog[]**) con los siguientes parámetros:



- *Name*: VMware metrics
- *Type*: Script
- *Key*: vmware.metrics
- *Type of information*: Text
- *Script*: copie y pegue el script proporcionado a continuación
- *Timeout*: 10
- *History*: Do not store
- *Enabled*: unmarked

Script

```
try {
    Zabbix.log(4, 'vmware metrics script');

    var result, resp,
    req = new HttpRequest();
    req.addHeader('Content-Type: application/xml');
    req.addHeader('SOAPAction: "urn:vim25/6.0"');

    login = '<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:urn="urn:vi
    <soapenv:Header/>\
    <soapenv:Body>\
        <urn:Login>\
            <urn:_this type="SessionManager">SessionManager</urn:_this>\
            <urn:userName>{$VMWARE.USERNAME}</urn:userName>\
            <urn:password>{$VMWARE.PASSWORD}</urn:password>\
        </urn:Login>\
    </soapenv:Body>\
</soapenv:Envelope>'
    resp = req.post("{$VMWARE.URL}", login);
    if (req.getStatus() != 200) {
```

```
        throw 'Response code: '+req.getStatus();
    }

    query = '<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:urn="urn:vi
<soapenv:Header/>\
    <soapenv:Body>\
        <urn:RetrieveProperties>\
            <urn:_this type="PropertyCollector">propertyCollector</urn:_this>\
            <urn:specSet>\
                <urn:propSet>\
                   <urn:type>PerformanceManager</urn:type>\
                   <urn:pathSet>perfCounter</urn:pathSet>\
                </urn:propSet>\
                <urn:objectSet>\
                   <urn:obj type="PerformanceManager">PerfMgr</urn:obj>\
                </urn:objectSet>\
            </urn:specSet>\
        </urn:RetrieveProperties>\
    </soapenv:Body>\
</soapenv:Envelope>'
    resp = req.post("{$VMWARE.URL}", query);
    if (req.getStatus() != 200) {
        throw 'Response code: '+req.getStatus();
    }
    Zabbix.log(4, 'vmware metrics=' + resp);
    result = resp;

    logout = '<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:urn="urn:v
    <soapenv:Header/>\
    <soapenv:Body>\
        <urn:Logout>\
            <urn:_this type="SessionManager">SessionManager</urn:_this>\
        </urn:Logout>\
    </soapenv:Body>\
</soapenv:Envelope>'

    resp = req.post("{$VMWARE.URL}",logout);
    if (req.getStatus() != 200) {
        throw 'Response code: '+req.getStatus();
    }

} catch (error) {
    Zabbix.log(4, 'vmware call failed : '+error);
    result = {};
}

return result;
```

Una vez configurado el item, haga clic en el botón *Test* y luego en *Get value*.



Copie el XML recibido en cualquier formateador XML y busque la métrica deseada.

Un ejemplo de XML para una métrica:

```
<PerfCounterInfo xsi:type="PerfCounterInfo">
    <key>6</key>
```

```
    <nameInfo>
        <label>Usage in MHz</label>
        <summary>CPU usage in megahertz during the interval</summary>
        <key>usagemhz</key>
    </nameInfo>
    <groupInfo>
        <label>CPU</label>
        <summary>CPU</summary>
        <key>cpu</key>
    </groupInfo>
    <unitInfo>
        <label>MHz</label>
        <summary>Megahertz</summary>
        <key>megaHertz</key>
    </unitInfo>
    <rollupType>average</rollupType>
    <statsType>rate</statsType>
    <level>1</level>
    <perDeviceLevel>3</perDeviceLevel>
</PerfCounterInfo>
```

Use XPath para extraer la ruta del contador del XML recibido. Para el ejemplo anterior, el XPath será:

field	xPath	value
group	//groupInfo[../key=6]/key	cpu
counter	//nameInfo[../key=6]/key	usagemhz
rollup	//rollupType[../key=6]	average

La ruta resultante del contador de rendimiento en este caso es: `cpu/usagemhz[average]`

Valores de retorno para system.sw.packages.get

Descripción general

Esta sección proporciona detalles sobre los valores de retorno para la métrica `system.sw.packages.get` del agente Zabbix.

Detalles

La salida de este item es un array de objetos, cada uno de los cuales contiene las siguientes propiedades:

- **name** - nombre del paquete
- **manager** - gestor de paquetes que informó estos datos (`rpm`, `dpkg`, `pacman`, `pkgtool` o `portage`)
- **version** - versión del paquete
- **size** - tamaño del paquete sin comprimir en bytes (si no está disponible, se establece en 0)
- **arch** - arquitectura del paquete
- **buildtime** - un objeto con 2 entradas:
  - **timestamp** - marca de tiempo UNIX de cuando se compiló el paquete (si no está disponible, se establece en 0)
  - **value** - fecha y hora legibles para humanos de cuando se compiló el paquete (si no está disponible, se establece en una cadena vacía)
- **installtime** - un objeto con 2 entradas:
  - **timestamp** - marca de tiempo UNIX de cuando se instaló el paquete (si no está disponible, se establece en 0)
  - **value** - fecha y hora legibles para humanos de cuando se instaló el paquete (si no está disponible, se establece en una cadena vacía)

Ejemplo:

```
[
    {
        "name": "util-linux-core",
        "manager": "rpm",
        "version": "2.37.4-3.el9",
        "size": 1296335,
        "arch": "x86_64",
        "buildtime": {
```

```
            "timestamp" : 1653552239,
            "value" : "Sep 20 01:39:40 2021 UTC"
        },
        "installtime": {
            "timestamp" : 1660780885,
            "value" : "Aug 18 00:01:25 2022 UTC"
        }
    },
    {
        "name": "xfonts-base",
        "manager": "dpkg",
        "version": "1:1.0.5",
        "size": 7337984,
        "arch": "all",
        "buildtime": {
            "timestamp": 0,
            "value": ""
        },
        "installtime": {
            "timestamp": 0,
            "value": ""
        }
    }
]
```

Valores de retorno para net.dns.get

Descripción general

Esta sección proporciona detalles sobre los valores de retorno para la métrica `net.dns.get` del agente Zabbix 2.

Detalles

La salida de este item es un objeto que contiene información de registros DNS basada en los parámetros proporcionados en la clave del item.

Por ejemplo, el item `net.dns.get[,example.com]` puede devolver el siguiente JSON de una consulta rechazada:

```
{
    "flags": [
        "RA"
    ],
    "query_time": "0.019030",
    "question_section": [
        {
            "qclass": "IN",
            "qname": "example.com.",
            "qtype": "SOA"
        }
    ],
    "response_code": "REFUSED",
    "zbx_error_code": 0
}
```

Al especificar la dirección IP del DNS server, el item `net.dns.get[192.0.2.1,example.com]` puede devolver el siguiente JSON:

```
{
    "answer_section": [
        {
            "class": "IN",
            "name": "example.com.",
            "rdata": {
                "expire": 1209600,
                "mbox": "noc.dns.example.org.",
                "minttl": 3600,
```

```
                "ns": "ns.example.org.",
                "refresh": 7200,
                "retry": 3600,
                "serial": 2022091378
            },
            "rdlength": 44,
            "ttl": 1205,
            "type": "SOA"
        }
    ],
    "flags": [
        "RA"
    ],
    "query_time": "0.029556",
    "question_section": [
        {
            "qclass": "IN",
            "qname": "example.com.",
            "qtype": "SOA"
        }
    ],
    "response_code": "NOERROR",
    "zbx_error_code": 0
}
```


Si hay un problema de conexión, el item `net.dns.get[192.0.2.1,example.com]` puede devolver un error:


```
{
    "zbx_error_code": -1,
    "zbx_error_msg": "Communication error: read udp 192.0.2.1:12345->192.0.2.1:53: i/o timeout"
}
```


Son posibles los siguientes tipos de códigos de error:

Scenario	"zbx_error_code"	"zbx_error_msg"
No hay errores y la respuesta DNS se recibió y analizó.	0	
DNS no está disponible.	-1	"Communication error"
Se produce un error durante el análisis JSON	-2	"Received unexpected response"

Con parámetros adicionales, el item `net.dns.get[192.0.2.1,example.com,ANY,5,5,tcp,"cdflag,rdflag,dnssec,nsid,edns0,` puede devolver el siguiente JSON:


```
{
    "additional_section": [
        {
            "extended_rcode": 32768,
            "name": ".",
            "rdata": {
                "options": [
                {
                    "code": 0,
                    "nsid": "67 70 64 6e 73 2d 6c 70 70"
                }
                ]
        },
            "rdlength": 13,
            "type": "OPT",
            "udp_payload": 512
        }
    ],
    "answer_section": [
        {
```

```
            "class": "IN",
            "name": "example.com.",
            "rdata": {
                "a": "192.0.2.1"
            },
            "rdlength": 4,
            "ttl": 19308,
            "type": "A"
        },
        {
            "class": "IN",
            "name": "example.com.",
            "rdata": {
                "algorithm": 13,
                "expiration": 1704715951,
                "inception": 1702910624,
                "key_tag": 21021,
                "labels": 2,
                "orig_ttl": 86400,
                "signature": "HVBOBcJJQy0S08J3f8kviPj8UkEUj7wmyiMyQqPSWgQIY9SCEJ5plq6KuxJmtAek1txZWXDo+6tp
                "signer_name": "example.com.",
                "type_covered": "A"
            },
            "rdlength": 95,
            "ttl": 19308,
            "type": "RRSIG"
        }
    ],
    "flags": [
        "RD",
        "RA",
        "AD",
        "CD"
    ],
    "query_time": "0.058221",
    "question_section": [
        {
            "qclass": "IN",
            "qname": "example.com.",
            "qtype": "ANY"
        }
    ],
    "response_code": "NOERROR",
    "zbx_error_code": 0
}
```


Véase también

Para obtener más información sobre los registros DNS, consulte:

- Nombres de dominio - Implementación y especificación
- Parámetros del Sistema de Nombres de Dominio (DNS)

Valores de retorno para net.if.get

Resumen

Esta sección proporciona detalles sobre los valores devueltos para el item `net.if.get`. El item es compatible en Linux con Zabbix agent y Zabbix agent 2.

Detalles

Este item devuelve información detallada sobre las interfaces de red cuyos nombres coinciden con la expresión regular proporcionada. La salida está estructurada como un objeto JSON que contiene los arrays `config` y `values`.

La salida contiene las siguientes propiedades:

- **name** - nombre de la interfaz, leído de `/proc/net/dev`;
- **ifalias** - alias de la interfaz, leído de `/sys/class/net/<name>/ifalias`;
- **mac** - dirección MAC de la interfaz, leída de `/sys/class/net/<name>/address`;
- **type** - tipo de interfaz (`physical`, `virtual`, `loopback`);
- **speed** - velocidad del enlace en Mbits/s, específica de Ethernet, leída de `/sys/class/net/<name>/speed`;
- **duplex** - valor más reciente o actual de dúplex de la interfaz (`half`, `full`), específico de Ethernet, leído de `/sys/class/net/<name>/`
- **administrative_state** - estado configurado (`down`, `up`), leído de `/sys/class/net/<name>/flags`;
- **operational_state** - estado operativo RFC2863 de la interfaz (`unknown`, `notpresent`, `down`, `lowerlayerdown`, `testing`, `dormant`, `up`), leído de `sys/class/net/<name>/operstate`;
- **carrier** - presencia de enlace físico (0 - el enlace físico está caído, 1 - el enlace físico está activo), leído de `/sys/class/net/<name>/ca`
- **carrier_changes** - número total de veces que el estado del enlace cambió de activo a caído y viceversa, leído de `/sys/class/net/<name>/carrier_changes`;
- **carrier_up_count** - número de veces que el enlace ha pasado a estar activo, leído de `/sys/class/net/<name>/carrier_up_count`
- **carrier_down_count** - número de veces que el enlace ha pasado a estar caído, leído de `/sys/class/net/<name>/carrier_down_c`
- **in** - tráfico entrante, leído de `/proc/net/dev`. Contiene contadores de la interfaz:
  - **bytes** - número de bytes correctos;
  - **packets** - número de paquetes correctos;
  - **errors** - número total de paquetes erróneos;
  - **dropped** - número de paquetes recibidos pero no procesados + cantidad de paquetes perdidos por el host;
  - **overruns** - número de eventos de desbordamiento;
  - **frame** - número de paquetes con errores de trama;
  - **compressed** - número de paquetes comprimidos recibidos correctamente;
  - **multicast** - paquetes multicast recibidos.
- **out** - tráfico saliente, leído de `/proc/net/dev`. Contiene contadores de la interfaz:
  - **bytes** - número de bytes correctos;
  - **packets** - número de paquetes correctos;
  - **errors** - número total de paquetes erróneos;
  - **dropped** - número de paquetes recibidos pero no procesados + cantidad de paquetes perdidos por el host;
  - **overruns** - número de eventos de desbordamiento;
  - **collisions** - número de colisiones durante la transmisión de paquetes;
  - **carrier** - número de errores de transmisión de tramas debidos a la pérdida de portadora durante la transmisión;
  - **compressed** - número de paquetes comprimidos transmitidos.

Las métricas no disponibles se excluyen de la salida JSON. En el siguiente ejemplo, no hay ningún alias configurado para la interfaz "lo", por lo que la propiedad `ifalias` se excluye.

Ejemplo:


```
{
    "config": [
        {
            "name": "lo",
            "mac": "00:00:00:00:00:00",
            "type": "loopback",
            "administrative_state": "up",
            "operational_state": "unknown",
            "carrier": 1
        },
        {
            "name": "enp3s0",
            "ifalias": "Wired Ethernet connection",
            "mac": "8c:16:45:e3:1d:58",
            "type": "physical",
            "duplex": "unknown",
            "administrative_state": "up",
            "operational_state": "down",
            "carrier": 0
        },
        {
            "name": "wlp5s0",
            "ifalias": "Wi-Fi network adapter",
            "mac": "34:e1:2d:72:67:c6",
            "type": "physical",
```

```
            "administrative_state": "up",
            "operational_state": "up",
            "carrier": 1
        }
    ],
    "values": [
        {
            "name": "lo",
            "mac": "00:00:00:00:00:00",
            "carrier": 1,
            "carrier_changes": 0,
            "carrier_up_count": 0,
            "carrier_down_count": 0,
            "in": {
                "bytes": 1040047,
                "packets": 10784,
                "errors": 0,
                "dropped": 0,
                "overruns": 0,
                "frame": 0,
                "compressed": 0,
                "multicast": 0
            },
            "out": {
                "bytes": 1040047,
                "packets": 10784,
                "errors": 0,
                "dropped": 0,
                "overruns": 0,
                "collisions": 0,
                "carrier": 0,
                "compressed": 0
            }
        },
        {
            "name": "enp3s0",
            "ifalias": "Wired Ethernet connection",
            "mac": "8c:16:45:e3:1d:58",
            "carrier": 0,
            "carrier_changes": 1,
            "carrier_up_count": 0,
            "carrier_down_count": 1,
            "in": {
                "bytes": 0,
                "packets": 0,
                "errors": 0,
                "dropped": 0,
                "overruns": 0,
                "frame": 0,
                "compressed": 0,
                "multicast": 0
            },
            "out": {
                "bytes": 0,
                "packets": 0,
                "errors": 0,
                "dropped": 0,
                "overruns": 0,
                "collisions": 0,
                "carrier": 0,
                "compressed": 0
            }
```

```
        },
        {
            "name": "wlp5s0",
            "ifalias": "Wi-Fi network adapter",
            "mac": "34:e1:2d:72:67:c6",
            "carrier": 1,
            "carrier_changes": 6,
            "carrier_up_count": 3,
            "carrier_down_count": 3,
            "in": {
                "bytes": 165537298,
                "packets": 216182,
                "errors": 0,
                "dropped": 291,
                "overruns": 0,
                "frame": 0,
                "compressed": 0,
                "multicast": 0
            },
            "out": {
                "bytes": 252574347,
                "packets": 246607,
                "errors": 0,
                "dropped": 0,
                "overruns": 0,
                "collisions": 0,
                "carrier": 0,
                "compressed": 0
            }
        }
    ]
}
```


Véase también

Para obtener más información sobre la ABI de sysfs para dispositivos de red, consulte:

- Documentación del kernel de Linux

Valores de retorno para vfs.dev.get

Resumen

Esta sección proporciona detalles sobre el valor de retorno del item `vfs.dev.get`. El item es compatible en Linux con Zabbix agent y Zabbix agent 2, y en AIX con Zabbix agent.

Detalles

Este item devuelve información detallada sobre los dispositivos de bloque cuyos nombres coinciden con la expresión regular proporcionada.

La salida está estructurada como un objeto JSON que contiene `config` y, según el parámetro <mode>, matrices `values`.

La matriz `config` contiene las propiedades disponibles para el modo seleccionado. La matriz `values` se devuelve solo para los modos que proporcionan estadísticas de dispositivos de bloque.

Las métricas no disponibles se excluyen de la salida JSON.

Las siguientes secciones describen las propiedades devueltas para cada modo.

modo discos

`vfs.dev.get[disks,.*]` devuelve datos de configuración de dispositivos de bloque.

Propiedades:

- **name** - nombre del dispositivo de bloque;
- **devid** - identificador del dispositivo de bloque, por ejemplo un ID de dispositivo basado en WWN o generado por el kernel;
- **type** - tipo de dispositivo de bloque (`disk`, `partition`, `rom`);

- **path** - ruta al nodo del dispositivo de bloque;
- **model** - modelo del dispositivo;
- **serial** - número de serie del dispositivo;
- **wwn** - World Wide Name (WWN) del dispositivo;
- **size_bytes** - tamaño del dispositivo en bytes;
- **logical_block_size** - tamaño lógico de bloque en bytes;
- **physical_block_size** - tamaño físico de bloque en bytes.

Ejemplo para Linux:


```
{
  "config": [
    {
      "name": "sda",
      "devid": "ata-ST2000LM015-2E8174_ZDZ3YHZY",
      "type": "disk",
      "path": "/dev/sda",
      "model": " ST2000LM015-2E81",
      "wwn": "naa.5000c500b2791423",
      "size_bytes": 2000398934016,
      "logical_block_size": 512,
      "physical_block_size": 4096
    },
    {
      "name": "sdb",
      "devid": "usb-Kingston_DataTraveler_3.0_E0D55E6C382C15A158991CBD-0:0",
      "type": "disk",
      "path": "/dev/sdb",
      "model": "DataTraveler 3.0",
      "size_bytes": 61991813632,
      "logical_block_size": 512,
      "physical_block_size": 512
    },
    {
      "name": "nvme0n1",
      "devid": "nvme-SAMSUNG_MZVLW256HEHP-000L7_S35ENX1K753633",
      "type": "disk",
      "path": "/dev/nvme0n1",
      "model": "SAMSUNG MZVLW256HEHP-000L7",
      "serial": "S35ENX1K753633",
      "wwn": "eui.002538b781b7dc55",
      "size_bytes": 256060514304,
      "logical_block_size": 512,
      "physical_block_size": 512
    }
  ]
}
```


Ejemplo para AIX:


```
{
  "config": [
    {
      "name": "cd0",
      "type": "disk",
      "path": "/dev/cd0",
      "size_bytes": 0,
      "logical_block_size": 512
    },
    {
      "name": "hdisk1",
      "type": "disk",
      "devid": "48333321360050768OC818022F000000000015EA404214503IBMfcp05VDASD03AIXvscsi",
      "path": "/dev/hdisk1",
      "size_bytes": 51136000000,
```

```
      "logical_block_size": 512
    },
    {
      "name": "hdisk0",
      "type": "disk",
      "devid": "48333321360050768OC818022F000000000015EA304214503IBMfcp05VDASD03AIXvscsi",
      "path": "/dev/hdisk0",
      "size_bytes": 0,
      "logical_block_size": 512
    }
  ]
}
```


modo dispositivos

`vfs.dev.get[devices,.*]` devuelve datos de configuración de dispositivos de bloque, incluidas las particiones.

Propiedades:

- **name** - nombre del dispositivo de bloque.
- **devid** - identificador del dispositivo de bloque.
- **type** - tipo de dispositivo de bloque (`disk`, `partition`, `rom`).
- **partitions** - particiones del dispositivo. Contiene los nombres de las particiones y sus tamaños en bytes.

Ejemplo para Linux:


```
{
  "config": [
    {
      "name": "sda",
      "devid": "ata-ST2000LM015-2E8174_ZDZ3YHZY",
      "type": "disk",
      "partitions": {
        "sda1": 16759808,
        "sda2": 2000381018112
      }
    },
    {
      "name": "sdb",
      "devid": "usb-Kingston_DataTraveler_3.0_E0D55E6C382C15A158991CBD-0:0",
      "type": "disk",
      "partitions": {
        "sdb1": 27631878144,
        "sdb2": 34359768576,
        "sdb3": 32256
      }
    },
    {
      "name": "nvme0n1",
      "devid": "nvme-SAMSUNG_MZVLW256HEHP-000L7_S35ENX1K753633",
      "type": "disk",
      "partitions": {
        "nvme0n1p1": 209715200,
        "nvme0n1p2": 16777216,
        "nvme0n1p3": 255060869120,
        "nvme0n1p4": 770703360
      }
    }
  ]
}
```


modo disk_stats

`vfs.dev.get[disk_stats,.*]` devuelve datos de configuración de dispositivos de bloque y estadísticas de disco.

Propiedades:

- **name** - nombre del dispositivo de bloque.
- **devid** - identificador del dispositivo de bloque.
- **type** - tipo de dispositivo de bloque (`disk`, `partition`, `rom`).
- **size_bytes** - tamaño del dispositivo en bytes.
- **logical_block_size** - tamaño lógico del bloque en bytes.
- **stats** - estadísticas de E/S del dispositivo de bloque. Contiene:
  - **reads_completed** - número de operaciones de lectura completadas;
  - **writes_completed** - número de operaciones de escritura completadas;
  - **bytes_read** - número de bytes leídos;
  - **bytes_written** - número de bytes escritos;
  - **io_time_ms** - tiempo total dedicado a E/S, en milisegundos.

Ejemplo para Linux


```
{
  "config": [
    {
      "name": "sda",
      "devid": "ata-ST2000LM015-2E8174_ZDZ3YHZY",
      "type": "disk",
      "size_bytes": 61991813632
    },
    {
      "name": "sdb",
      "devid": "usb-Kingston_DataTraveler_3.0_E0D55E6C382C15A158991CBD-0:0",
      "type": "disk",
      "size_bytes": 61991813632
    },
    {
      "name": "nvme0n1",
      "devid": "nvme-SAMSUNG_MZVLW256HEHP-000L7_S35ENX1K753633",
      "type": "disk",
      "size_bytes": 256060514304
    }
  ],
  "values": [
    {
      "name": "sda",
      "stats": {
        "reads_completed": 871,
        "writes_completed": 0,
        "bytes_read": 13045760,
        "bytes_written": 0,
        "io_time_ms": 1256
      }
    },
    {
      "name": "sdb",
      "stats": {
        "reads_completed": 51017,
        "writes_completed": 10682,
        "bytes_read": 10514627,
        "bytes_written": 1530472,
        "io_time_ms": 284988
      }
    },
    {
      "name": "nvme0n1",
      "stats": {
        "reads_completed": 1259,
        "writes_completed": 7,
        "bytes_read": 49698,
        "bytes_written": 6,
        "io_time_ms": 243
```

```
      }
    }
  ]
}
```

Ejemplo para AIX:

```
{
  "config": [
    {
      "name": "cd0",
      "type": "disk",
      "path": "/dev/cd0",
      "size_bytes": 0,
      "logical_block_size": 2048
    },
    {
      "name": "hdisk1",
      "type": "disk",
      "devid": "48333321360050768OC818022F000000000015EA404214503IBMfcp05VDASD03AIXvscsi",
      "path": "/dev/hdisk1",
      "size_bytes": 51136000000,
      "logical_block_size": 512
    },
    {
      "name": "hdisk0",
      "type": "disk",
      "devid": "48333321360050768OC818022F000000000015EA304214503IBMfcp05VDASD03AIXvscsi",
      "path": "/dev/hdisk0",
      "size_bytes": 0,
      "logical_block_size": 512
    }
  ],
  "values": [
    {
      "name": "cd0",
      "stats": {
        "reads_completed": 21,
        "writes_completed": 0,
        "bytes_read": 75776,
        "bytes_written": 0,
        "io_time_ms": 0
      }
    },
    {
      "name": "hdisk1",
      "stats": {
        "reads_completed": 35888,
        "writes_completed": 1165147,
        "bytes_read": 593699840,
        "bytes_written": 10009646592,
        "io_time_ms": 69
      }
    },
    {
      "name": "hdisk0",
      "stats": {
        "reads_completed": 3,
        "writes_completed": 0,
        "bytes_read": 12800,
        "bytes_written": 0,
        "io_time_ms": 0
      }
```

```
    }
  ]
}
```

modo device_stats

`vfs.dev.get[device_stats,.*]` devuelve datos de configuración de dispositivos de bloque y estadísticas de dispositivos.

Propiedades:

- **name** - nombre del dispositivo de bloque.
- **devid** - identificador del dispositivo de bloque.
- **type** - tipo de dispositivo de bloque (`disk`, `partition`, `rom`).
- **size_bytes** - tamaño del dispositivo en bytes.
- **stats** - estadísticas de E/S del dispositivo de bloque. Contiene:
  - **reads_completed** - número de operaciones de lectura completadas;
  - **writes_completed** - número de operaciones de escritura completadas;
  - **bytes_read** - número de bytes leídos;
  - **bytes_written** - número de bytes escritos;
  - **io_time_ms** - tiempo total dedicado a E/S, en milisegundos.

Ejemplo para Linux:

```
{
  "config": [
    {
      "name": "sda",
      "devid": "ata-ST2000LM015-2E8174_ZDZ3YHZY",
      "type": "disk",
      "size_bytes": 61991813632
    },
    {
      "name": "sda1",
      "type": "partition",
      "devid": "ata-ST2000LM015-2E8174_ZDZ3YHZY-part1",
      "size_bytes": 1234567890
    },
    {
      "name": "sda2",
      "type": "partition",
      "devid": "ata-ST2000LM015-2E8174_ZDZ3YHZY-part2",
      "size_bytes": 1234567890
    },
    {
      "name": "sdb",
      "devid": "usb-Kingston_DataTraveler_3.0_E0D55E6C382C15A158991CBD-0:0",
      "type": "disk",
      "size_bytes": 61991813632
    },
    {
      "name": "sdb1",
      "type": "partition",
      "devid": "usb-Kingston_DataTraveler_3.0_E0D55E6C382C15A158991CBD-0:0-part1",
      "size_bytes": 1234567890
    },
    {
      "name": "sdb2",
      "type": "partition",
      "devid": "usb-Kingston_DataTraveler_3.0_E0D55E6C382C15A158991CBD-0:0-part2",
      "size_bytes": 1234567890
    },
    {
      "name": "sdb3",
      "type": "partition",
      "devid": "usb-Kingston_DataTraveler_3.0_E0D55E6C382C15A158991CBD-0:0-part3",
```

```
      "size_bytes": 1234567890
    },
    {
      "name": "nvme0n1",
      "devid": "nvme-SAMSUNG_MZVLW256HEHP-000L7_S35ENX1K753633",
      "type": "disk",
      "size_bytes": 256060514304
    },
    {
      "name": "nvme0n1p1",
      "type": "partition",
      "devid": "nvme-SAMSUNG_MZVLW256HEHP-000L7_S35ENX1K753633-part1",
      "size_bytes": 1234567890
    },
    {
      "name": "nvme0n1p2",
      "type": "partition",
      "devid": "nvme-SAMSUNG_MZVLW256HEHP-000L7_S35ENX1K753633-part2",
      "size_bytes": 1234567890
    },
    {
      "name": "nvme0n1p3",
      "type": "partition",
      "devid": "nvme-SAMSUNG_MZVLW256HEHP-000L7_S35ENX1K753633-part3",
      "size_bytes": 1234567890
    },
    {
      "name": "nvme0n1p4",
      "type": "partition",
      "devid": "nvme-SAMSUNG_MZVLW256HEHP-000L7_S35ENX1K753633-part4",
      "size_bytes": 1234567890
    }
  ],
  "values": [
    {
      "name": "sda",
      "stats": {
        "reads_completed": 871,
        "writes_completed": 0,
        "bytes_read": 13045760,
        "bytes_written": 0,
        "io_time_ms": 1256
      }
    },
    {
      "name": "sda1",
      "stats": {
        "reads_completed": 123,
        "writes_completed": 123,
        "bytes_read": 123,
        "bytes_written": 123,
        "io_time_ms": 123
      }
    },
    {
      "name": "sda2",
      "stats": {
        "reads_completed": 123,
        "writes_completed": 123,
        "bytes_read": 123,
        "bytes_written": 123,
        "io_time_ms": 123
```

```
    }
  },
  {
    "name": "sdb",
    "stats": {
      "reads_completed": 51017,
      "writes_completed": 10682,
      "bytes_read": 10514627,
      "bytes_written": 1530472,
      "io_time_ms": 284988
    }
  },
  {
    "name": "sdb1",
    "stats": {
      "reads_completed": 123,
      "writes_completed": 123,
      "bytes_read": 123,
      "bytes_written": 123,
      "io_time_ms": 123
    }
  },
  {
    "name": "sdb2",
    "stats": {
      "reads_completed": 123,
      "writes_completed": 123,
      "bytes_read": 123,
      "bytes_written": 123,
      "io_time_ms": 123
    }
  },
  {
    "name": "sdb3",
    "stats": {
      "reads_completed": 123,
      "writes_completed": 123,
      "bytes_read": 123,
      "bytes_written": 123,
      "io_time_ms": 123
    }
  },
  {
    "name": "nvme0n1",
    "stats": {
      "reads_completed": 1259,
      "writes_completed": 7,
      "bytes_read": 49698,
      "bytes_written": 6,
      "io_time_ms": 243
    }
  },
  {
    "name": "nvme0n1p1",
    "stats": {
      "reads_completed": 123,
      "writes_completed": 123,
      "bytes_read": 123,
      "bytes_written": 123,
      "io_time_ms": 123
    }
  },
```

```
    {
      "name": "nvme0n1p2",
      "stats": {
        "reads_completed": 123,
        "writes_completed": 123,
        "bytes_read": 123,
        "bytes_written": 123,
        "io_time_ms": 123
      }
    },
    {
      "name": "nvme0n1p3",
      "stats": {
        "reads_completed": 123,
        "writes_completed": 123,
        "bytes_read": 123,
        "bytes_written": 123,
        "io_time_ms": 123
      }
    },
    {
      "name": "nvme0n1p4",
      "stats": {
        "reads_completed": 123,
        "writes_completed": 123,
        "bytes_read": 123,
        "bytes_written": 123,
        "io_time_ms": 123
      }
    }
  ]
}
```

Notas sobre los items system.cpu.util en Windows

La métrica system.cpu.util proporciona el porcentaje de utilización de la CPU.

Cuando se inicia un proceso recolector en el agente Zabbix para Windows, se inicializa un búfer para N métricas para N procesadores lógicos (hilos). Los valores se actualizan periódicamente en el búfer utilizando contadores de rendimiento de Windows. Estos valores se informan cuando se solicita `system.cpu.util[n]`, donde 0 <= n < N.

Se utiliza una métrica adicional en el búfer para `system.cpu.util[total]`.

Zabbix selecciona automáticamente diferentes contadores de rendimiento para admitir tanto sistemas NUMA como versiones antiguas de Windows sin soporte para grupos de procesadores.

Si el número de procesadores lógicos (hilos) es menor o igual a 64 y el número de grupos de procesadores es igual a 1, entonces:

```
N es el número de procesadores lógicos (hilos).

\Processor(_Total)\% Processor Time
\Processor(0)\% Processor Time
\Processor(1)\% Processor Time
\Processor(2)\% Processor Time
...
\Processor(N-1)\% Processor Time
```

De lo contrario:

```
G es el número de grupos de procesadores y N es el número de procesadores lógicos (hilos) en el grupo.

\Processor Information(_Total)\% Processor Time
\Processor Information(0,0)\% Processor Time
\Processor Information(0,1)\% Processor Time
\Processor Information(0,2)\% Processor Time
```

```
...
\Processor Information(0,N-1)\% Processor Time
...
\Processor Information(G-1,0)\% Processor Time
\Processor Information(G-1,1)\% Processor Time
\Processor Information(G-1,2)\% Processor Time
...
\Processor Information(G-1,N-1)\% Processor Time
```

Compatibilidad con valores JSON grandes

Para garantizar un manejo fiable de valores grandes con el tipo de datos JSON data type (1 MiB o más), revise y aplique los siguientes cambios de configuración según sea necesario:

1. Si usa MySQL o MariaDB, aumente el tamaño máximo permitido del paquete en `/etc/mysql/my.cnf` (en algunos sistemas puede tener un valor predeterminado de 1 MB):

```
[mysqld]
max_allowed_packet=128M

[mysqldump]
max_allowed_packet=1G
```

2. Ajuste otras server system variables de MySQL/MariaDB y InnoDB system variables según sea necesario. Por ejemplo:

```
innodb_io_capacity=1900
connect_timeout=600
wait_timeout=57600
interactive_timeout=57600
net_read_timeout=7200
net_write_timeout=7200
net-buffer-length=32704
```

3. Aumente `HistoryCacheSize` a 2G en el archivo de configuración del server de Zabbix.

4. Si usa Zabbix proxy, aumente `HistoryCacheSize` a 2G y `ProxyMemoryBufferSize` a 2G en el archivo de configuración del proxy.

5. Si usa Zabbix agent, aumente los tiempos de espera de comunicación entre el server/proxy de Zabbix y el agent (por ejemplo, los item timeouts o el valor del parámetro `Timeout` en los archivos de configuración del server/proxy y del agent de Zabbix). De lo contrario, los mensajes entre ellos pueden no completarse y es posible que vea errores como *"message length does not match expected length"*. Además, revise otros tiempos de espera según sea necesario, como el tiempo de espera de script timeout o de item test.

Funciones compatibles

Consulte Expresión de trigger y funciones.

Macros

Es posible usar Macros compatibles de forma predeterminada y Macros de usuario compatibles por ubicación.

Macros compatibles con la ubicación

Resumen

La tabla de macros compatibles por ubicación contiene una referencia de todas las macros (integradas, de expresión y definibles por el usuario) que son compatibles en la ubicación específica.

Si todas las macros de un grupo relacionado son compatibles, solo se indica el nombre del grupo. Haga clic en cada nombre de macro o nombre de grupo para ver los detalles completos de las macros.

La lista de detalles de macros contiene una lista detallada de todas las macros **integradas**, agrupadas por área de aplicación. Tenga en cuenta que, para personalizar los valores de las macros (por ejemplo, acortar o extraer subcadenas específicas), puede usar funciones de macro.

Macros compatibles por ubicación

Location	Supported macros	Comments
Action		
Time period condition, Default operation step duration, Step duration	macros definibles por el usuario (solo globales)	Un solo macro debe ocupar todo el campo; no se admiten varios macros ni la mezcla con texto
Notifications and commands		
Trigger-based	macros Action macros de fecha/hora {ESC.HISTORY} macros de Event, macros de Cause/symptom event macros de actualización de event: {EVENT.UPDATE.HISTORY}, {EVENT.UPDATE.STATUS} macro de expresión: {?EXPRESSION} {FUNCTION.VALUE} macros de host: {HOST.CONN}, {HOST.DESCRIPTION}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} macros de host de destino (solo comandos) macros de Host inventory macros de Item (excepto {ITEM.STATE}, {ITEM.STATE.ERROR}) macros de Proxy macros de trigger: {TRIGGER.DESCRIPTION}, {TRIGGER.EVENTS.*}, {TRIGGER.EXPRESSION}, {TRIGGER.EXPRESSION.EXPLAIN}, {TRIGGER.EXPRESSION.RECOVERY}, {TRIGGER.ID}, {TRIGGER.HOSTGROUP.NAME}, {TRIGGER.NAME}, {TRIGGER.NAME.ORIG}, {TRIGGER.NSEVERITY}, {TRIGGER.SEVERITY}, {TRIGGER.STATUS}, {TRIGGER.TEMPLATE.NAME}, {TRIGGER.URL}, {TRIGGER.URL.NAME}, {TRIGGER.VALUE} macros definibles por el usuario	Véase también: Using macros in messages

Location	Supported macros	Comments
Problem update	macros Action macros de fecha/hora {ESC.HISTORY} macros de Event, macros de Cause/symptom event, macros de Recovery event macros de Event update: todas (excepto {EVENT.UPDATE.NSEVERITY}, {EVENT.UPDATE.SEVERITY}) macro de expresión: {?EXPRESSION} macros de Function macros de host: {HOST.CONN}, {HOST.DESCRIPTION}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} macros de host de destino (solo comandos) macros de Host inventory macros de Item (excepto {ITEM.STATE}, {ITEM.STATE.ERROR}) macros de Proxy macros de trigger: {TRIGGER.DESCRIPTION}, {TRIGGER.EVENTS.*}, {TRIGGER.EXPRESSION}, {TRIGGER.EXPRESSION.EXPLAIN}, {TRIGGER.EXPRESSION.RECOVERY}, {TRIGGER.ID}, {TRIGGER.HOSTGROUP.NAME}, {TRIGGER.NAME}, {TRIGGER.NAME.ORIG}, {TRIGGER.NSEVERITY}, {TRIGGER.SEVERITY}, {TRIGGER.STATUS}, {TRIGGER.TEMPLATE.NAME}, {TRIGGER.URL}, {TRIGGER.URL.NAME}, {TRIGGER.VALUE} {USER.FULLNAME} macros definibles por el usuario	
Problem recovery	macros de Recovery event macros de Event {FUNCTION.RECOVERY.VALUE} {TRIGGER.EXPRESSION.RECOVERY.EXPLAIN}	
Discovery	macros Action macros de fecha/hora macros de Discovery macros de Event: {EVENT.AGE}, {EVENT.DATE}, {EVENT.ID}, {EVENT.OBJECT}, {EVENT.SOURCE}, {EVENT.TIME}, {EVENT.TIMESTAMP} macros de host de destino (solo comandos) macros de Proxy	
Autoregistration	macros Action macros de fecha/hora macros de Event: {EVENT.AGE}, {EVENT.DATE}, {EVENT.ID}, {EVENT.OBJECT}, {EVENT.SOURCE}, {EVENT.TIME}, {EVENT.TIMESTAMP} macros de host: {HOST.HOST}, {HOST.IP}, {HOST.PORT}, {HOST.METADATA} macros de host de destino (solo comandos) macros de Proxy	
Service-based	macros Action macros de fecha/hora {ESC.HISTORY} macros de Event: todas excepto {EVENT.ACK.STATUS}, {EVENT.OPDATA} macros de Service macros definibles por el usuario	
Service recovery	macros de Event (excepto {EVENT.ACK.STATUS}, {EVENT.OPDATA}), macros de Recovery event	

Location	Supported macros	Comments
Service update	macros Action macros de fecha/hora {ESC.HISTORY} macros de Event (excepto {EVENT.ACK.STATUS}, {EVENT.OPDATA}) macros de actualización de event: {EVENT.UPDATE.DATE}, {EVENT.UPDATE.NSEVERITY}, {EVENT.UPDATE.SEVERITY}, {EVENT.UPDATE.STATUS}, {EVENT.UPDATE.TIME}, {EVENT.UPDATE.TIMESTAMP} macros de Service macros definibles por el usuario	
Internal notifications	Action macros Date/time macros {ESC.HISTORY} Event macros (excepto {EVENT.NSEVERITY}, {EVENT.SEVERITY}), Recovery event macros macros de host: {HOST.CONN}, {HOST.DESCRIPTION}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Host inventory macros macros de Item: {ITEM.DESCRIPTION}, {ITEM.DESCRIPTION.ORIG}, {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG}, {ITEM.NAME}, {ITEM.NAME.ORIG}, {ITEM.VALUETYPE} Proxy macros	
Item-based	macros de Item: {ITEM.STATE}, {ITEM.STATE.ERROR}	
LLD rule-based	macros de Low-level discovery notification	
Trigger-based	macros de trigger: {TRIGGER.DESCRIPTION}, {TRIGGER.EXPRESSION}, {TRIGGER.EXPRESSION.RECOVERY}, {TRIGGER.HOSTGROUP.NAME}, {TRIGGER.ID}, {TRIGGER.NAME}, {TRIGGER.NAME.ORIG}, {TRIGGER.NSEVERITY}, {TRIGGER.SEVERITY}, {TRIGGER.STATE}, {TRIGGER.STATE.ERROR}, {TRIGGER.TEMPLATE.NAME}, {TRIGGER.URL}, {TRIGGER.URL.NAME} macros definibles por el usuario	
Alert script parameters	Alert macros macros definibles por el usuario (solo globales)	
Connector		
URL, Username, Password, Bearer token, HTTP proxy, SSL certificate, SSL key file, SSL key password field	macros definibles por el usuario (solo globales)	
Timeout field	macros definibles por el usuario (solo globales)	Un solo macro debe ocupar todo el campo; no se admiten varios macros ni la mezcla con texto.

Location	Supported macros	Comments
Graph name	macro de expresión: {?EXPRESSION}	En este macro, en esta ubicación, solo se admite una única función **avg**, **last**, **max** o **min** con segundos como parámetro (se pueden usar sufijos de tiempo). El macro {HOST.HOST<1-9>} puede usarse como host dentro del macro (véase también indexed macros). Ejemplo de uso: {?avg(/{HOST.HOST}/item.key,1h)} Uso no admitido: {?last(/host/item1)/last(/host/item2)} {?last(/host/item1)*10} {?count(/host/item1,5m)}
Host, host prototype		
Interface IP/DNS	macros de host: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME} macros definibles por el usuario [1]	[1] Para la dirección IP, un solo macro debe ocupar todo el campo; no se admiten varios macros ni la mezcla con texto
Interface port	macros definibles por el usuario	Un solo macro debe ocupar todo el campo; no se admiten varios macros ni la mezcla con texto.
SNMP community *SNMPv3 Context name, Security name, Authentication passphrase, Privacy passphrase* *IPMI username, password*	macros definibles por el usuario	
Item, item prototype, LLD rule		
Name	macros definibles por el usuario	No se admite en los nombres de reglas LLD.
Description	macros definibles por el usuario	

Location	Supported macros	Comments
Key parameters	macros de host: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} macros definibles por el usuario	Los macros {HOST.*} admitidos se resolverán en la interfaz seleccionada para el item. Cuando se usan en items sin interfaces, se resolverán en la interfaz de Zabbix agent, SNMP, JMX o IPMI del host, en ese orden de prioridad, o en 'UNKNOWN' si el host no tiene ninguna interfaz. Los macros no se admiten en la clave del item de los items Zabbix trapper.
Update interval, *Custom intervals*, *Timeout (available for supported item types)*, *Store up to (for history/trends)* fields	macros definibles por el usuario	Un solo macro debe ocupar todo el campo; no se admiten varios macros ni la mezcla con texto.
Delete lost resources, *Disable lost resources* fields	macros definibles por el usuario	Estos campos solo se admiten para reglas LLD. Un solo macro debe ocupar todo el campo; no se admiten varios macros ni la mezcla con texto.
Preprocessing step parameters (including custom scripts), *custom error-handling parameters* (*Set value to* and *Set error to* fields)	macros definibles por el usuario	
Filter regular expressions	macros de host: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} macros definibles por el usuario	Estos campos solo se admiten para reglas LLD.
Override filter regular expressions	macros definibles por el usuario	Estos campos solo se admiten para reglas LLD.
Override operations: *Update interval*, *History storage period*, *Trend storage period* (for item prototypes)	macros definibles por el usuario	Estos campos solo se admiten para reglas LLD. Un solo macro debe ocupar todo el campo; no se admiten varios macros ni la mezcla con texto.
Item types		

Location	Supported macros	Comments
Browser item *parameter names and values*	macros de host: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} macros de Host inventory macros de Item: {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG} macros definibles por el usuario	
Script item *parameter names and values*		
Calculated/aggregate item formula (*expression constants*, *function parameters*, *item key parameters*, (aggregate item only) *filter conditions* (host group name and tag name))	macros definibles por el usuario	
Database monitor item *Username, Password, SQL query* fields	macros definibles por el usuario	
HTTP agent item *URL, Query fields, Request body, Headers (names and values), SSL certificate file, SSL key file* fields	macros de host: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} macros de Item: {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG} macros definibles por el usuario	Las URL que contienen un secret macro no funcionarán, ya que el macro en ellas se resolverá como "******"
HTTP agent item *HTTP Proxy* field	macros de Item: {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG} macros definibles por el usuario	
HTTP agent item *Required status codes, HTTP authentication username/password, SSL key password* fields	macros definibles por el usuario	
HTTP agent item *Allowed hosts* field	macros de host: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME} macros definibles por el usuario	
JMX agent item *Endpoint* field	macros de host: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.PORT} macros definibles por el usuario	
Trapper item *Allowed hosts* field	macros de host: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP} macros definibles por el usuario	
SNMP agent item *SNMP OID* field	macros definibles por el usuario	
SSH and Telnet item *script*	macros de host: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} macros definibles por el usuario	
SSH agent item *Username, Public key file, Private key file, Password* fields	macros definibles por el usuario	
Telnet agent item *Username, Password* fields	macros definibles por el usuario	
Map		

Location	Supported macros	Comments
URL, URL name	macros de host: {HOST.CONN}, {HOST.DESCRIPTION}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} {HOSTGROUP.ID} macros de Host inventory macros de Map {TRIGGER.ID}	
Element label	macro de expresión: {?EXPRESSION} [1] macros de host: {HOST.CONN}, {HOST.DESCRIPTION}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} {HOSTGROUP.ID} macros de Host inventory macros de Map macros de trigger: {TRIGGER.EVENTS.*}, {TRIGGER.ID}, {TRIGGER.PROBLEM.EVENTS.*}, {TRIGGERS.ACK}, {TRIGGERS.PROBLEM.ACK}, {TRIGGERS.PROBLEM.UNACK}, {TRIGGERS.UNACK}	[1] Solo se admite una única función **avg**, **last**, **max** o **min** con segundos como parámetro (se pueden usar sufijos de tiempo) en este macro en esta ubicación. El macro {HOST.HOST<1-9>} puede usarse como host dentro del macro. Ejemplo de uso: {?avg(/{HOST.HOST}/item.key,1h)} Uso no admitido: {?last(/host/item1)/last(/host/item2)} {?last(/host/item1)*10} {?count(/host/item1,5m)}
Link label	macro de expresión: {?EXPRESSION} [1]	
Shape label	macro de expresión: {?EXPRESSION} [1]	
Text field in map shapes	{MAP.NAME}	
Network discovery		
Update interval field	macros definibles por el usuario (solo globales)	Un solo macro debe ocupar todo el campo; no se admiten varios macros ni la mezcla con texto.
SNMP community, SNMP OID fields *SNMPv3 Context name, Security name, Authentication passphrase, Privacy passphrase* fields	macros definibles por el usuario (solo globales)	
Proxy		
Address for active agents > *Port* (if the proxy belongs to a group) For passive proxy mode: interface *Address* and *Port* fields *Timeouts for item types* fields in override mode	macros definibles por el usuario (solo globales)	Un solo macro debe ocupar todo el campo; no se admiten varios macros ni la mezcla con texto.
Proxy group		

Location	Supported macros	Comments
Failover period, Minimum number of proxies fields	macros definibles por el usuario (solo globales)	Un solo macro debe ocupar todo el campo; no se admiten varios macros ni la mezcla con texto.
Scripts		
Commands, confirmation text (Type: script, manual **host action**)	macros de host: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.ID} [1], {HOST.IP}, {HOST.NAME}, {HOST.PORT} macros de Host inventory [2] {MANUALINPUT} macros de Username macros definibles por el usuario	[1] Solo texto de confirmación [2] Compatible con Zabbix server y Zabbix proxy
Commands (Type: script, manual **event action**)	macros de fecha/hora macros de Event, macros de Cause/symptom event, macros de Recovery event (si tuvo lugar la recuperación) macros de actualización de event: {EVENT.UPDATE.HISTORY}, {EVENT.UPDATE.STATUS} {FUNCTION.VALUE} macros de host: {HOST.CONN}, {HOST.DESCRIPTION}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} macros de Host inventory macros de Item (excepto {ITEM.STATE}, {ITEM.STATE.ERROR}) {MANUALINPUT} macros de Proxy macros de trigger: {TRIGGER.DESCRIPTION}, {TRIGGER.EVENTS.*}, {TRIGGER.EXPRESSION}, {TRIGGER.EXPRESSION.EXPLAIN}, {TRIGGER.EXPRESSION.RECOVERY}, {TRIGGER.EXPRESSION.RECOVERY.EXPLAIN}, {TRIGGER.HOSTGROUP.NAME}, {TRIGGER.ID}, {TRIGGER.NAME}, {TRIGGER.NAME.ORIG}, {TRIGGER.NSEVERITY}, {TRIGGER.SEVERITY}, {TRIGGER.STATUS}, {TRIGGER.TEMPLATE.NAME}, {TRIGGER.URL}, {TRIGGER.URL.NAME}, {TRIGGER.VALUE} macros de Username macros definibles por el usuario	
Confirmation text (Type: script, manual **event action**)	macros de Event: {EVENT.ID}, {EVENT.NAME}, {EVENT.NSEVERITY}, {EVENT.SEVERITY}, {EVENT.STATUS}, {EVENT.VALUE} macros de host: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} macros de Host inventory {MANUALINPUT} macros de Username macros definibles por el usuario	
URL, confirmation text (Type: URL, manual **host/event action**)		
Tag name and value	macros de host: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} macros de Host inventory macros de Item: {ITEM.LASTVALUE*}, {ITEM.LOG.*}, {ITEM.VALUE*} macros de trigger: {TRIGGER.ID} (solo valor de etiqueta de trigger) macros definibles por el usuario	Los macros de nombre y valor de etiqueta solo se resuelven durante el proceso de generación del event.
Trigger		

Location	Supported macros	Comments
Name	macros de host: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} macros de Item: {ITEM.LASTVALUE*}, {ITEM.LOG.*}, {ITEM.VALUE*} macros/referencias posicionales: $1...$9 macros definibles por el usuario	
Event name	macros de fecha/hora: {TIME}, {TIMESTAMP} macro de expresión: {?EXPRESSION} macros de Function macros de host: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} macros de Item: {ITEM.LASTVALUE*}, {ITEM.LOG.*}, {ITEM.VALUE*} macros de trigger: {TRIGGER.EXPRESSION.EXPLAIN}	
Operational data	macros de host: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} macros de Item: {ITEM.LASTVALUE*}, {ITEM.LOG.*}, {ITEM.VALUE*} macros definibles por el usuario	
Expression	{TRIGGER.VALUE} macros definibles por el usuario [1]	[1] Solo en constantes y parámetros de función; no se admiten macros secretos
Tag for matching, Menu entry name, Menu entry URL fields	macros definibles por el usuario	Las URL que contienen un secret macro no funcionarán, ya que el macro en ellas se resolverá como "******".
Description	macros de host: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} macros de Item: {ITEM.LASTVALUE*}, {ITEM.LOG.*}, {ITEM.VALUE*} macros definibles por el usuario	
URL	{EVENT.ID} macros de host: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} macros de Item: {ITEM.LASTVALUE*}, {ITEM.LOG.*}, {ITEM.VALUE*} {TRIGGER.ID}	
User parameter commands	macros/referencias posicionales: $1...$9	
Web scenario		
Name, Agent, HTTP proxy fields	macros definibles por el usuario	
Update interval field	macros definibles por el usuario	Un solo macro debe ocupar todo el campo; no se admiten varios macros ni la mezcla con texto.
Variables, Headers, SSL certificate, SSL key file fields	macros de host: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} macros definibles por el usuario	
User, Password, SSL key password fields	macros definibles por el usuario	

Location	Supported macros	Comments
Web scenario step *Name, Variables (values only)* fields	macros definibles por el usuario	
Web scenario step *Timeout, Required status codes* fields	macros definibles por el usuario	Un solo macro debe ocupar todo el campo; no se admiten varios macros ni la mezcla con texto.
Web scenario step *URL, Headers (names and values), Required string* fields	macros de host: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} macros definibles por el usuario	
Web scenario step *Post* field	macros de host: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT}	
Webhook media type		
Parameter values	macros Alert Todos los macros compatibles con las notificaciones de problemas basadas en trigger (véase arriba)	Un solo macro debe ocupar todo el campo; no se admiten varios macros ni la mezcla con texto.
JavaScript script, *JavaScript* script parameter name	macros definibles por el usuario (solo globales)	
Menu entry name, *Menu entry URL*	{EVENT.TAGS.<tag name>}	
Widgets		
Gauge widget *Description* field	macros de host: {HOST.CONN}, {HOST.DESCRIPTION}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} macros de Host inventory macros de Item (excepto {ITEM.STATE.ERROR}) macros definibles por el usuario (solo globales)	
Honeycomb widget *Primary/secondary label* field **Item value** widget *Description* field		
Top hosts widget *Text data* column	macros de host: {HOST.CONN}, {HOST.DESCRIPTION}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} macros de Host inventory macros definibles por el usuario (solo globales)	
URL (Dynamic) widget *URL* field	macros de host: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} macros definibles por el usuario	
***Custom text* column**		
Monitoring > Problems	macros Host (excepto {HOST.METADATA}) macros de Host inventory macros de Item (excepto {ITEM.STATE.ERROR}) macros de trigger: {TRIGGER.ID} macros de Event: {EVENT.ID} macros definibles por el usuario	
Monitoring > Hosts	macros Host (excepto {HOST.METADATA}) macros de Host inventory macros definibles por el usuario	

Location	Supported macros	Comments
Monitoring > Latest data	macros Host (excepto {HOST.METADATA}) macros de Host inventory macros de Item (excepto {ITEM.STATE.ERROR}) macros definibles por el usuario	
Data collection > Templates	macros de host: {HOST.DESCRIPTION}, {HOST.HOST}, {HOST.ID}, {HOST.NAME}	
Data collection > Hosts	macros Host (excepto {HOST.METADATA}) macros de Host inventory macros definibles por el usuario	
Other locations		
Working time field	macros definibles por el usuario (solo globales)	Un solo macro debe ocupar todo el campo; no se admiten varios macros ni la mezcla con texto.
Timeouts for item types field (global setting)		
User media *When active* field	macros definibles por el usuario (solo globales)	
Email media type *Username, Password* fields	macros definibles por el usuario (solo globales)	Se recomiendan secret macros.
Scheduled reports *Subject, Message* fields	macros de fecha/hora: {TIME}, {TIMESTAMP}	

Detalles de macro

Esta lista contiene detalles para cada macro incorporada agrupada por área de aplicación:

- Macros de acción
- Macros de alerta
- Macros de fecha/hora
- Macros de descubrimiento
- Macros de evento
  - Macros de actualización de evento
  - Macros de causa/síntoma de evento
  - Macros de evento de recuperación
- Macros de escalado
- Macros de función
- Macros de equipo
  - Macros de equipo de destino
- Macros de grupo de equipos
- Macros de inventario de equipo
- Macros de métrica
- Macros de notificación de descubrimiento de bajo nivel
- Macros de mapa
- Macros de proxy
- Macros de script
- Macros de servicio
- Macros de disparador
- Macros de nombre de usuario

Macros de acción

{ACTION.ID}

El ID numérico de la acción activada.

{ACTION.NAME}

El nombre de la acción activada.

Macros de alerta

{ALERT.MESSAGE}

El valor de *Mensaje por defecto* de la configuración de la acción.

{ALERT.SENDTO}

El valor *Enviar a* de la configuración de medios del usuario.

{ALERT.SUBJECT}

El valor de *Asunto por defecto* de la configuración de la acción.

Macros de fecha y hora

{DATE}

La fecha actual en formato `yyyy.mm.dd.`.

{TIME}

La hora actual en formato `hh:mm:ss`.

{TIMESTAMP}

La hora actual en formato de marca de tiempo UNIX.

Macros de descubrimiento

{DISCOVERY.DEVICE.IPADDRESS}

La dirección IP del dispositivo descubierto.
 Siempre disponible, no depende de que se añada el equipo.

{DISCOVERY.DEVICE.DNS}

El nombre DNS del dispositivo descubierto.
 Siempre disponible, no depende de que el equipo sea añadido.

{DISCOVERY.DEVICE.STATUS}

El estado del dispositivo descubierto (UP/DOWN).

{DISCOVERY.DEVICE.UPTIME}

El tiempo transcurrido desde el último cambio de estado de descubrimiento para un dispositivo en particular.
 Por ejemplo: 1h 29m 01s.
 Para los dispositivos con estado DOWN, este es el período de su tiempo de inactividad.

{DISCOVERY.RULE.NAME}

El nombre de la regla de descubrimiento que detectó la presencia o ausencia de un dispositivo/servicio.

{DISCOVERY.SERVICE.NAME}

El nombre del servicio descubierto.
 Por ejemplo: HTTP.

{DISCOVERY.SERVICE.PORT}

El puerto del servicio descubierto.
 Por ejemplo: 80.

{DISCOVERY.SERVICE.STATUS}

El estado del servicio descubierto (UP/DOWN).

{DISCOVERY.SERVICE.UPTIME}

El tiempo transcurrido desde el último cambio de estado de descubrimiento para un servicio en particular.
 Por ejemplo: 1h 29m 01s.
 Para los servicios con estado DOWN, este es el período de su tiempo de inactividad.

Macros de escalado

{ESC.HISTORY}

El registro de notificaciones enviadas previamente, su paso de escalado y estado (*enviado*, *en progreso* o *fallido*).

Macros de evento

{EVENT.ACK.STATUS}

El estado de reconocimiento del evento (Sí/No).

{EVENT.AGE}

La antigüedad del evento que activó una acción, con precisión de hasta un segundo.
 Útil en mensajes escalados. Por ejemplo: 9m 13s.

{EVENT.DATE}

La fecha del evento que activó una acción en formato `yyyy.mm.dd`.
 Por ejemplo: 2025.04.14.

{EVENT.DURATION}

La duración del evento (diferencia de tiempo entre los eventos de problema y recuperación), con precisión de hasta un segundo.
 Útil en mensajes de recuperación de problemas.

{EVENT.ID}

El ID numérico del evento que activó una acción.

{EVENT.NAME}

El nombre del evento que activó una acción.

{EVENT.NSEVERITY}

El valor numérico de la severidad del evento.
 Valores posibles: *0* - No clasificado, *1* - Información, *2* - Advertencia, *3* - Media, *4* - Alta, *5* - Desastre.

{EVENT.OBJECT}

El valor numérico del objeto del evento.
 Valores posibles: *0* - Disparador, *1* - Equipo descubierto, *2* - Servicio descubierto, *3* - Autorregistro, *4* - Métrica, *5* - Regla de descubrimiento de bajo nivel.

{EVENT.OPDATA}

Los datos operativos del disparador subyacente del problema.

{EVENT.SEVERITY}

El nombre de la severidad del evento.

{EVENT.SOURCE}

El valor numérico de la fuente del evento.
 Valores posibles: *0* - Disparador, *1* - Descubrimiento, *2* - Autorregistro, *3* - Interno, *4* - Servicio.

{EVENT.STATUS}

El valor verbal del evento que activó una acción.

{EVENT.TAGS}

Una lista separada por comas de etiquetas de evento.
Se expande a una cadena vacía si no existen etiquetas.

{EVENT.TAGSJSON}

Una matriz JSON que contiene objetos de etiquetas de evento.
Se expande a una matriz vacía si no existen etiquetas.

{EVENT.TAGS.<tag name>}

El valor de la etiqueta del evento referenciado por el nombre de la etiqueta.
 Un nombre de etiqueta que contenga caracteres no alfanuméricos (incluidos caracteres UTF multibyte no ingleses) debe ir entre comillas dobles. Las comillas y las barras invertidas dentro de un nombre de etiqueta entre comillas deben escaparse con una barra invertida.

{EVENT.TIME}

La hora del evento que activó una acción en formato `hh:mm:ss`.
 Por ejemplo: 12:57:53.

{EVENT.TIMESTAMP}

La marca de tiempo UNIX del evento que activó una acción.

{EVENT.VALUE}

El valor numérico del evento que activó una acción.
 Valores posibles: *1* - problema, *0* - recuperación.

Macros de causa/síntoma de evento

{EVENT.CAUSE.ACK.STATUS}

El estado de reconocimiento del evento causa (Sí/No).
 Se utiliza en el contexto de un evento síntoma.

{EVENT.CAUSE.AGE}

La antigüedad del evento causa, con precisión hasta el segundo.
 Útil en mensajes escalados.
 Se utiliza en el contexto de un evento síntoma.

{EVENT.CAUSE.DATE}

La fecha del evento causa en formato `yyyy.mm.dd`.
 Se utiliza en el contexto de un evento síntoma.

{EVENT.CAUSE.DURATION}

La duración del evento causa (diferencia de tiempo entre los eventos de problema y recuperación), con precisión hasta el segundo.
 Útil en mensajes de recuperación de problemas.
 Se utiliza en el contexto de un evento síntoma.

{EVENT.CAUSE.ID}

El ID numérico del evento causa.
 Se utiliza en el contexto de un evento síntoma.

{EVENT.CAUSE.NAME}

El nombre del evento causa.
 Se utiliza en el contexto de un evento síntoma.

{EVENT.CAUSE.NSEVERITY}

El valor numérico de la severidad del evento causa.
 Valores posibles: *0* - No clasificado, *1* - Información, *2* - Advertencia, *3* - Media, *4* - Alta, *5* - Desastre.
 Se utiliza en el contexto de un evento síntoma.

{EVENT.CAUSE.OBJECT}

El valor numérico del objeto del evento causante.
 Valores posibles: *0* - Disparador, *1* - Equipo descubierto, *2* - Servicio descubierto, *3* - Autorregistro, *4* - Métrica, *5* - Regla de descubrimiento de bajo nivel.
 Se utiliza en el contexto de un evento síntoma.

{EVENT.CAUSE.OPDATA}

Los datos operativos del disparador subyacente del problema causa.
 Se utiliza en el contexto de un evento síntoma.

{EVENT.CAUSE.SEVERITY}

El nombre de la severidad del evento causa.
 Valores posibles: *No clasificado*, *Información*, *Advertencia*, *Media*, *Alta*, *Desastre*.
 Se utiliza en el contexto de un evento síntoma.

{EVENT.CAUSE.SOURCE}

El valor numérico de la fuente del evento causa.
 Valores posibles: *0* - Disparador, *1* - Descubrimiento, *2* - Autorregistro, *3* - Interno.
 Se utiliza en el contexto de un evento síntoma.

{EVENT.CAUSE.STATUS}

El valor verbal del evento causa.
 Se utiliza en el contexto de un evento síntoma.

{EVENT.CAUSE.TAGS}

Una lista separada por comas de etiquetas del evento causa.
Se expande a una cadena vacía si no existen etiquetas.
 Se utiliza en el contexto de un evento síntoma.

{EVENT.CAUSE.TAGSJSON}

Una matriz JSON que contiene los objetos de etiquetas del evento causa.
Se expande a una matriz vacía si no existen etiquetas.
 Se utiliza en el contexto de un evento síntoma.

{EVENT.CAUSE.TAGS.<tag name>}

El valor de la etiqueta del evento de causa referenciado por el nombre de la etiqueta.
 Un nombre de etiqueta que contenga caracteres no alfanuméricos (incluidos caracteres multibyte UTF no ingleses) debe ir entre comillas dobles. Las comillas y las barras invertidas dentro de un nombre de etiqueta entre comillas deben escaparse con una barra invertida.
 Se usa en el contexto de un evento de síntoma.

{EVENT.CAUSE.TIME}

La hora del evento causa en formato `hh:mm:ss`.
 Se utiliza en el contexto de un evento síntoma.

{EVENT.CAUSE.TIMESTAMP}

La marca de tiempo UNIX del evento causa.
 Se utiliza en el contexto de un evento síntoma.

{EVENT.CAUSE.UPDATE.HISTORY}

El registro de actualizaciones de la causa del problema (reconocimientos, etc).
 Se utiliza en el contexto de un evento de síntoma.

{EVENT.CAUSE.VALUE}

El valor numérico del evento causa
 Valores posibles: *1* - Problema, *0* - Recuperación.
 Se utiliza en el contexto de un evento síntoma.

{EVENT.SYMPTOMS}

La lista de eventos de síntoma. Incluye los siguientes detalles: nombre del host, nombre del evento, severidad, antigüedad, etiquetas del servicio y valores.
 Esta macro se usa en el contexto del evento de causa y devuelve información sobre los eventos de síntoma.

Macros de evento de recuperación

{EVENT.RECOVERY.DATE}

La fecha del evento de recuperación en formato `yyyy.mm.dd`.

{EVENT.RECOVERY.ID}

El ID numérico del evento de recuperación.

{EVENT.RECOVERY.NAME}

El nombre del evento de recuperación.

{EVENT.RECOVERY.STATUS}

El valor verbal del evento de recuperación.

{EVENT.RECOVERY.TAGS}

Una lista de etiquetas del evento de recuperación separadas por comas. Se expande a una cadena vacía si no existen etiquetas.

{EVENT.RECOVERY.TAGSJSON}

Una matriz JSON que contiene objetos de etiquetas del evento de recuperación. Se expande a una matriz vacía si no existen etiquetas.

{EVENT.RECOVERY.TIME}

La hora del evento de recuperación en formato `hh:mm:ss`.

{EVENT.RECOVERY.TIMESTAMP}

La marca de tiempo UNIX del evento de recuperación.

{EVENT.RECOVERY.VALUE}

El valor numérico del evento de recuperación.

Macros de actualización de eventos

{EVENT.UPDATE.ACTION}

Nombre legible de la acción o acciones realizadas durante la actualización del problema.
 Se resuelve en los siguientes valores: *acknowledged*, *unacknowledged*, *commented*, *changed severity from (original severity) to (updated severity)* y *closed* (según cuántas acciones se realicen en una sola actualización).

{EVENT.UPDATE.ACTIONJSON}

Un array JSON que contiene detalles de la(s) acción(es) realizada(s) durante la actualización de problemas.
 Valores posibles de las propiedades JSON:
- true (para las propiedades `acknowledge`, `unacknowledge`, `close`, `unsuppress`, `cause` y `symptom`);
- <cadena de mensaje> (para la propiedad `message`);
- timestamp (para las propiedades `suppress_until` y `timestamp`) o 0 (para la propiedad `suppress_until` si está suprimido indefinidamente);
- 0, 1, 2, 3, 4, 5 (para las propiedades de severidad `old` y `new`).

 Por ejemplo: `{"acknowledge":true,"message":"Monthly maintenance.","severity":{"old":2,"new":1},"suppress_until":1730851199,"timestamp":1730822048}`.

{EVENT.UPDATE.DATE}

La fecha de la actualización del evento (reconocimiento, etc) en formato `yyyy.mm.dd`.

{EVENT.UPDATE.HISTORY}

El registro de actualizaciones de problemas (reconocimientos, etc).

{EVENT.UPDATE.MESSAGE}

El mensaje de actualización del problema.

{EVENT.UPDATE.NSEVERITY}

El valor numérico de la nueva severidad del evento establecida durante la operación de actualización del problema.

{EVENT.UPDATE.SEVERITY}

El nombre de la nueva severidad del evento establecida durante la operación de actualización del problema.

{EVENT.UPDATE.STATUS}

El valor numérico del estado de actualización del problema.
 Valores posibles: *0* - El webhook se llamó debido a un evento de problema/recuperación, *1* - Operación de actualización.

{EVENT.UPDATE.TIME}

La hora de la actualización del evento (reconocimiento, etc.) en formato `hh:mm:ss`.

{EVENT.UPDATE.TIMESTAMP}

La marca de tiempo UNIX de la actualización del evento (reconocimiento, etc.)

Macros de función

{FUNCTION.VALUE}

El valor de la función basada en item N-ésima en la expresión del trigger en el momento del evento.
 Solo se cuentan las funciones con `/host/key` como primer parámetro.

 Esta macro puede usarse con un índice numérico, por ejemplo {FUNCTION.VALUE<1-9>} para señalar la primera, segunda, tercera, etc. función en una expresión de trigger. Consulte indexed macros.

{FUNCTION.RECOVERY.VALUE}

El valor de la enésima función basada en item en la expresión de recuperación en el momento del evento.*
 Solo se cuentan las funciones con `/host/keys` como primer parámetro.

 Esta macro puede usarse con un índice numérico, por ejemplo {FUNCTION.RECOVERY.VALUE<1-9>} para señalar la primera, segunda, tercera, etc. función en una expresión de trigger. Consulte macros indexadas.

Macros de equipo

{HOST.CONN}

La dirección IP o el nombre DNS del host, según la configuración del host.
 En scripts globales, los campos IP/DNS de la interfaz y los escenarios web, la macro se resolverá en la interfaz principal de agent. Si no se define una interfaz de agent, se utilizará la interfaz principal de SNMP. Si tampoco se define una interfaz de SNMP, se utilizará la interfaz principal de JMX. Si tampoco se define una interfaz de JMX, se utilizará la interfaz principal de IPMI. Si el host no tiene ninguna interfaz, la macro se resuelve como 'UNKNOWN'.

 Esta macro puede usarse con un índice numérico como {HOST.CONN<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte macros indexadas.

{HOST.DESCRIPTION}

La descripción del host.

 Esta macro puede usarse con un índice numérico como {HOST.DESCRIPTION<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte macros indexadas.

{HOST.DNS}

El nombre DNS del host.
 En scripts globales, los campos IP/DNS de la interfaz y los escenarios web, la macro se resolverá en la interfaz agent principal. Si no se define una interfaz agent, se usará la interfaz SNMP principal. Si tampoco se define una interfaz SNMP, se usará la interfaz JMX principal. Si tampoco se define una interfaz JMX, se usará la interfaz IPMI principal. Si el host no tiene ninguna interfaz, la macro se resuelve como 'UNKNOWN'.

 Esta macro puede usarse con un índice numérico como {HOST.DNS<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte macros indexadas.

{HOST.HOST}

El nombre técnico del host.
 Las funciones de macro no son compatibles con esta macro si se usa como marcador de posición en el primer parámetro de una función de historial, por ejemplo, `last(/{HOST.HOST}/{ITEM.KEY})`.

 Esta macro puede usarse con un índice numérico como {HOST.HOST<1-9>} para señalar el primer, segundo, tercer, etc. host en una expresión de trigger. Consulte las macros indexadas.

{HOST.ID}

El ID del host.

 Esta macro puede usarse con un índice numérico como {HOST.ID<1-9>} para señalar el primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{HOST.IP}

La dirección IP del host.
 En scripts globales, los campos IP/DNS de la interfaz y los escenarios web, la macro se resolverá en la interfaz principal de agent. Si no se define una interfaz de agent, se utilizará la interfaz principal SNMP. Si tampoco se define una interfaz SNMP, se utilizará la interfaz principal JMX. Si no se define una interfaz JMX, se utilizará la interfaz principal IPMI. Si el host no tiene ninguna interfaz, la macro se resuelve como 'UNKNOWN'.

 Esta macro puede usarse con un índice numérico como {HOST.IP<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte macros indexadas.

{HOST.METADATA}

Los metadatos del equipo.

 Se utiliza solo para el autoregistro de agentes activos.

{HOST.NAME}

El nombre visible del host.

 Esta macro puede usarse con un índice numérico como {HOST.NAME<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte macros indexadas.

{HOST.PORT}

El puerto del host (agent).
 En scripts globales, los campos IP/DNS de la interfaz y los escenarios web, la macro se resolverá a la interfaz principal de agent. Si no se define una interfaz de agent, se utilizará la interfaz principal SNMP. Si tampoco se define una interfaz SNMP, se utilizará la interfaz principal JMX. Si no se define una interfaz JMX, se utilizará la interfaz principal IPMI. Si el host no tiene ninguna interfaz, la macro se resuelve como 'UNKNOWN'.

 Esta macro puede usarse con un índice numérico como {HOST.PORT<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte macros indexadas.

Macros de equipo de destino

{HOST.TARGET.CONN}

La dirección IP o el nombre DNS del equipo de destino.

{HOST.TARGET.DNS}

El nombre DNS del equipo de destino.

{HOST.TARGET.HOST}

El nombre técnico del equipo de destino.

{HOST.TARGET.IP}

La dirección IP del equipo de destino.

{HOST.TARGET.NAME}

El nombre visible del equipo de destino.

Macros de grupo de equipos

{HOSTGROUP.ID}

El ID del grupo de equipos.

Macros de inventario de equipo

{INVENTORY.ALIAS}

El campo *Alias* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.ALIAS<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte macros indexadas.

{INVENTORY.ASSET.TAG}

El campo *Asset tag* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.ASSET.TAG<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.CHASSIS}

El campo *Chassis* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.CHASSIS<1-9>} para señalar el primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.CONTACT}

El campo *Contact* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.CONTACT<1-9>} para señalar el primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.CONTRACT.NUMBER}

El campo *Contract number* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.CONTRACT.NUMBER<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.DEPLOYMENT.STATUS}

El campo *Deployment status* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.DEPLOYMENT.STATUS<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.HARDWARE}

El campo *Hardware* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.HARDWARE<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.HARDWARE.FULL}

El campo *Hardware (detalles completos)* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.HARDWARE.FULL<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte macros indexadas.

{INVENTORY.HOST.NETMASK}

El campo *Máscara de subred del host* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.HOST.NETMASK<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte macros indexadas.

{INVENTORY.HOST.NETWORKS}

El campo *Host networks* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.HOST.NETWORKS<1-9>} para señalar el primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.HOST.ROUTER}

El campo *Host router* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.HOST.ROUTER<1-9>} para señalar el primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.HW.ARCH}

El campo *Hardware architecture* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.HW.ARCH<1-9>} para señalar al primer, segundo, tercero, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.HW.DATE.DECOMM}

El campo *Date hardware decommissioned* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.HW.DATE.DECOMM<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.HW.DATE.EXPIRY}

El campo *Date hardware maintenance expires* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.HW.DATE.EXPIRY<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.HW.DATE.INSTALL}

El campo *Date hardware installed* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.HW.DATE.INSTALL<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.HW.DATE.PURCHASE}

El campo *Date hardware purchased* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.HW.DATE.PURCHASE<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.INSTALLER.NAME}

El campo *Installer name* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.INSTALLER.NAME<1-9>} para señalar el primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.LOCATION}

El campo *Location* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.LOCATION<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.LOCATION.LAT}

El campo *Latitud de la ubicación* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.LOCATION.LAT<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte macros indexadas.

{INVENTORY.LOCATION.LON}

El campo *Longitud de la ubicación* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.LOCATION.LON<1-9>}, para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte macros indexadas.

{INVENTORY.MACADDRESS.A}

El campo *MAC address A* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.MACADDRESS.A<1-9>} para señalar el primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.MACADDRESS.B}

El campo *MAC address B* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.MACADDRESS.B<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.MODEL}

El campo *Model* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.MODEL<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.NAME}

El campo *Name* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.NAME<1-9>}, para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.NOTES}

El campo *Notes* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.NOTES<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.OOB.IP}

El campo *dirección IP OOB* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.OOB.IP<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte macros indexadas.

{INVENTORY.OOB.NETMASK}

El campo *máscara de subred OOB* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.OOB.NETMASK<1-9>}, para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte macros indexadas.

{INVENTORY.OOB.ROUTER}

El campo *OOB router* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.OOB.ROUTER<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.OS}

El campo *OS* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.OS<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte macros indexadas.

{INVENTORY.OS.FULL}

El campo *OS (detalles completos)* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.OS.FULL<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte macros indexadas.

{INVENTORY.OS.SHORT}

El campo *SO (corto)* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.OS.SHORT<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte macros indexadas.

{INVENTORY.POC.PRIMARY.CELL}

El campo *Primary POC cell* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.POC.PRIMARY.CELL<1-9>}, para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.POC.PRIMARY.EMAIL}

El campo *Correo electrónico del POC principal* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.POC.PRIMARY.EMAIL<1-9>}, para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte macros indexadas.

{INVENTORY.POC.PRIMARY.NAME}

El campo *Nombre del POC principal* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.POC.PRIMARY.NAME<1-9>}, para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.POC.PRIMARY.NOTES}

El campo *Notas del POC principal* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.POC.PRIMARY.NOTES<1-9>} para señalar el primer, segundo, tercer, etc. host en una expresión de trigger. Consulte macros indexadas.

{INVENTORY.POC.PRIMARY.PHONE.A}

El campo *Primary POC phone A* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.POC.PRIMARY.PHONE.A<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.POC.PRIMARY.PHONE.B}

El campo *Primary POC phone B* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.POC.PRIMARY.PHONE.B<1-9>} para señalar el primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.POC.PRIMARY.SCREEN}

El campo *Primary POC screen name* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.POC.PRIMARY.SCREEN<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.POC.SECONDARY.CELL}

El campo *Secondary POC cell* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.POC.SECONDARY.CELL<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.POC.SECONDARY.EMAIL}

El campo *Correo electrónico secundario del POC* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.POC.SECONDARY.EMAIL<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte macros indexadas.

{INVENTORY.POC.SECONDARY.NAME}

El campo *Nombre del POC secundario* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.POC.SECONDARY.NAME<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte macros indexadas.

{INVENTORY.POC.SECONDARY.NOTES}

El campo *Notas del POC secundario* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.POC.SECONDARY.NOTES<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte macros indexadas.

{INVENTORY.POC.SECONDARY.PHONE.A}

El campo *Secondary POC phone A* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.POC.SECONDARY.PHONE.A<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.POC.SECONDARY.PHONE.B}

El campo *Secondary POC phone B* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.POC.SECONDARY.PHONE.B<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.POC.SECONDARY.SCREEN}

El campo *Secondary POC screen name* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.POC.SECONDARY.SCREEN<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.SERIALNO.A}

El campo *Número de serie A* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.SERIALNO.A<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte macros indexadas.

{INVENTORY.SERIALNO.B}

El campo *Número de serie B* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.SERIALNO.B<1-9>} para señalar el primer, segundo, tercer, etc. host en una expresión de trigger. Consulte macros indexadas.

{INVENTORY.SITE.ADDRESS.A}

El campo *Site address A* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.SITE.ADDRESS.A<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.SITE.ADDRESS.B}

El campo *Site address B* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.SITE.ADDRESS.B<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.SITE.ADDRESS.C}

El campo *Site address C* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.SITE.ADDRESS.C<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.SITE.CITY}

El campo *Site city* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.SITE.CITY<1-9>} para señalar el primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.SITE.COUNTRY}

El campo *Site country* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.SITE.COUNTRY<1-9>} para señalar el primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.SITE.NOTES}

El campo *Site notes* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.SITE.NOTES<1-9>}, para señalar el primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.SITE.RACK}

El campo *Site rack location* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.SITE.RACK<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.SITE.STATE}

El campo *Estado/provincia del sitio* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.SITE.STATE<1-9>}, para señalar el primer, segundo, tercer, etc. host en una expresión de trigger. Consulte macros indexadas.

{INVENTORY.SITE.ZIP}

El campo *Código postal de la ubicación* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.SITE.ZIP<1-9>} para señalar el primer, segundo, tercer, etc. host en una expresión de trigger. Consulte macros indexadas.

{INVENTORY.SOFTWARE}

El campo *Software* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.SOFTWARE<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.SOFTWARE.APP.A}

El campo *Software application A* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.SOFTWARE.APP.A<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte macros indexadas.

{INVENTORY.SOFTWARE.APP.B}

El campo *Software application B* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.SOFTWARE.APP.B<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.SOFTWARE.APP.C}

El campo *Software application C* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.SOFTWARE.APP.C<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.SOFTWARE.APP.D}

El campo *Software application D* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.SOFTWARE.APP.D<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.SOFTWARE.APP.E}

El campo *Software application E* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.SOFTWARE.APP.E<1-9>} para señalar al primer, segundo, tercero, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.SOFTWARE.FULL}

El campo *Software (detalles completos)* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.SOFTWARE.FULL<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte macros indexadas.

{INVENTORY.TAG}

El campo *Tag* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.TAG<1-9>} para señalar el primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.TYPE}

El campo *Type* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.TYPE<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{INVENTORY.TYPE.FULL}

El campo *Tipo (detalles completos)* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.TYPE.FULL<1-9>}, para señalar el primer, segundo, tercer, etc. host en una expresión de trigger. Consulte macros indexadas.

{INVENTORY.URL.A}

El campo *URL A* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.URL.A<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte macros indexadas.

{INVENTORY.URL.B}

El campo *URL B* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.URL.B<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte macros indexadas.

{INVENTORY.URL.C}

El campo *URL C* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.URL.C<1-9>}, para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte macros indexadas.

{INVENTORY.VENDOR}

El campo *Vendor* en el inventario del host.

 Esta macro puede usarse con un índice numérico, por ejemplo {INVENTORY.VENDOR<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte macros indexadas.

Macros de métricas

{ITEM.DESCRIPTION}

La descripción del N.º item en la expresión del trigger que provocó una notificación.

 Esta macro puede usarse con un índice numérico como {ITEM.DESCRIPTION<1-9>} para señalar el primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{ITEM.DESCRIPTION.ORIG}

La descripción original (con macros sin resolver) del N-ésimo item en la expresión del trigger que provocó una notificación.

 Este macro puede usarse con un índice numérico como {ITEM.DESCRIPTION.ORIG<1-9>} para señalar el primer, segundo, tercer, etc. host en una expresión del trigger. Consulte indexed macros.

{ITEM.ID}

El ID numérico del N.º item en la expresión del trigger que provocó una notificación.

 Esta macro puede usarse con un índice numérico como {ITEM.ID<1-9>} para señalar el primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{ITEM.KEY}

La clave del enésimo item en la expresión del trigger que provocó una notificación.
 Las funciones de macro no son compatibles con esta macro si se usa como marcador de posición en el primer parámetro de una función de historial, por ejemplo, `last(/{HOST.HOST}/{ITEM.KEY})`.

 Esta macro puede usarse con un índice numérico como {ITEM.KEY<1-9>} para señalar al primer, segundo, tercer, etc. host en una expresión de trigger. Consulte las macros indexadas.

{ITEM.KEY.ORIG}

La clave original (con macros sin resolver) del N.º item en la expresión del trigger que provocó una notificación.

 Esta macro puede usarse con un índice numérico como {ITEM.KEY.ORIG<1-9>} para señalar el primer, segundo, tercer, etc. host en una expresión de trigger. Consulte indexed macros.

{ITEM.LASTVALUE}

El valor más reciente del Nth item en la expresión de trigger que provocó una notificación.
 Se resolverá como *UNKNOWN* en el frontend si el último valor del historial se recopiló hace más tiempo que el período *Max history display period* (configurado en la sección de menú *Administration > General*).
 Cuando se usa en el nombre del problema, la macro no se resolverá al valor más reciente del item al ver eventos de problema; en su lugar, conservará el valor del item del momento en que ocurrió el problema.
Cuando se usa en notificaciones, en algunos casos la macro podría no resolverse al valor más reciente del item en el momento en que se activó el trigger. Por ejemplo, si un item recibe rápidamente dos valores, ”A” y ”B”, y el trigger se activa para ”A”, las notificaciones podrían mostrar ”B” como el valor más reciente debido a un ligero retraso en el procesamiento: el valor más reciente del item cambió entre el momento en que se activó el trigger y cuando se creó la notificación. Para evitar esto, puede usar la macro {ITEM.VALUE}, que se resuelve al valor en el momento en que se activa el trigger, garantizando que se use el valor correcto en la notificación.
 Es un alias de `last(/{HOST.HOST}/{ITEM.KEY})`.
 El valor resuelto para items de texto/log se trunca a 20 caracteres por el frontend en las siguientes ubicaciones:
- Operational data;
- Trigger description;
- Trigger URLs;
- Trigger URL labels;
- Description of the item value widget.
Para resolverlo con el valor completo, puede usar macro functions, ya que el server no trunca ningún valor. Por ejemplo: `{{ITEM.LASTVALUE}.regsub("(.*)", \1)}`.

 Esta macro puede usarse con un índice numérico, por ejemplo {ITEM.LASTVALUE<1-9>} para señalar al primer, segundo, tercer, etc. item en una expresión de trigger. Consulte indexed macros.

{ITEM.LASTVALUE.AGE}

El tiempo transcurrido entre la última recopilación del valor del item y la evaluación de la macro.
 Se muestra en un formato legible para humanos (por ejemplo, 1m 45s).

 Esta macro puede usarse con un índice numérico, por ejemplo {ITEM.LASTVALUE.AGE<1-9>} para señalar el primer, segundo, tercer, etc. item en una expresión de trigger. Consulte indexed macros.

{ITEM.LASTVALUE.DATE}

La fecha en que se recopiló el último valor del item.
 Se muestra en formato `YYYYMMDD`.
 La zona horaria del server se usa en los nombres de trigger (en la lista *Monitoring* > *Problems*), los nombres de eventos y los nombres y valores de etiquetas.
 En todos los demás casos se usa la zona horaria del usuario.

 Esta macro puede usarse con un índice numérico, por ejemplo, {ITEM.LASTVALUE.DATE<1-9>} para señalar el primer, segundo, tercer, etc. item en una expresión de trigger. Consulte indexed macros.

{ITEM.LASTVALUE.TIME}

La hora en que se recopiló el último valor del item.
 Se muestra en formato `HHMMSS`.
 La zona horaria del server se usa en los nombres de trigger (en la lista *Monitoring* > *Problems*), los nombres de eventos y los nombres y valores de etiquetas. En todos los demás casos se usa la zona horaria del usuario.

 Esta macro puede usarse con un índice numérico, por ejemplo, {ITEM.LASTVALUE.TIME<1-9>} para señalar el primer, segundo, tercer, etc. item en una expresión de trigger. Consulte macros indexadas.

{ITEM.LASTVALUE.TIMESTAMP}

La marca de tiempo UNIX en la que se recopiló el último valor del item.

 Esta macro puede usarse con un índice numérico, por ejemplo {ITEM.LASTVALUE.TIMESTAMP<1-9>} para señalar el primer, segundo, tercer, etc. item en una expresión de trigger. Consulte indexed macros.

{ITEM.LOG.AGE}

La antigüedad del evento de registro.
Con precisión de hasta un segundo.

 Esta macro puede usarse con un índice numérico, por ejemplo {ITEM.LOG.AGE<1-9>} para señalar el primer, segundo, tercer, etc. item en una expresión de trigger. Consulte indexed macros.

{ITEM.LOG.DATE}

La fecha en que la entrada de registro se escribió en el log.

 Esta macro puede usarse con un índice numérico, por ejemplo {ITEM.LOG.DATE<1-9>} para señalar el primer, segundo, tercero, etc. item en una expresión de trigger. Consulte indexed macros.

{ITEM.LOG.EVENTID}

El ID del evento en el registro de eventos.
 Solo para la supervisión del registro de eventos de Windows.

 Esta macro puede usarse con un índice numérico, por ejemplo {ITEM.LOG.EVENTID<1-9>} para señalar el primer, segundo, tercer, etc. item en una expresión de trigger. Consulte indexed macros.

{ITEM.LOG.NSEVERITY}

La gravedad numérica del evento en el registro de eventos.
 Solo para la supervisión del registro de eventos de Windows.

 Esta macro puede usarse con un índice numérico, por ejemplo {ITEM.LOG.NSEVERITY<1-9>} para señalar el primer, segundo, tercer, etc. item en una expresión de trigger. Consulte macros indexadas.

{ITEM.LOG.SEVERITY}

La severidad verbal del evento en el registro de eventos.
 Solo para la monitorización del registro de eventos de Windows.

 Esta macro puede usarse con un índice numérico, por ejemplo {ITEM.LOG.SEVERITY<1-9>} para señalar el primer, segundo, tercer, etc. item en una expresión de trigger. Consulte macros indexadas.

{ITEM.LOG.SOURCE}

La fuente del evento en el registro de eventos.
 Solo para la supervisión del registro de eventos de Windows.

 Esta macro puede usarse con un índice numérico, por ejemplo {ITEM.LOG.SOURCE<1-9>} para señalar el primer, segundo, tercer, etc. item en una expresión de trigger. Consulte macros indexadas.

{ITEM.LOG.TIME}

La hora en que la entrada de registro se escribió en el registro.

 Esta macro puede usarse con un índice numérico, por ejemplo {ITEM.LOG.TIME<1-9>} para señalar el primer, segundo, tercer, etc. item en una expresión de trigger. Consulte indexed macros.

{ITEM.LOG.TIMESTAMP}

La marca de tiempo UNIX en la que la entrada de registro se escribió en el registro.

 Esta macro puede usarse con un índice numérico, por ejemplo {ITEM.LOG.TIMESTAMP<1-9>} para señalar el primer, segundo, tercer, etc. item en una expresión de trigger. Consulte macros indexadas.

{ITEM.NAME}

El nombre del item con todas las macros resueltas.

 Esta macro puede usarse con un índice numérico, por ejemplo {ITEM.NAME<1-9>} para señalar el primer, segundo, tercer, etc. item en una expresión de trigger. Consulte indexed macros.

{ITEM.NAME.ORIG}

El nombre original (con macros sin resolver) del item.

 Esta macro puede usarse con un índice numérico, por ejemplo {ITEM.NAME.ORIG<1-9>} para señalar el primer, segundo, tercer, etc. item en una expresión de trigger. Consulte indexed macros.

{ITEM.STATE}

El estado más reciente del N-ésimo item en la expresión del trigger que provocó una notificación.
 Valores posibles: *Not supported*, *Normal*.

 Esta macro puede usarse con un índice numérico, por ejemplo {ITEM.STATE<1-9>} para señalar el primer, segundo, tercer, etc. item en una expresión del trigger. Consulte indexed macros.

{ITEM.STATE.ERROR}

El mensaje de error con detalles de por qué una métrica se volvió no soportada.
 Si una métrica pasa al estado no soportado y luego inmediatamente vuelve a estar soportada, el campo de error puede estar vacío.

{ITEM.VALUE}

Se resuelve como uno de los siguientes valores:

- El valor histórico (en el momento del evento) del N-ésimo item en la expresión del trigger, si se usa en el contexto de un cambio de estado del trigger, por ejemplo, al mostrar eventos o enviar notificaciones;
- El valor más reciente del N-ésimo item en la expresión del trigger, si se usa sin el contexto de un cambio de estado del trigger, por ejemplo, al mostrar una lista de triggers en una ventana emergente de selección. En este caso funciona igual que {ITEM.LASTVALUE}.

En el primer caso, se resolverá como *UNKNOWN* si el valor histórico ya se ha eliminado o nunca se ha almacenado.
En el segundo caso, y solo en el frontend, se resolverá como *UNKNOWN* si el valor histórico más reciente se recopiló hace más tiempo que el período *Max history display period* (configurado en la sección de menú *Administration > General*).
El valor resuelto para items de texto/log se trunca a 20 caracteres por el frontend en las siguientes ubicaciones:
- Operational data;
- Trigger description;
- Trigger URLs;
- Trigger URL labels;
- Description of the item value widget.
Para obtener el valor completo, puede usar macro functions, ya que el server no trunca ningún valor. Por ejemplo: `{{ITEM.VALUE}.regsub("(.*)", \1)}`.

Esta macro puede usarse con un índice numérico, por ejemplo {ITEM.VALUE<1-9>} para apuntar al primer, segundo, tercer, etc. item en una expresión de trigger. Véase indexed macros.

{ITEM.VALUE.AGE}

El tiempo transcurrido entre la recopilación del valor del item y la evaluación de la macro.
 Se muestra en un formato legible para humanos (por ejemplo, 1m 45s).

 Esta macro puede usarse con un índice numérico, por ejemplo {ITEM.VALUE.AGE<1-9>} para señalar el primer, segundo, tercer, etc. item en una expresión de trigger. Consulte macros indexadas.

{ITEM.VALUE.DATE}

La fecha en que se recopiló el valor del item.
 Se muestra en formato `YYYYMMDD`.
 La zona horaria del server se usa en los nombres de trigger (en la lista *Monitoring* > *Problems*), los nombres de eventos y los nombres y valores de etiquetas. En todos los demás casos se usa la zona horaria del usuario.

Esta macro puede usarse con un índice numérico, por ejemplo, {ITEM.VALUE.DATE<1-9>} para señalar el primer, segundo, tercer, etc. item en una expresión de trigger. Consulte macros indexadas.

{ITEM.VALUE.TIME}

La hora en que se recopiló el valor del item.
 Se muestra en formato `HHMMSS`.
 La zona horaria del server se usa en los nombres de trigger (en la lista *Monitoring* > *Problems*), los nombres de eventos y los nombres y valores de las etiquetas. En todos los demás casos se usa la zona horaria del usuario.

Esta macro puede usarse con un índice numérico, por ejemplo, {ITEM.VALUE.TIME<1-9>} para señalar el primer, segundo, tercer, etc. item en una expresión de trigger. Consulte macros indexadas.

{ITEM.VALUE.TIMESTAMP}

La marca de tiempo UNIX en la que se recopiló el valor del item.

 Esta macro puede usarse con un índice numérico, por ejemplo {ITEM.VALUE.TIMESTAMP<1-9>} para señalar el primer, segundo, tercer, etc. item en una expresión de trigger. Consulte indexed macros.

{ITEM.VALUETYPE}

El tipo de valor del N-ésimo item en la expresión del trigger que provocó una notificación.
 Valores posibles: *0* - Numérico flotante, *1* - Carácter, *2* - Registro, *3* - Numérico sin signo, *4* - Texto.

 Esta macro puede usarse con un índice numérico, por ejemplo {ITEM.VALUETYPE<1-9>} para señalar el primer, segundo, tercer, etc. item en una expresión del trigger. Consulte macros indexadas.

Macros de notificación de descubrimiento de bajo nivel

{LLDRULE.DESCRIPTION}

La descripción de la regla de descubrimiento de bajo nivel que provocó una notificación.

{LLDRULE.DESCRIPTION.ORIG}

La descripción original (con macros sin resolver) de la regla de descubrimiento de bajo nivel que provocó una notificación.

{LLDRULE.ID}

El ID numérico de la regla de descubrimiento de bajo nivel que provocó una notificación.

{LLDRULE.KEY}

La clave de la regla de descubrimiento de bajo nivel que provocó una notificación.

{LLDRULE.KEYORIG}

La clave original (con macros sin resolver) de la regla de descubrimiento de bajo nivel que provocó una notificación.

{LLDRULE.NAME}

El nombre de la regla de descubrimiento de bajo nivel que provocó una notificación.

{LLDRULE.NAMEORIG}

El nombre original (con macros sin resolver) de la regla de descubrimiento de bajo nivel que provocó una notificación.

{LLDRULE.STATE}

El estado más reciente de la regla de descubrimiento de bajo nivel.
 Valores posibles: *Not supported*, *Normal*.

 Esta macro puede usarse con un índice numérico, por ejemplo {LLDRULE.STATE<1-9>} para señalar el primer, segundo, tercer, etc. item en una expresión de trigger. Consulte indexed macros.

{LLDRULE.STATE.ERROR}

El mensaje de error con detalles de por qué la regla de descubrimiento de bajo nivel se volvió no soportada.
 Si una regla de descubrimiento de bajo nivel pasa al estado no soportado y luego inmediatamente vuelve a estar soportada, el campo de error puede estar vacío.

Macros de mapa

{MAP.ID}

El ID del mapa de red.

{MAP.NAME}

El nombre del mapa de red.

Macros de proxy

{PROXY.DESCRIPTION}

La descripción del proxy.
 Se resuelve como una de las siguientes opciones:

- Proxy del enésimo item en la expresión del trigger (en notificaciones basadas en trigger). Puede usar macros indexadas aquí;
- Proxy que ejecutó el descubrimiento (en notificaciones de descubrimiento). Use {PROXY.DESCRIPTION} aquí, sin indexación;
- Proxy al que se registró un agent activo (en notificaciones de autoregistro). Use {PROXY.DESCRIPTION} aquí, sin indexación.

Esta macro puede usarse con un índice numérico, por ejemplo {PROXY.DESCRIPTION<1-9>} para apuntar al primer, segundo, tercer, etc. item en una expresión de trigger. Consulte las macros indexadas.

{PROXY.NAME}

El nombre del proxy.
 Se resuelve como cualquiera de los siguientes:

- Proxy del enésimo item en la expresión del trigger (en notificaciones basadas en trigger). Puede usar macros indexadas aquí;
- Proxy que ejecutó el descubrimiento (en notificaciones de descubrimiento). Use {PROXY.NAME} aquí, sin indexación;
- Proxy al que se registró un agent activo (en notificaciones de autoregistro). Use {PROXY.NAME} aquí, sin indexación.

Esta macro puede usarse con un índice numérico, por ejemplo {PROXY.NAME<1-9>} para señalar el primer, segundo, tercer, etc. item en una expresión de trigger. Consulte las macros indexadas.

Macros de script

{MANUALINPUT}

El valor de entrada manual especificado por el usuario en el momento de la ejecución del script.

Macros de servicio

{SERVICE.DESCRIPTION}

La descripción del servicio con macros resueltas.

{SERVICE.ID}

El ID numérico del servicio que activó una acción.

{SERVICE.NAME}

El nombre del servicio con macros resueltas.

{SERVICE.ROOTCAUSE}

La lista de eventos de problema de disparadores que causaron que un servicio fallara, ordenados por gravedad y nombre de equipo.
 Incluye los siguientes detalles: nombre de equipo, nombre del evento, gravedad, antigüedad, etiquetas de servicio y valores.

{SERVICE.TAGS}

Una lista separada por comas de etiquetas de eventos del servicio.
 Las etiquetas de eventos del servicio se pueden definir en la sección de etiquetas de configuración del servicio. Se expande a una cadena vacía si no existen etiquetas.

{SERVICE.TAGSJSON}

Un array JSON que contiene objetos de etiquetas de eventos de servicio.
 Las etiquetas de eventos de servicio se pueden definir en la sección de etiquetas de configuración del servicio. Se expande a un array vacío si no existen etiquetas.

{SERVICE.TAGS.<tag name>}

El valor de la etiqueta del evento de servicio referenciado por el nombre de la etiqueta.
 Las etiquetas del evento de servicio se pueden definir en la sección de etiquetas de configuración del servicio.
 Un nombre de etiqueta que contenga caracteres no alfanuméricos (incluidos caracteres multibyte UTF no ingleses) debe ir entre comillas dobles. Las comillas y las barras invertidas dentro de un nombre de etiqueta entre comillas deben escaparse con una barra invertida.

Macros de disparador

{TRIGGER.DESCRIPTION}

La descripción del trigger.
 Todas las macros soportadas en la descripción del trigger serán expandidas si `{TRIGGER.DESCRIPTION}` se utiliza en el texto de la notificación.

{TRIGGER.EVENTS.ACK}

El número de eventos reconocidos para un elemento de mapa en mapas, o para el disparador que generó el evento actual en notificaciones.

{TRIGGER.EVENTS.PROBLEM.ACK}

El número de eventos de problema reconocidos para todos los disparadores sin tener en cuenta su estado.

{TRIGGER.EVENTS.PROBLEM.UNACK}

El número de eventos de problema no reconocidos para todos los disparadores, sin tener en cuenta su estado.

{TRIGGER.EVENTS.UNACK}

El número de eventos no reconocidos para un elemento de mapa en mapas, o para el disparador que generó el evento actual en notificaciones.

{TRIGGER.EXPRESSION}

La expresión de la condición.

{TRIGGER.EXPRESSION.EXPLAIN}

Una expresión de trigger parcialmente evaluada.
 Las funciones basadas en item se evalúan y se reemplazan por los resultados en el momento de la generación del evento, mientras que todas las demás funciones se muestran tal como están escritas en la expresión. Puede utilizarse para depurar expresiones de trigger.

{TRIGGER.EXPRESSION.RECOVERY}

La expresión de recuperación del disparador si la opción *Generación de evento OK* en la configuración del disparador está establecida en ”Expresión de recuperación”; de lo contrario, se devuelve una cadena vacía.

{TRIGGER.EXPRESSION.RECOVERY.EXPLAIN}

Una expresión de recuperación de trigger evaluada parcialmente.
 Las funciones basadas en item se evalúan y se reemplazan por los resultados en el momento de la generación del evento, mientras que todas las demás funciones se muestran tal como están escritas en la expresión. Puede utilizarse para depurar expresiones de recuperación de trigger.

{TRIGGER.HOSTGROUP.NAME}

Una lista ordenada (por consulta SQL), separada por comas y espacios, de los grupos de equipos en los que se define el disparador.

{TRIGGER.ID}

El ID numérico del disparador que activó esta acción.

{TRIGGER.NAME}

El nombre del disparador con las macros resueltas.
 Tenga en cuenta que `{EVENT.NAME}` puede usarse en acciones para mostrar el nombre del evento/problema disparado con las macros resueltas.

{TRIGGER.NAME.ORIG}

El nombre original de la condición (con macros sin resolver).

{TRIGGER.NSEVERITY}

La gravedad numérica del disparador.
 Valores posibles: *0* - No clasificado, *1* - Información, *2* - Advertencia, *3* - Media, *4* - Alta, *5* - Desastre.

{TRIGGER.PROBLEM.EVENTS.PROBLEM.ACK}

El número de eventos de problema reconocidos para disparadores en estado de problema.

{TRIGGER.PROBLEM.EVENTS.PROBLEM.UNACK}

El número de eventos de problema no reconocidos para disparadores en estado de problema.

{TRIGGER.SEVERITY}

El nombre de la severidad del disparador.
 Puede definirse en *Administración > General > Opciones de visualización de disparadores*.

{TRIGGER.STATE}

El último estado de la expresión del trigger.
 Valores posibles: *Unknown*, *Normal*.

{TRIGGER.STATE.ERROR}

El mensaje de error con detalles sobre por qué un disparador se volvió no soportado.
 Si un disparador pasa al estado no soportado y luego inmediatamente vuelve a estar soportado, el campo de error puede estar vacío.

{TRIGGER.STATUS}

El valor del trigger en el momento de la ejecución del paso de la operación.
 Valores posibles: *Problem*, *OK*.

{TRIGGER.TEMPLATE.NAME}

Una lista ordenada (por consulta SQL), separada por comas y espacios, de las plantillas en las que se define el disparador, o *UNKNOWN* si el disparador se define en un equipo.

{TRIGGER.URL}

La URL del disparador.

{TRIGGER.URL.NAME}

La etiqueta para la URL del disparador.

{TRIGGER.VALUE}

El valor numérico actual del trigger.
 Valores posibles: *0* - el trigger está en estado OK, *1* - el trigger está en estado Problem.

{TRIGGERS.ACK}

El número de disparadores reconocidos para un elemento del mapa, sin tener en cuenta el estado del disparador.
 Un disparador se considera reconocido si todos sus eventos de problema están reconocidos.

{TRIGGERS.PROBLEM.ACK}

El número de disparadores de problema reconocidos para un elemento del mapa.
 Un disparador se considera reconocido si todos sus eventos de problema están reconocidos.

{TRIGGERS.PROBLEM.UNACK}

El número de disparadores de problema no reconocidos para un elemento del mapa.
 Un disparador se considera no reconocido si al menos uno de sus eventos de problema no está reconocido.

{TRIGGERS.UNACK}

El número de disparadores no reconocidos para un elemento del mapa, sin tener en cuenta el estado del disparador.
 Un disparador se considera no reconocido si al menos uno de sus eventos de problema no está reconocido.

Macros de nombre de usuario

{USER.FULLNAME}

El nombre, apellido y nombre de usuario del usuario que añadió el reconocimiento del evento o inició el script.

{USER.NAME}

El nombre del usuario que inició el script.

{USER.SURNAME}

El apellido del usuario que inició el script.

{USER.USERNAME}

El nombre de usuario del usuario que inició el script.

Macros indexadas

La sintaxis de macro indexada **{MACRO<1-9>}** solo puede hacer referencia al N-ésimo item, función o host en el campo *Expression* de un trigger:

- **{HOST.IP1}, {HOST.IP2}, {HOST.IP3}** se resuelven como las direcciones IP del primer, segundo y tercer host en la expresión del trigger (si están presentes).
- **{ITEM.VALUE1}, {ITEM.VALUE2}, {ITEM.VALUE3}** se resuelven como los valores del primer, segundo y tercer item en la expresión del trigger en el momento del evento (si están presentes).
- **{FUNCTION.VALUE1}, {FUNCTION.VALUE2}, {FUNCTION.VALUE3}** se resuelven como los valores de la primera, segunda y tercera función basada en item en el momento del evento (si están presentes).

En el contexto de los triggers, las macros indexadas siempre hacen referencia al campo *Expression* de la configuración del trigger, no a la *Recovery expression*. Por ejemplo, en un evento de recuperación, **{ITEM.VALUE2}** se resolverá como el valor del segundo item de la expresión de problema en el momento de la recuperación.

La macro **{HOST.HOST<1-9>}** también es compatible dentro de la macro de expresión `{?func(/host/key,param)}` en los **nombres de gráfico**. Por ejemplo, `{?func(/{HOST.HOST2}/key,param)}` en un nombre de gráfico se resolverá como el host del segundo item en el gráfico.

Warning:
Las macros indexadas no se resolverán en ningún otro contexto fuera de los casos mencionados aquí. Para otros contextos, use macros sin índice (**{HOST.HOST}, {HOST.IP}**, etc.).

Macros de usuario compatibles con location

Descripción general

Esta sección contiene una lista de ubicaciones donde se admiten macros definibles por el usuario.

Note:
Solo se admiten macros de usuario a nivel global para *Acciones*, *Descubrimiento de red*, *Proxies* y todas las ubicaciones listadas en la sección *Otras ubicaciones* de esta página. En las ubicaciones mencionadas, las macros a nivel de equipo y a nivel de plantilla no se resolverán.

Note:
Para personalizar los valores de las macros (por ejemplo, acortar o extraer subcadenas específicas), puede utilizar funciones de macros.

Acciones

En acciones, las macros de usuario pueden utilizarse en los siguientes campos:

Ubicación		Múltiples macros/mezcla con texto[1]
Notificaciones y comandos basados en disparadores		sí
Notificaciones internas basadas en disparadores		sí
Notificaciones de actualización de problema		sí
Notificaciones y comandos basados en servicios		sí
Notificaciones de actualización de servicio		sí
Condición de periodo de tiempo		no
Operaciones		
	Duración por defecto del paso de operación	no
	Duración del paso	no

Equipos/prototipos de equipo

En una configuración de equipo y prototipo de equipo, se pueden utilizar macros de usuario en los siguientes campos:

Ubicación		Múltiples macros/mezcla con texto[1]
IP/DNS de la interfaz		Solo DNS
Puerto de la interfaz		no
SNMP v1, v2		
	Comunidad SNMP	sí
SNMP v3		
	Nombre de contexto	sí
	Nombre de seguridad	sí
	Frase de autenticación	sí
	Frase de privacidad	sí
IPMI		
	Nombre de usuario	sí
	Contraseña	sí
Etiquetas[2]		
	Nombres de etiqueta	sí
	Valores de etiqueta	sí

Items / prototipos de item

En la configuración de un item o un prototipo de item, las macros de usuario se pueden utilizar en los siguientes campos:

Ubicación	¿Varias macros/mezcla con texto[1]
Nombre del item	sí
Parámetros de la clave del item	sí[4]
Intervalo de actualización	no
Intervalos personalizados	no
Tiempo de espera (disponible para tipos de item compatibles)	no

Ubicación	¿Varias macros/mezcla con texto[1]
Almacenar hasta (para el historial y las tendencias)	no
Descripción	sí
Item calculado/agregado	
Fórmula (constantes de expresión y parámetros de función; parámetros de la clave del item; (*solo para items agregados*) condiciones de filtro (nombre del grupo de hosts y nombre de la etiqueta))	sí
Monitor de base de datos	
Nombre de usuario	sí
Contraseña	sí
Consulta SQL	sí
HTTP agent	
URL[3]	sí
Campos de consulta	sí
Cuerpo de la solicitud	sí
Encabezados (nombres y valores)	sí
Códigos de estado requeridos	sí
Proxy HTTP	sí
Nombre de usuario de autenticación HTTP	sí
Contraseña de autenticación HTTP	sí
Archivo de certificado SSl	sí
Archivo de clave SSl	sí
Contraseña de la clave SSl	sí
Hosts permitidos	sí
JMX agent	
Endpoint JMX	sí
Item de script	
Nombres y valores de los parámetros	sí
Item de navegador	
Nombres y valores de los parámetros	sí
Item de consulta de telemetría	

Ubicación	¿Varias macros/mezcla con texto[1]
Desplazamiento temporal	no
Límite de retrospectiva	no
Granularidad	no
SNMP agent	
OID SNMP	sí
SSH agent	
Nombre de usuario	sí
Archivo de clave pública	sí
Archivo de clave privada	sí
Contraseña	sí
Script	sí
TELNET agent	
Nombre de usuario	sí
Contraseña	sí
Script	sí
Zabbix trapper	
Hosts permitidos	sí
Etiquetas[2]	
Nombres de las etiquetas	sí
Valores de las etiquetas	sí
Pasos de preprocesamiento	
Parámetros (incluidos los scripts personalizados)	sí
Parámetros personalizados de gestión de errores (campos *Establecer valor en* y *Establecer error en*)	sí

Descubrimiento de bajo nivel

En una regla de descubrimiento de bajo nivel, las macros de usuario pueden utilizarse en los siguientes campos:

Ubicación	Múltiples macros/mezcla con texto[1]
Parámetros de clave	sí
Intervalo de actualización	no
Intervalo personalizado	no

Ubicación	Múltiples macros/mezcla con texto[1]
Tiempo de espera (disponible para los tipos de item compatibles)	no
Eliminar recursos perdidos	no
Deshabilitar recursos perdidos	no
Descripción	sí
SNMP agent	
OID SNMP	sí
SSH agent	
Nombre de usuario	sí
Archivo de clave pública	sí
Archivo de clave privada	sí
Contraseña	sí
Script	sí
TELNET agent	
Nombre de usuario	sí
Contraseña	sí
Script	sí
Zabbix trapper	
Hosts permitidos	sí
Monitor de base de datos	
Nombre de usuario	sí
Contraseña	sí
Consulta SQL	sí
JMX agent	
Endpoint JMX	sí
HTTP agent	

Ubicación	Múltiples macros/mezcla con texto[1]
URL[3]	sí
Campos de consulta	sí
Cuerpo de la solicitud	sí
Cabeceras (nombres y valores)	sí
Códigos de estado requeridos	sí
Nombre de usuario para autenticación HTTP	sí
Contraseña para autenticación HTTP	sí
Filtros	
Expresión regular	sí
Sobrescrituras	
Filtros: expresión regular	sí
Operaciones: intervalo de actualización (para prototipos de item)	no
Operaciones: período de almacenamiento del historial (para prototipos de item)	no
Operaciones: período de almacenamiento de tendencias (para prototipos de item)	no

Descubrimiento de red

En una regla de descubrimiento de red, se pueden usar macros de usuario en los siguientes campos:

Ubicación		Múltiples macros/mezcla con texto[1]
Intervalo de actualización		no
SNMP v1, v2		
	Comunidad SNMP	sí
	OID SNMP	sí
SNMP v3		
	Nombre de contexto	sí
	Nombre de seguridad	sí
	Frase de autenticación	sí
	Frase de privacidad	sí
	OID SNMP	sí

Proxies

En una configuración de proxy, se pueden usar macros de usuario en los siguientes campos:

Location	Multiple macros/mix with text[1]
Address for active agents > *Port* (if the proxy belongs to a group)	no
Interface address and port (for passive proxy)	no
Timeouts for item types	no

Grupos de proxy

En una configuración de grupo de proxy, se pueden usar macros de usuario en los siguientes campos:

Location	Multiple macros/mix with text[1]
Failover period	no
Minimum number of proxies	no

Plantillas

En una configuración de plantilla, se pueden utilizar macros de usuario en los siguientes campos:

Ubicación		Múltiples macros/mezcla con texto[1]
Etiquetas[2]		
	Nombres de etiquetas	sí
	Valores de etiquetas	sí

Disparadores

En una disparador configuración, las macros de usuario pueden utilizarse en los siguientes campos:

Ubicación	Múltiples macros/mezcla con texto[1]
Nombre	sí
Datos operativos	sí
Expresión (solo en constantes y parámetros de función; las macros secretas no son compatibles)	sí
Etiqueta para coincidencia	sí
Nombre de entrada de menú	sí
URL de entrada de menú[3]	sí
Descripción	sí
Etiquetas[2]	
Nombres de etiquetas	sí
Valores de etiquetas	sí

Escenario web

En una configuración de escenario web, las macros de usuario pueden utilizarse en los siguientes campos:

Ubicación		Múltiples macros/mezcla con texto[1]
Nombre		sí
Intervalo de actualización		no
Agente		sí
Proxy HTTP		sí
Variables (solo valores)		sí
Cabeceras (nombres y valores)		sí
Pasos		
	Nombre	sí
	URL[3]	sí
	Variables (solo valores)	sí
	Cabeceras (nombres y valores)	sí
	Tiempo de espera	no
	Cadena requerida	sí
	Códigos de estado requeridos	no
Autenticación		
	Usuario	sí
	Contraseña	sí
	Certificado SSL	sí
	Archivo de clave SSL	sí
	Contraseña de clave SSL	sí
Etiquetas[2]		
	Nombres de etiqueta	sí
	Valores de etiqueta	sí

Otras ubicaciones

Además de las ubicaciones enumeradas aquí, se pueden usar macros de usuario en los siguientes campos:

Ubicación	Múltiples macros/mezcla con texto[1]
Scripts globales (URL, script, SSH, Telnet, IPMI), incluido el texto de confirmación	yes
Webhooks	
Script de JavaScript	no
Nombre del parámetro del script de JavaScript	no
Valor del parámetro del script de JavaScript	yes
Dashboards	
Columna de tipo de dato *Text* en el widget del dashboard *Top hosts*	yes
Parámetro *Description* en el widget del dashboard *Item value* y *Gauge*	yes
Parámetro *Text* de la etiqueta primaria/secundaria en el widget del dashboard *Honeycomb*	yes
Parámetro *URL*[3] en el widget del dashboard *URL*	yes

Ubicación	Múltiples macros/mezcla con texto[1]
Monitoring > *Problems*, *Hosts*, *Latest data*	
Columna *Custom text*	yes
Data collection > *Templates*, *Hosts*	
Columna *Custom text*	yes
Alerts > *Media types* > *Message templates*	
Asunto	yes
Mensaje	yes
Alerts > *Media types* > *Script*	
Parámetros del script	yes
Alerts > *Media types* > *Media type*	
Campos *Username* y *Password* para el tipo de medio *Email* (cuando *Authentication* está configurado como ”Username and password”; se recomiendan secret macros)	yes
Users > *Users* > *Media*	
Cuando está activo	yes

Ubicación	Múltiples macros/mezcla con texto[1]
Administration > General > GUI	
Tiempo de trabajo	no
Administration > General > Timeouts	
Tiempos de espera para tipos de item	no
Administration > General > Connectors	
URL	yes
Nombre de usuario	yes
Contraseña	yes
Token Bearer	yes
Tiempo de espera	no
Proxy HTTP	yes
Archivo de certificado SSL	yes
Archivo de clave SSL	yes
Contraseña de la clave SSL	yes

Para obtener una lista completa de todas las macros compatibles en Zabbix, consulte macros compatibles.

Notas al pie

[1] Si no se admiten múltiples macros en un campo o macros mezcladas con texto para la ubicación, una sola macro debe ocupar todo el campo.

[2] Las macros usadas en nombres y valores de etiquetas solo se resuelven durante el proceso de generación de eventos.

[3] Las URL que contienen una macro secreta no funcionarán, ya que la macro en ellas se resolverá como "******".

[4] Las macros no se admiten en la clave del item de los items Zabbix trapper.

Compatibilidad de versiones

Agentes compatibles

Para ser compatible con Zabbix 8.0, Zabbix agent no debe ser anterior a la versión 1.6 ni posterior a la 8.0.

Es posible que deba revisar la configuración de los agentes antiguos, ya que algunos parámetros han cambiado, por ejemplo, los parámetros relacionados con el registro en las versiones anteriores a la 3.0.

Para aprovechar al máximo las funciones más recientes, las métricas, el rendimiento mejorado y el menor uso de memoria, utilice el agent compatible más reciente.

Notas para Windows XP

- En Windows XP de 32 bits, no utilice Zabbix agents posteriores a la versión 6.0.x;
- En Windows XP/Server 2003, no utilice templates de agents posteriores a Zabbix 4.0.x. Los templates más recientes utilizan contadores de rendimiento en inglés, que solo son compatibles desde Windows Vista/Server 2008.

Agentes 2 compatibles

Los agentes 2 de Zabbix más antiguos desde la versión 4.4 en adelante son compatibles con Zabbix 8.0; el agente 2 de Zabbix no debe ser más reciente que la versión 8.0.

Tenga en cuenta que al utilizar las versiones 4.4 y 5.0 del agente 2 de Zabbix, se utiliza el intervalo predeterminado de 10 minutos para actualizar las métricas no soportadas.

Para aprovechar al máximo la funcionalidad más reciente, las métricas, el rendimiento mejorado y el menor uso de memoria, utilice el agente 2 más reciente compatible.

Proxies de Zabbix compatibles

Para una compatibilidad total con Zabbix 8.0, los proxies deben coincidir con la versión principal del servidor. Solo los proxies Zabbix 8.0.x son totalmente compatibles con un servidor Zabbix 8.0.x.

Los proxies desactualizados son parcialmente compatibles: aún pueden recopilar datos y ejecutar scripts, pero no pueden recibir actualizaciones de configuración, como nuevas métricas.

En relación con el servidor Zabbix, los proxies pueden ser:

- *Actuales* (el proxy y el servidor tienen la misma versión principal);
- *Desactualizados* (la versión del proxy es anterior a la del servidor, pero es parcialmente compatible);
- *No compatibles* (la versión del proxy es anterior a la versión LTS anterior del servidor *o* la versión del proxy es más reciente que la versión principal del servidor).

Ejemplos:

Versión del servidor	*Proxy actual*	*Proxy desactualizado*	*Proxy no compatible*
6.4	6.4	6.0, 6.2	Anterior a 6.0; posterior a 6.4
7.0	7.0	6.0, 6.2, 6.4	Anterior a 6.0; posterior a 7.0
7.2	7.2	7.0	Anterior a 7.0; posterior a 7.2
7.4	7.4	7.0	Anterior a 7.0; posterior a 7.4
8.0	8.0	7.0, 7.2, 7.4	Anterior a 7.0; posterior a 8.0

Funcionalidad compatible con proxies:

Versión de proxy	Actualización de datos	Actualización de configuración	Tareas
Actual	Sí	Sí	Sí
Obsoleta	Sí	No	Comandos remotos (por ejemplo, scripts de shell); Comprobaciones inmediatas del valor de item (es decir, *Ejecutar ahora*); Nota: No se admiten las pruebas de preprocesamiento con un valor real.
No compatible	No	No	No

Se registran advertencias sobre el uso de versiones incompatibles de demonios de Zabbix.

Archivos XML compatibles

Los archivos XML no anteriores a la versión 1.8 son compatibles para la importación en Zabbix 8.0.

Attention:
En el formato de exportación XML, las dependencias de disparadores se almacenan solo por nombre. Si hay varios disparadores con el mismo nombre (por ejemplo, con diferentes severidades y expresiones) que tienen una dependencia definida entre ellos, no es posible importarlos. Dichas dependencias deben eliminarse manualmente del archivo XML y volver a agregarse después de la importación.

Sintaxis del período de tiempo

Descripción general

Para establecer un período de tiempo, se debe utilizar el siguiente formato:

`d-d,hh:mm-hh:mm`

donde los símbolos significan lo siguiente:

Símbolo	Descripción
d	Día de la semana: 1 - Lunes, 2 - Martes, ... , 7 - Domingo
hh	Horas: 00-24
mm	Minutos: 00-59

Puede especificar más de un período de tiempo utilizando un punto y coma (;) como separador:

`d-d,hh:mm-hh:mm;d-d,hh:mm-hh:mm...`

Dejar el período de tiempo vacío equivale a 1-7,00:00-24:00, que es el valor por defecto.

Attention:
El límite superior de un período de tiempo no está incluido. Por lo tanto, si especifica 09:00-18:00, el último segundo incluido en el período de tiempo es 17:59:59.

Ejemplos

Horario laboral. Lunes - Viernes de 9:00 a 18:00:

`1-5,09:00-18:00`

Horario laboral más fin de semana. Lunes - Viernes de 9:00 a 18:00 y Sábado, Domingo de 10:00 a 16:00:

`1-5,09:00-18:00;6-7,10:00-16:00`

Símbolos de unidades

Descripción general

Trabajar con valores grandes como 86400, 104857600 o 1000000 puede ser complicado y propenso a errores. Para simplificar la configuración y mejorar la legibilidad, Zabbix admite símbolos de unidad (sufijos) que funcionan como multiplicadores de valor.

Expresiones de disparador sin sufijos:

```
last(/equipo/system.uptime)<86400
avg(/equipo/system.cpu.load,600s)<10
last(/equipo/vm.memory.size[available])<20971520
```

Expresiones de disparador con sufijos:

```
last(/equipo/system.uptime)<1d
avg(/equipo/system.cpu.load,10m)<10
last(/equipo/vm.memory.size[available])<20M
```

Los sufijos también pueden simplificar la configuración de otras entidades, como métricas, widgets, etc., y ayudar a mostrar los valores de las métricas en un formato legible para humanos.

Note:
Para ver si un campo de configuración de una entidad admite sufijos, consulte siempre la página correspondiente a la entidad que se está configurando.

Sufijos de tiempo

El frontend de Zabbix admite los siguientes sufijos de tiempo en la configuración de entidades:
 s - segundos *(cuando se usa, funciona igual que el valor bruto)*;
 m - minutos;
 h - horas;
 d - días;
 w - semanas;
 M - meses *(solo para funciones de tendencia)*;
 y - años *(solo para funciones de tendencia)*.

Los sufijos de tiempo solo se admiten con números enteros. Por ejemplo, se admite 1h, pero no se admite 1,5h ni 1.5h; use 90m en su lugar.

Sufijos de tamaño de memoria

Zabbix admite los siguientes sufijos de tamaño de memoria:
 K - kilobytes
 M - megabytes
 G - gigabytes
 T - terabytes

Sufijos de valor de item

Los sufijos también se pueden usar para mostrar valores numéricos de item en un formato legible para humanos.

Para habilitar esto, use uno de los siguientes sufijos en el campo *Units* al configurar un item:
 B - bytes;
 Bps - bytes por segundo;
 s - segundos, mostrados usando hasta tres unidades de tiempo no nulas más grandes;
 uptime - tiempo transcurrido en hh:mm:ss o N días, hh:mm:ss;
 unixtime - marca de tiempo Unix, con formato yyyy.mm.dd hh:mm:ss.

Se aplican las siguientes reglas adicionales sobre cómo se interpretan y muestran estos sufijos:

- Para B y Bps, Zabbix usa conversión en base 2 (1K = 1024B), siguiendo el estándar JEDEC.
- Para otras unidades (como Hz, W, etc.), Zabbix usa conversión en base 10 (1K = 1000).
- Para s (segundos):
  - El formato incluye yyy mmm ddd hhh mmm sss ms; solo se muestran hasta tres unidades de tiempo no nulas más grandes (por ejemplo, 1M 10d 4h).
  - Si una unidad es cero y está entre dos unidades no nulas, se omite (por ejemplo, 10d 56m en lugar de 10d 0h 56m).

Cuando se usan *Units*, los siguientes sufijos multiplicadores se aplican automáticamente a los valores de item:
 K, **M**, **G**, **T** - kilo, mega, giga, tera;
 P, **E**, **Z**, **Y** - peta, exa, zetta, yotta *(estos se aplican solo en frontend)*.

Para evitar la conversión de unidades, use el prefijo ! (por ejemplo, !B o !s).

Los siguientes ejemplos muestran cómo se convierten los valores de item recibidos según las unidades especificadas:

```
1 B → 1 B
1024 B → 1 KB
1536 B → 1.5 KB
881764 B → 881.76 KB
881764 !B → 881764 B

0.0000155 s → 0.016ms
3470400 s → 1M 10d 4h
2606400 s → 1M 4h
2592000 s → 1M
2592001 s → 1M
2592001 !s → 2592001 s

17764 uptime → 04:56:04
86400 uptime → 1 day, 00:00:00
881764 uptime → 10 days, 04:56:04
32417764 uptime → 375 days, 04:56:04
32417764 !uptime → 32417764 uptime

881764 unixtime → 1970-01-11 04:56:04 AM

17764 Hz → 17.76 KHz
86400 Hz → 86.4 KHz
881764 Hz → 881.76 KHz
32417764 Hz → 32.42 MHz

0 ! → 0
0 !! → 0 !
```

Note:
Antes de Zabbix 4.0, existía una lista de exclusión de unidades codificada de forma fija que consistía en `ms`, `rpm`, `RPM`, `%`. Esta lista de exclusión ha quedado obsoleta; por lo tanto, la forma correcta de evitar la conversión de estas unidades es `!ms`, `!rpm`, `!RPM`, `!%`.

Ejecución de comandos

Zabbix utiliza una funcionalidad común para comprobaciones externas, parámetros de usuario, items system.run, scripts de alerta personalizados, comandos remotos y scripts globales.

Pasos de ejecución

> **Note:**
> Por defecto, todos los scripts en Zabbix se ejecutan utilizando el intérprete de comandos *sh*, y no es posible modificar el intérprete por defecto. Para utilizar un intérprete diferente, puede emplear una solución alternativa: cree un archivo de script e invoque ese script durante la ejecución del comando.

El comando/script se ejecuta de manera similar tanto en plataformas Unix como en Windows:

1. Zabbix (el proceso padre) crea un pipe para la comunicación
2. Zabbix establece el pipe como la salida para el proceso hijo que se va a crear
3. Zabbix crea el proceso hijo (ejecuta el comando/script)
4. Se crea un nuevo grupo de procesos (en Unix) o un trabajo (en Windows) para el proceso hijo
5. Zabbix lee del pipe hasta que ocurre un timeout o nadie esté escribiendo en el otro extremo (TODOS los manejadores/descriptores de archivo han sido cerrados). Tenga en cuenta que el proceso hijo puede crear más procesos y salir antes de que estos salgan o cierren el manejador/descriptor de archivo.
6. Si no se ha alcanzado el timeout, Zabbix espera hasta que el proceso hijo inicial salga o ocurra el timeout
7. Si el proceso hijo inicial salió y no se ha alcanzado el timeout, Zabbix verifica el código de salida del proceso hijo inicial y lo compara con 0 (un valor distinto de cero se considera como fallo de ejecución, solo para scripts de alerta personalizados, comandos remotos y scripts de usuario ejecutados en el servidor Zabbix y el proxy Zabbix)
8. En este punto se asume que todo ha finalizado y todo el árbol de procesos (es decir, el grupo de procesos o el trabajo) se termina

> **Attention:**
> Zabbix asume que un comando/script ha finalizado el procesamiento cuando el proceso hijo inicial ha salido Y ningún otro proceso sigue manteniendo abierto el manejador/descriptor de salida. Cuando el procesamiento ha finalizado, TODOS los procesos creados se terminan.

Todas las comillas dobles y las barras invertidas en el comando se escapan con barras invertidas y el comando se encierra entre comillas dobles.

Verificación del código de salida

Los códigos de salida se verifican con las siguientes condiciones:

- Solo para scripts de alerta personalizados, comandos remotos y scripts de usuario ejecutados en Zabbix server y Zabbix proxy.
- Cualquier código de salida distinto de 0 se considera un fallo de ejecución.
- El contenido de la salida de error estándar y de la salida estándar para las ejecuciones fallidas se recopila y está disponible en frontend (donde se muestra el resultado de la ejecución).
- Se puede crear una entrada adicional en el registro para comandos remotos ejecutados en Zabbix agent/proxy habilitando el parámetro LogRemoteCommands en agent/proxy.

Posibles mensajes de frontend y entradas de registro para comandos/scripts fallidos:

- El contenido de la salida de error estándar y de la salida estándar para las ejecuciones fallidas (si lo hay).
- "Process exited with code: N." (para salida vacía y código de salida no igual a 0).
- "Process killed by signal: N." (para un proceso terminado por una señal, solo en Linux).
- "Process terminated unexpectedly." (para un proceso terminado por razones desconocidas).

Véase también

- Comprobaciones externas
- Parámetros de usuario
- items de system.run
- Scripts de alerta personalizados
- Comandos remotos
- Scripts globales

Ejemplos de escape

Descripción general

Esta página proporciona ejemplos de cómo utilizar el escape correcto al usar expresiones regulares en varios contextos.

Note:
Al utilizar el constructor de expresiones de disparadores, el escape correcto en las expresiones regulares se añade automáticamente.

Ejemplos

Macro de usuario con contexto

Expresión regular: `\.+\"[a-z]+<br>` Macro de usuario con contexto: `{$MACRO:regex:"\.+\\"[a-z]+"}`

Nota:

- las barras invertidas no se escapan;
- las comillas se escapan.

Función de macro dentro del parámetro de clave de métrica

Expresión regular: `.+:(\d+)$<br>` Clave de métrica: `net.tcp.service[tcp,,"{{$ENDPOINT}.regsub(\".+:(\d+)$\",\1)}"]`

Nota:

- la expresión regular dentro de la función de macro `regsub` está entre comillas dobles (porque contiene paréntesis de cierre);
- las comillas alrededor de la expresión regular se escapan (porque todo el tercer parámetro de la métrica está entre comillas dobles);
- el tercer parámetro de la clave de métrica está entre comillas dobles porque contiene una coma.

Función de macro LLD

Expresión regular: `\.+\"([a-z]+)<br>` Macro LLD: `{{#MACRO}.iregsub("\.+\\"([a-z]+)", \1)}`

Nota:

- las barras invertidas no se escapan;
- las comillas se escapan.

Función de macro LLD dentro del contexto de macro de usuario

Expresión regular: `\.+\"([a-z]+)<br>` Macro LLD: `{{#MACRO}.iregsub("\.+\\"([a-z]+)", \1)}<br>` Macro de usuario con contexto: `{$MACRO:"{{#MACRO}.iregsub(\"\.+\\\"[a-z]+\", \1)}"}`

Nota:

- el escape de barra invertida para LLD no cambia;
- al insertar la macro LLD en el contexto de macro de usuario, necesitamos ponerla en una cadena:

1. Se agregan comillas alrededor de la expresión de macro;
2. Las comillas se escapan; en total, se introducen 3 nuevas barras invertidas.

Parámetro de cadena de la función (cualquiera)

`concat` se usa como ejemplo.

Contenido de la cadena: `\.+\"[a-z]+<br>` Expresión: `concat("abc", "\\.\\\"[a-z]+")`

Nota:

- Los parámetros de cadena requieren escape tanto para barras invertidas como para comillas.

Función de macro LLD dentro del parámetro de cadena de la función

Expresión regular: `\.+\"([a-z]+)<br>` Macro LLD: `{{#MACRO}.iregsub("\.+\\"([a-z]+)", \1)}<br>` Expresión: `concat("abc, "{{#MACRO}.iregsub(\"\\.+\\\\\"([a-z]+)\", \\1)}")`

Nota:

- Los parámetros de cadena requieren escape tanto para barras invertidas como para comillas;
- Se agrega otra capa de escape, porque la macro se resolverá solo después de que la cadena se desentrecomille;

Macro de usuario con contexto dentro del parámetro de cadena de la función

Expresión regular: `\.+\"[a-z]+<br>` Macro de usuario con contexto: `{$MACRO:regex:"\.+\\"[a-z]+"}<br>` Expresión: `concat("abc, "{$MACRO:regex:\"\\.+\\\\\"[a-z]+\"}")`

Nota:

- Igual que en el ejemplo anterior, se necesita una capa adicional de escape;
- Las barras invertidas y las comillas se escapan solo para el escape de nivel superior (por ser un parámetro de cadena).

Función de macro LLD dentro del contexto de macro de usuario dentro de la función

Expresión regular: `\.+\"([a-z])+<br>` Macro LLD: `{{#MACRO}.iregsub("\.+\\"([a-z]+)", \1)}<br>` Macro de usuario con contexto: `{$MACRO:"{{#MACRO}.iregsub(\"\.+\\\"([a-z]+)\", \1)}"}<br>` Expresión: `concat("abc, "{$MACRO:\"{{#MACRO}.iregsub(\\\"\.+\\\\\\\"([a-z]+)\\\", \\1)}\"}")`

Observe las tres capas de escape:

1. Para la función de macro LLD, sin escape de barras invertidas;
2. Para la macro de usuario con contexto, sin escape de barras invertidas;
3. Para el parámetro de cadena de una función, con escape de barras invertidas.

Macro de usuario con contexto solo dentro de una cadena

Expresión regular: `\.+\"[a-z]+<br>` Macro de usuario con contexto: `{$MACRO:regex:"\.+\\"[a-z]+"}<br>` Dentro de la cadena de alguna expresión, por ejemplo: `func(arg1, arg2, arg3)="{$MACRO:regex:\"\\.+\\\\\"[a-z]+\"}"`

Nota:

- Las cadenas también requieren escape de barras invertidas;
- Las cadenas también requieren escape de comillas;
- De nuevo un caso con 2 niveles de escape:

1. Escape para el contexto de macro de usuario sin escape de barras invertidas;
2. Escape por ser una cadena con escape de barras invertidas.

Qué es Zabbix

Zabbix fue creado por Alexei Vladishev y actualmente Zabbix SIA lo desarrolla y respalda activamente.

Zabbix es una solución de monitorización distribuida de código abierto de nivel empresarial.

Zabbix es un software que monitoriza numerosos parámetros de una red, así como el estado y la integridad de servidores, máquinas virtuales, aplicaciones, servicios, bases de datos, sitios web, la nube y mucho más. Zabbix utiliza un mecanismo flexible de notificaciones que permite a los usuarios configurar alertas basadas en correo electrónico para prácticamente cualquier evento. Esto permite reaccionar rápidamente ante los problemas del servidor. Zabbix ofrece excelentes funciones de generación de informes y visualización de datos basadas en los datos almacenados. Esto hace que Zabbix sea ideal para la planificación de capacidad.

Zabbix admite tanto el sondeo como la recepción de traps. Todos los informes y estadísticas de Zabbix, así como los parámetros de configuración, se consultan a través de un frontend basado en la web. Un frontend basado en la web garantiza que el estado de su red y la salud de sus servidores puedan evaluarse desde cualquier ubicación. Si se configura correctamente, Zabbix puede desempeñar un papel importante en la monitorización de la infraestructura de TI. Esto es igualmente válido para organizaciones pequeñas con unos pocos servidores y para grandes empresas con multitud de servidores.

Zabbix está disponible sin coste alguno. Zabbix está escrito y se distribuye bajo la licencia AGPL-3.0. Esto significa que su código fuente se distribuye libremente y está disponible para el público en general.

El soporte comercial está disponible y lo proporcionan Zabbix Company y sus socios en todo el mundo.

Obtenga más información sobre las funciones de Zabbix.

Usuarios de Zabbix Muchas organizaciones de distintos tamaños en todo el mundo confían en Zabbix como su plataforma principal de monitoreo.

Arquitectura Zabbix consta de varios componentes de software principales. Sus responsabilidades se describen a continuación.

Server

Zabbix server es el componente central al que los agent reportan información y estadísticas sobre disponibilidad e integridad. El server es el repositorio central en el que se almacenan todos los datos de configuración, estadísticos y operativos.

Almacenamiento de la base de datos

Toda la información de configuración, así como los datos recopilados por Zabbix, se almacenan en una base de datos.

Interfaz web

Para un acceso fácil a Zabbix desde cualquier lugar y desde cualquier plataforma, se proporciona la interfaz basada en web. La interfaz forma parte de Zabbix server y, por lo general, aunque no necesariamente, se ejecuta en la misma máquina física que la que ejecuta el server.

Proxy

Zabbix proxy puede recopilar datos de rendimiento y disponibilidad en nombre de Zabbix server. Un proxy es una parte opcional de la implementación de Zabbix; sin embargo, puede ser muy beneficioso para distribuir la carga de un único Zabbix server.

Agent

Los agentes de Zabbix se implementan en los objetivos de monitoreo para supervisar de forma activa los recursos y aplicaciones locales, y enviar los datos recopilados al server de Zabbix. Desde Zabbix 4.4, hay dos tipos de agentes disponibles: el Zabbix agent (ligero, compatible con muchas plataformas, escrito en C) y el Zabbix agent 2 (muy flexible, fácilmente ampliable con plugins, escrito en Go).

Flujo de datos Además, es importante dar un paso atrás y observar el flujo de datos general dentro de Zabbix. Para crear un item que recopile datos, primero debe crear un host. Pasando al otro extremo del espectro de Zabbix, primero debe tener un item para crear un trigger. Debe tener un trigger para crear una acción. Así, si desea recibir una alerta de que la carga de CPU es demasiado alta en *Server X*, primero debe crear una entrada de host para *Server X*, seguida de un item para supervisar su CPU, luego un trigger que se active si la CPU es demasiado alta, seguido de una acción que le envíe un correo electrónico. Aunque eso pueda parecer muchos pasos, con el uso de templates en realidad no lo es. Sin embargo, debido a este diseño, es posible crear una configuración muy flexible.

Características Zabbix es una solución de monitorización de red altamente integrada, que ofrece una multiplicidad de funciones en un solo paquete.

Recopilación de datos

- comprobaciones de disponibilidad y rendimiento
- compatibilidad con SNMP (tanto trapping como polling), IPMI, JMX, monitorización de VMware
- comprobaciones personalizadas
- recopilación de los datos deseados a intervalos personalizados
- realizada por server/proxy y por agents

Definiciones flexibles de umbrales

- puede definir umbrales de problema muy flexibles, llamados triggers, que hacen referencia a valores de la base de datos backend

Alertas altamente configurables

- el envío de notificaciones se puede personalizar según el programa de escalado, el destinatario y el tipo de medio
- las notificaciones pueden hacerse significativas y útiles mediante variables macro
- las acciones automáticas incluyen comandos remotos

Generación de gráficos en tiempo real

- los items monitorizados se representan inmediatamente en gráficos mediante la funcionalidad de gráficos integrada

Capacidades de monitorización web

- Zabbix puede seguir una ruta de clics de ratón simulados en un sitio web y comprobar su funcionalidad y tiempo de respuesta

Amplias opciones de visualización

- capacidad para crear gráficos personalizados que pueden combinar múltiples items en una sola vista
- mapas de red
- presentaciones en forma de diapositivas en una vista general estilo dashboard
- informes
- vista de alto nivel (de negocio) de los recursos monitorizados

Almacenamiento de datos históricos

- datos almacenados en una base de datos
- historial configurable
- procedimiento de housekeeping integrado

Configuración sencilla

- añadir dispositivos monitorizados como hosts
- los hosts se incorporan para su monitorización una vez que están en la base de datos
- aplicar templates a los dispositivos monitorizados

Uso de templates

- agrupación de comprobaciones en templates

- los templates pueden heredar otros templates

Descubrimiento de red

- descubrimiento automático de dispositivos de red
- autoregistro de agent
- descubrimiento de sistemas de archivos, interfaces de red y OID de SNMP

Frontend web rápido

- un frontend basado en web en PHP
- accesible desde cualquier lugar
- puede navegar haciendo clic
- registro de auditoría

API de Zabbix

- la API de Zabbix proporciona una interfaz programable para Zabbix para manipulaciones masivas, integración con software de terceros y otros fines.

Sistema de permisos

- autenticación segura de usuarios
- ciertos usuarios pueden limitarse a determinadas vistas

agent completo y fácilmente extensible

- desplegado en los objetivos de monitorización
- puede desplegarse tanto en Linux como en Windows

Daemons binarios

- escritos en C, para ofrecer rendimiento y un pequeño consumo de memoria
- fácilmente portables

Preparado para entornos complejos

- la monitorización remota se simplifica mediante el uso de un Zabbix proxy

Glosario En esta sección puede conocer el significado de algunos términos utilizados habitualmente en Zabbix.

host

- cualquier dispositivo físico o virtual, aplicación, servicio u otra colección relacionada lógicamente de parámetros supervisados.

host group

- una agrupación lógica de hosts. Los grupos de hosts se utilizan al asignar permisos de acceso a los hosts para diferentes grupos de usuarios.

item

- un dato concreto que desea recibir de un host, una métrica de datos.

value preprocessing

- una transformación del valor de la métrica recibida antes de guardarlo en la base de datos.

trigger

- una expresión lógica que define un umbral de problema y se utiliza para "evaluar" los datos recibidos en los items.

Cuando los datos recibidos superan el umbral, los triggers pasan del estado *Ok* al estado *Problem*. Cuando los datos recibidos están por debajo del umbral, los triggers permanecen o vuelven al estado *Ok*.

template

- un conjunto de entidades (items, triggers, gráficos, reglas de descubrimiento de bajo nivel y escenarios web) listo para aplicarse a uno o varios hosts.

La función de los templates es acelerar la implementación de tareas de supervisión en un host y facilitar la aplicación de cambios masivos a las tareas de supervisión. Los templates se vinculan directamente a hosts individuales.

template group

- una agrupación lógica de templates. Los grupos de templates se utilizan al asignar permisos de acceso a los templates para diferentes grupos de usuarios.

event

- una ocurrencia individual de algo que requiere atención, como un cambio de estado de un trigger o la realización de un descubrimiento o un registro automático de un agent.

event tag

- un marcador predefinido para el evento. Puede utilizarse en la correlación de eventos, la granularidad de permisos, etc.

event correlation

- un método para correlacionar problemas con su resolución de forma flexible y precisa.

Por ejemplo, puede definir que un problema notificado por un trigger se resuelva mediante otro trigger, que incluso puede utilizar un método diferente de recopilación de datos.

problem

- un trigger que se encuentra en estado "Problem".

problem update

- opciones de gestión de problemas proporcionadas por Zabbix, como añadir un comentario, confirmar, cambiar la gravedad o cerrar manualmente.

action

- un medio predefinido para reaccionar ante un evento.

Una acción consta de operaciones (por ejemplo, enviar una notificación) y condiciones (*cuándo* se ejecuta la operación).

escalation

- un escenario personalizado para ejecutar operaciones dentro de una acción; una secuencia de envío de notificaciones o ejecución de comandos remotos.

macro

- un marcador de posición variable que se resuelve en un valor específico según el contexto.

media

- un medio para entregar notificaciones; un canal de entrega.

notification

- un mensaje sobre algún evento enviado a un usuario mediante el canal de media elegido.

remote command

- un comando predefinido que se ejecuta automáticamente en un host supervisado cuando se cumple alguna condición.

web scenario

- una o varias solicitudes HTTP para comprobar la disponibilidad de un sitio web.

frontend

- la interfaz web proporcionada con Zabbix.

dashboard

- una sección personalizable de la interfaz web que muestra resúmenes y visualizaciones de información importante en unidades visuales denominadas widgets.

widget

- una unidad visual que muestra información de un tipo y origen determinados (un resumen, un mapa, un gráfico, el reloj, etc.), utilizada en el dashboard.

Zabbix API

- Zabbix API permite utilizar el protocolo JSON RPC para crear, actualizar y obtener objetos de Zabbix (como hosts, items, gráficos y otros) o realizar cualquier otra tarea personalizada.

Zabbix server

- un proceso central del software Zabbix que realiza la supervisión, interactúa con proxies y agents de Zabbix, calcula triggers y envía notificaciones; un repositorio central de datos.

Zabbix proxy

- un proceso que puede recopilar datos en nombre de Zabbix server, descargando al server de parte de la carga de procesamiento.

Zabbix agent

- un proceso implementado en los objetivos de supervisión para supervisar activamente los recursos y las aplicaciones locales.

Zabbix agent 2

- una nueva generación de Zabbix agent para supervisar activamente los recursos y las aplicaciones locales, que permite utilizar plugins personalizados para la supervisión.

> **Attention:**
> Dado que Zabbix agent 2 comparte gran parte de la funcionalidad con Zabbix agent, el término "Zabbix agent" en la documentación se refiere a ambos: Zabbix agent y Zabbix agent 2, cuando el comportamiento funcional es el mismo. Zabbix agent 2 se menciona específicamente solo cuando su funcionalidad difiere.

encryption

- compatibilidad con comunicaciones cifradas entre los componentes de Zabbix (server, proxy, agent, las utilidades zabbix_sender y zabbix_get) mediante el protocolo Transport Layer Security (TLS).

agent autoregistration

- proceso automatizado mediante el cual un agent de Zabbix se registra como host y comienza a supervisarlo.

network discovery

- descubrimiento automatizado de dispositivos de red.

low-level discovery

- descubrimiento automatizado de entidades de bajo nivel en un dispositivo determinado (por ejemplo, sistemas de archivos, interfaces de red, etc.).

low-level discovery rule

- conjunto de definiciones para el descubrimiento automatizado de entidades de bajo nivel en un dispositivo.

item prototype

- una métrica con determinados parámetros como variables, lista para el descubrimiento de bajo nivel. Después del descubrimiento de bajo nivel, las variables se sustituyen automáticamente por los parámetros reales descubiertos y la métrica comienza automáticamente a recopilar datos.

trigger prototype

- un trigger con determinados parámetros como variables, listo para el descubrimiento de bajo nivel. Después del descubrimiento de bajo nivel, las variables se sustituyen automáticamente por los parámetros reales descubiertos y el trigger comienza automáticamente a evaluar los datos.

También se utilizan *prototipos* de otras entidades de Zabbix en el descubrimiento de bajo nivel: prototipos de gráficos, prototipos de hosts y prototipos de grupos de hosts.

Zabbix Cloud

Zabbix Cloud

Descubra las opciones de configuración disponibles y los controles de acceso para organizaciones y nodos en Zabbix Cloud.

Tenga en cuenta que los nodos de Zabbix Cloud actualmente ejecutan **Zabbix 7.0**.
 Para obtener información detallada sobre Zabbix 7.0, consulte la documentación oficial.

Configurar y administrar Zabbix Cloud

Comenzar a usar Zabbix Cloud Instrucciones guiadas para poner en marcha rápidamente su instancia de Zabbix Cloud, incluida la configuración inicial y la conexión de sus dispositivos monitorizados.

Configuración del nodo Aprenda a configurar su nodo Zabbix Cloud, centrándose en el acceso, el manejo de datos y la configuración de recursos.

Añadir usuarios Descubra cómo añadir y asignar roles a los usuarios dentro de una organización Zabbix Cloud y sus nodos.

Conozca Zabbix Cloud

Zabbix Cloud vs. on-premises Compare las diferencias en gestión, escalabilidad y mantenimiento entre las dos implementaciones y descubra cuál puede soportar su flujo de trabajo de la manera más fluida.

Explora Zabbix Cloud Obtén una visión general de Zabbix Cloud, incluidas sus características clave, ventajas y lo que incluye una suscripción. Para preguntas comunes sobre la configuración y el uso, consulta las preguntas frecuentes.

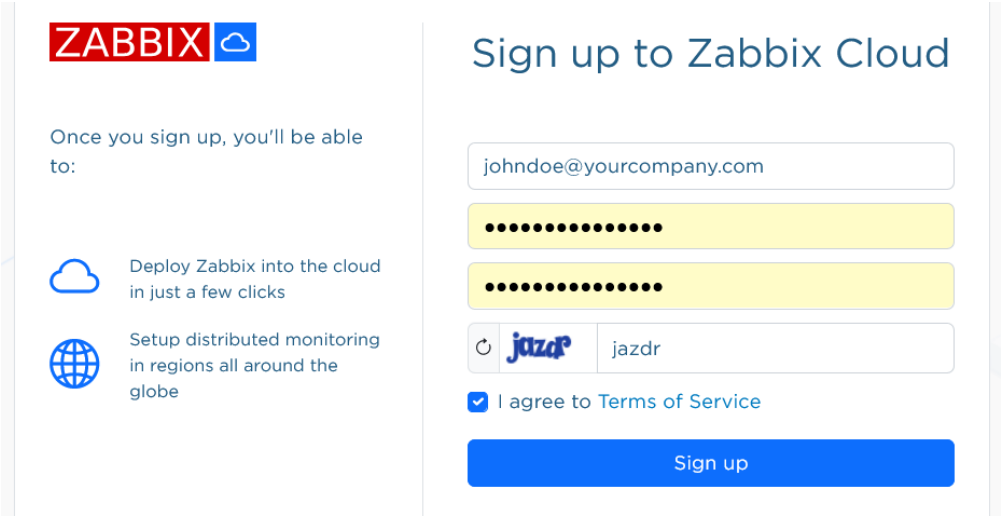
Desplegar Zabbix en la nube

Para comenzar a usar Zabbix en la nube, siga este proceso:

1. Inicie sesión o regístrese en cloud.zabbix.com.
2. Cree un nodo cloud.
3. Comience a usar Zabbix en la nube.

Iniciar sesión o registrarse Puede iniciar sesión en cloud.zabbix.com con una cuenta existente de Google, GitHub o Microsoft.

También puede ir a la página de registro para registrarse.



Después de registrarse, recibirá un correo electrónico de confirmación de su nueva cuenta.

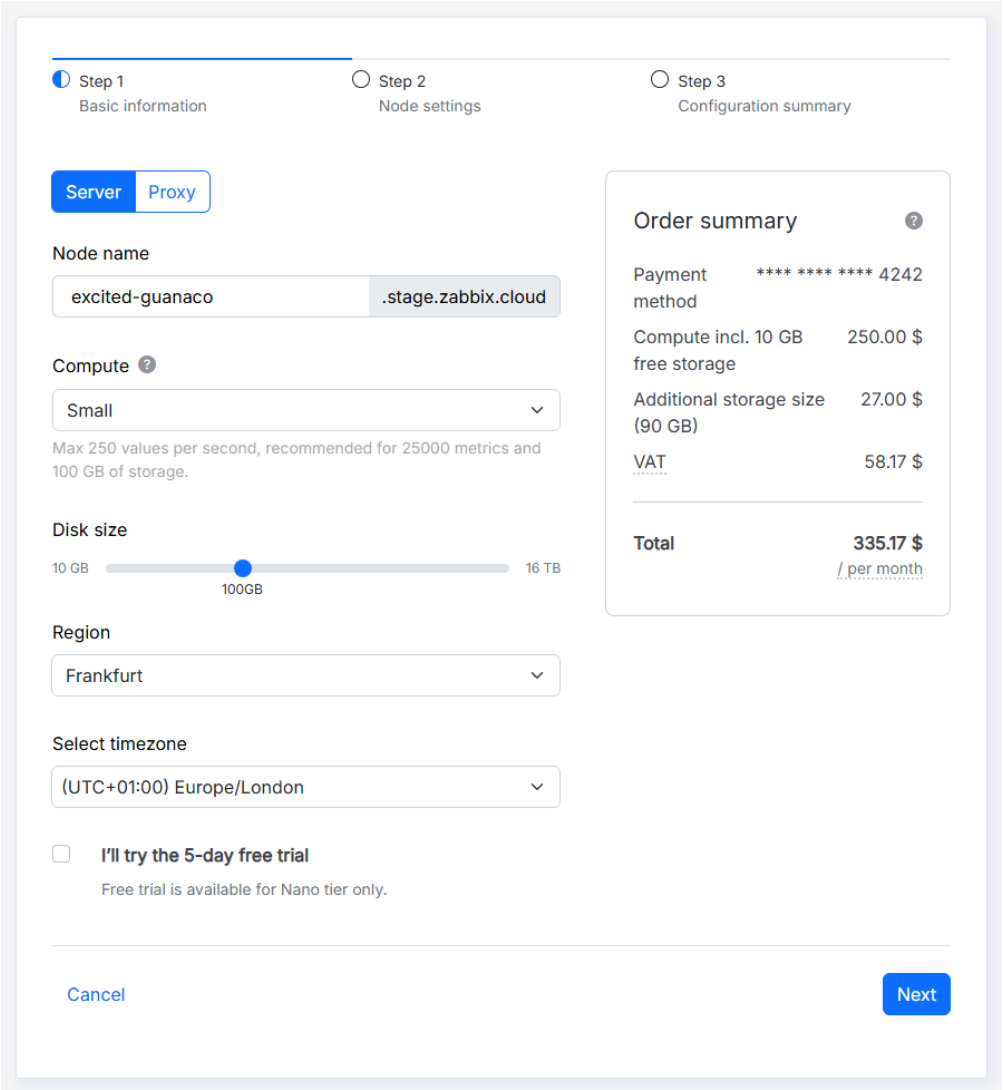
Vaya a cloud.zabbix.com para iniciar sesión. Se le solicitará una contraseña de un solo uso al iniciar sesión. Esta contraseña se le enviará por correo electrónico.

Crear un nodo cloud Un nodo cloud es una instancia de Zabbix implementada en Zabbix Cloud. Puede implementarse como un nodo server o un nodo proxy.

Al iniciar sesión, verá que ya existe una organización *Private* para su cuenta de forma predeterminada. Más adelante podrá cambiar el nombre de la organización predeterminada.

Haga clic en *Create New Node*. Se abrirá un formulario de configuración del nodo de tres pasos.

En el primer paso, seleccionará la información básica.



Parameter	Description
Server/Proxy	Seleccione si desea implementar una instancia de Zabbix server o una instancia de Zabbix proxy.
Node name	El nombre del nodo forma un subdominio único con el formato `<your-name>.zabbix.cloud`, que se utiliza para acceder a los servicios del nodo.
Compute	Seleccione la capacidad de procesamiento del nodo. Para determinar qué nivel de recursos necesita, consulte los detalles de precios. Siempre puede actualizarlo más adelante si necesita mayor capacidad. Antes de crear un nodo que no sea de prueba, será necesario configurar un método de facturación. Añada la información de facturación mediante el menú *Billing*. Hay una **prueba gratuita** de 5 días disponible para probar un nodo de nivel *Nano*. No se le cobrará durante el período de prueba, pero cuando finalice, el nodo se eliminará si no se convierte en un nodo de pago.
Disk size	El control deslizante utiliza de forma predeterminada el mínimo recomendado según el nivel. Auméntelo si almacena grandes volúmenes de datos históricos o tendencias. Tenga en cuenta que los discos más grandes tienen costes mensuales más elevados y que no puede reducir el tamaño de un disco por su cuenta; deberá ponerse en contacto con el soporte para reducirlo.
Region	Seleccione la región en la que se implementará el nodo.
Select timezone	Seleccione la zona horaria del Zabbix server. Esta zona horaria es utilizada por el Zabbix server y es independiente de las zonas horarias del mantenimiento y del perfil de usuario.

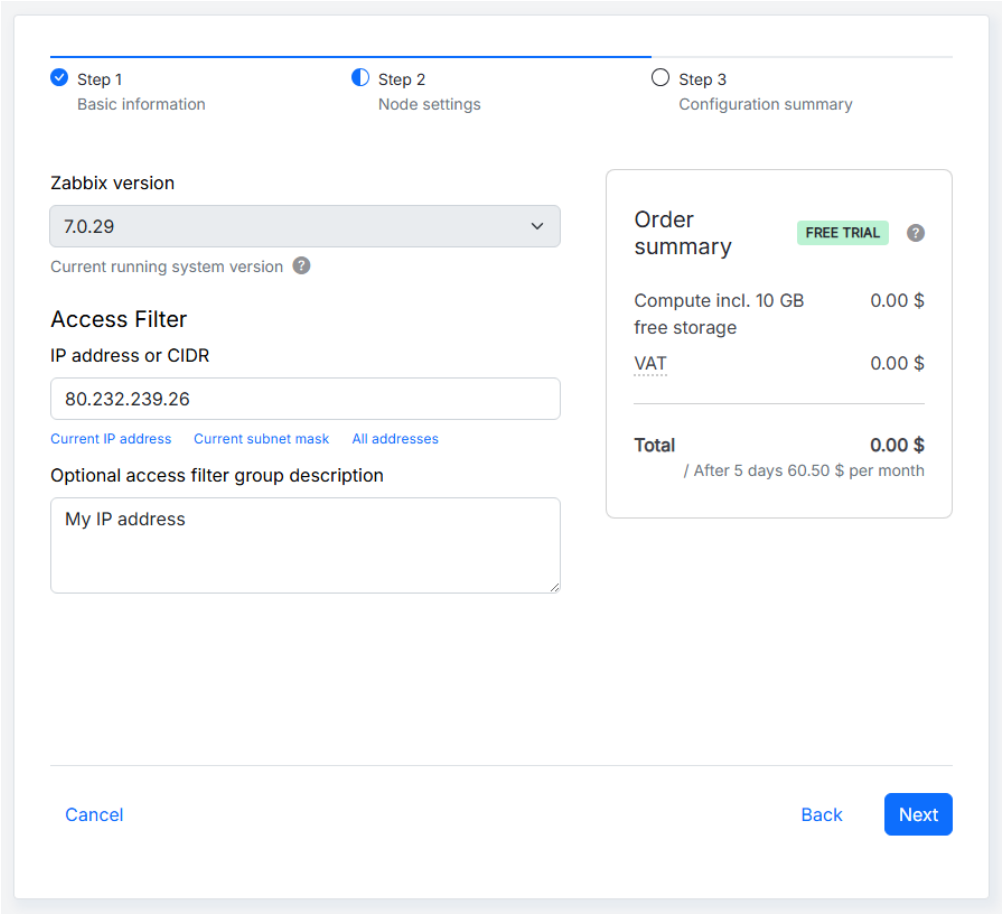
Después de completar la configuración, haga clic en *Next*.

En el segundo paso, seleccionará la configuración del nodo.

Las opciones de configuración del segundo paso dependen de si está creando un nodo server o proxy.

Configurar un nodo server

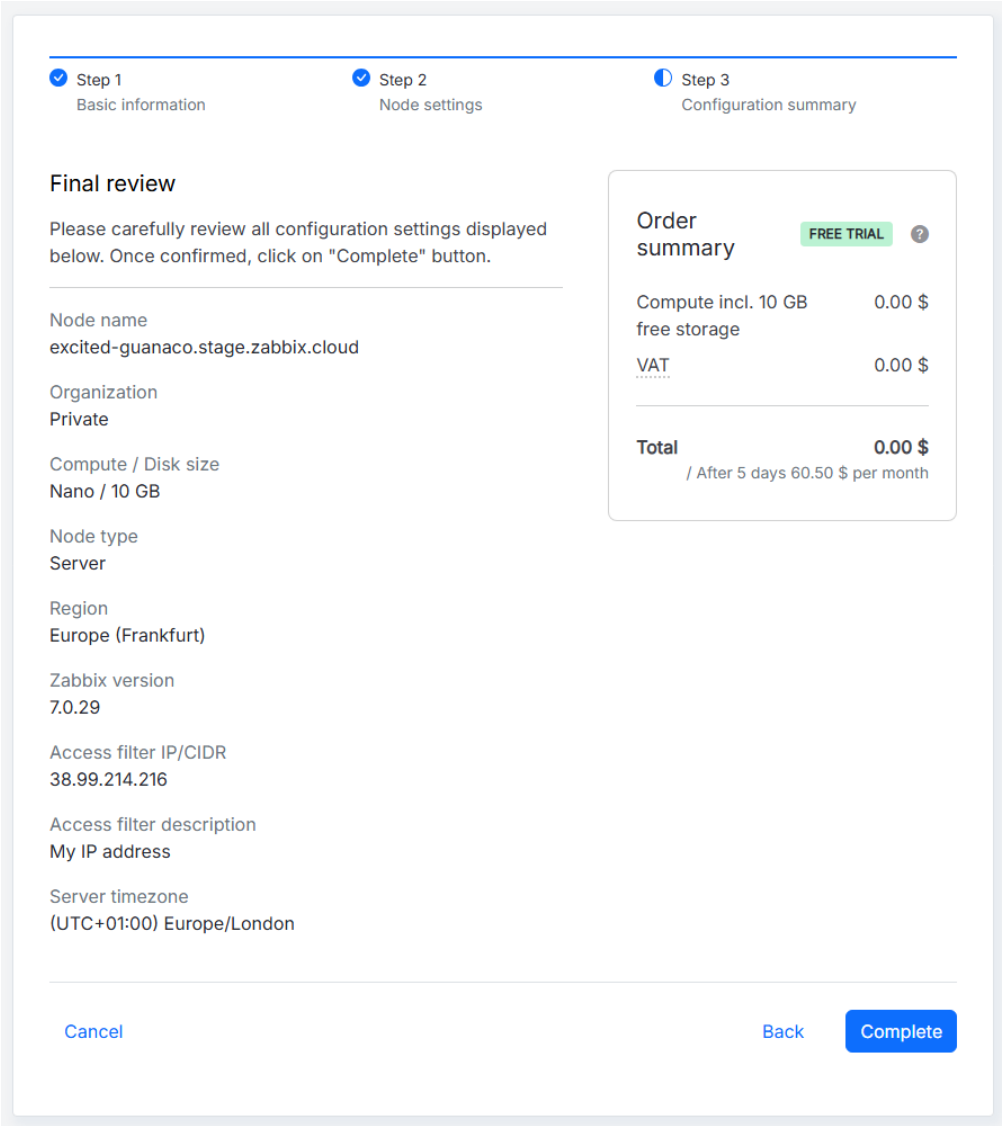
Durante la creación de un nodo server, están disponibles los siguientes parámetros:



Parameter	Description
Zabbix version	Seleccione la versión de Zabbix. Zabbix Cloud admite la versión LTS más reciente y aplica automáticamente las actualizaciones de versiones menores.
IP address or CIDR	Especifique el filtro de acceso inicial para el nodo. Introduzca manualmente la dirección IP o CIDR, o complete el campo automáticamente seleccionando la opción correspondiente. Para los nodos server, este filtro se aplica a las conexiones al server, la API y el frontend. El filtro puede modificarse posteriormente en la pestaña Access filters.
Optional access filter group description	Añada una descripción opcional para el grupo de filtros de acceso, que puede contener varias direcciones IP o CIDR.

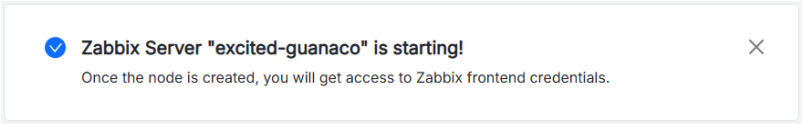
Warning:
Durante la creación del nodo, el campo *IP address or CIDR* se completa previamente con la dirección IP utilizada para acceder a Zabbix Cloud. Puede cambiar este valor para especificar un filtro de acceso inicial diferente. Después de crear el nodo, puede añadir, editar o eliminar filtros de acceso desde la pestaña *Access filters*.

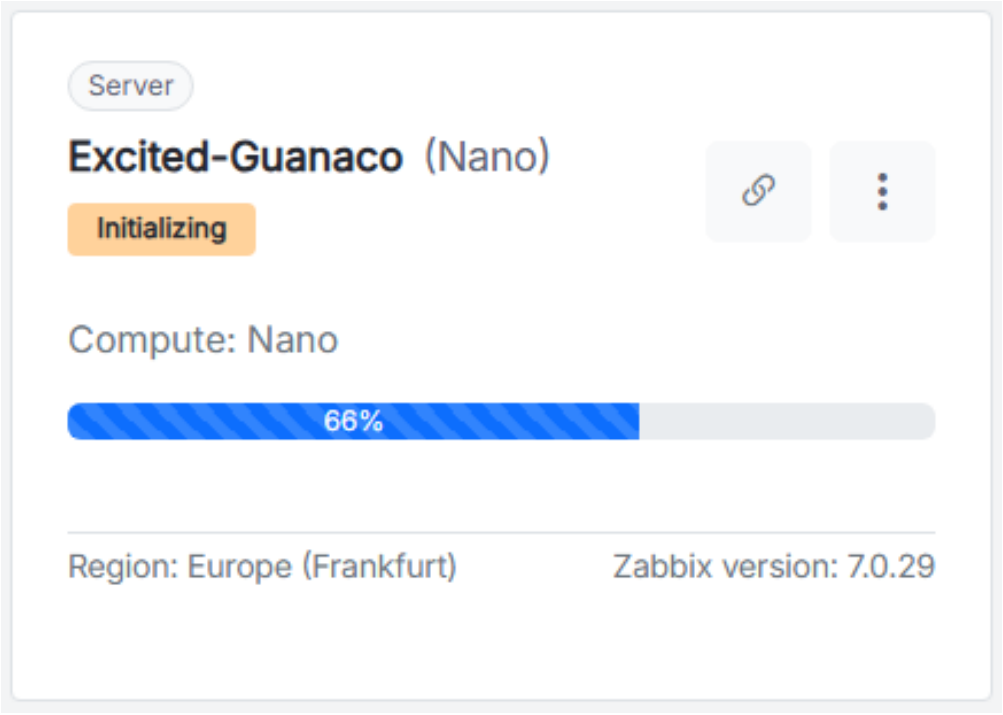
El tercer paso muestra un resumen de la configuración del nodo.



Después de revisar la configuración y confirmar que todo es correcto, haga clic en *Complete* para un nodo de prueba o en *Proceed to payment* para un nodo de pago. De lo contrario, vuelva a un paso anterior haciendo clic en el botón *Back*, o haga clic en *Cancel* para comenzar desde el principio.

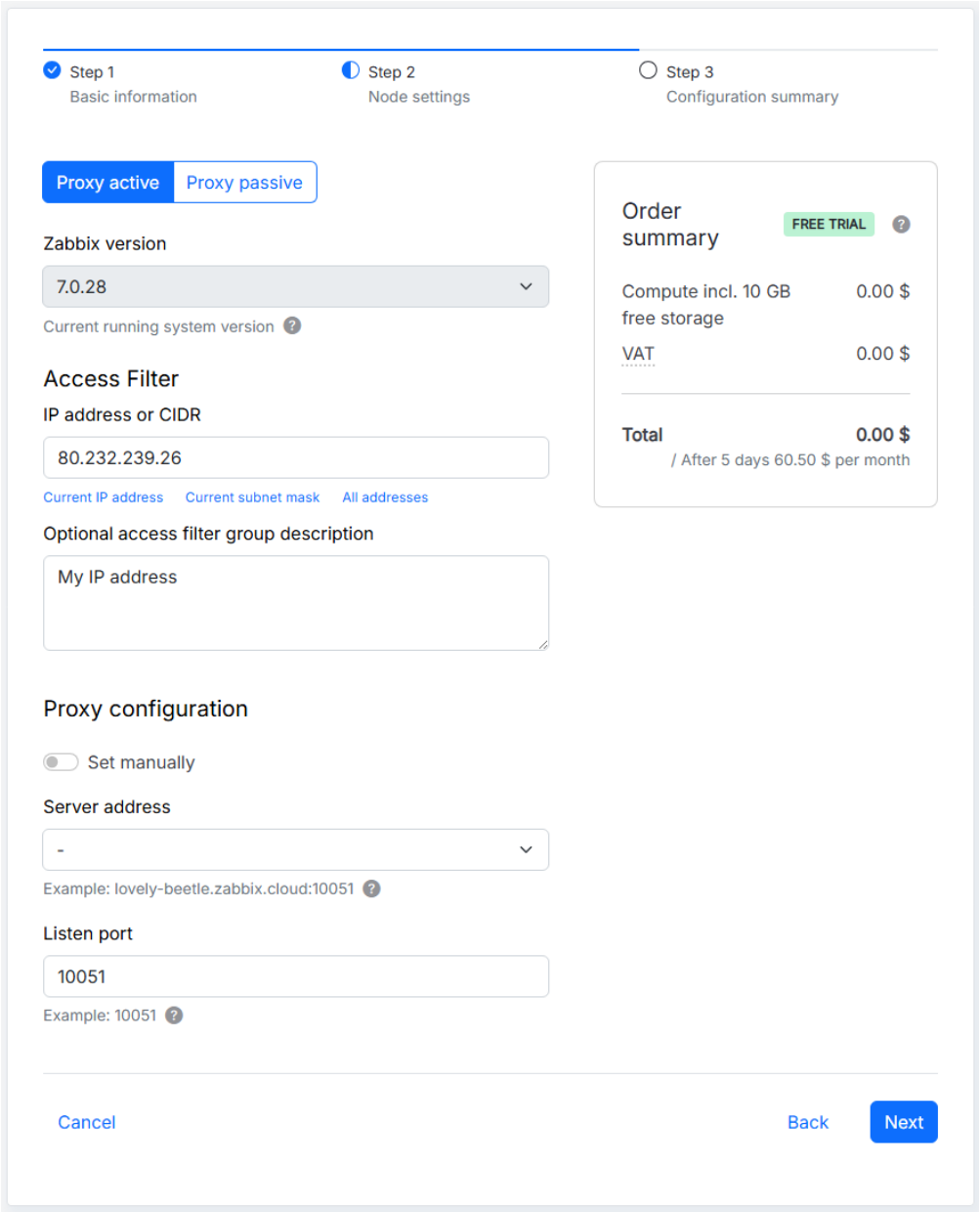
Mientras se inicializa el nodo, se muestra el mensaje correspondiente y el nodo ya aparece en *Organizations*.





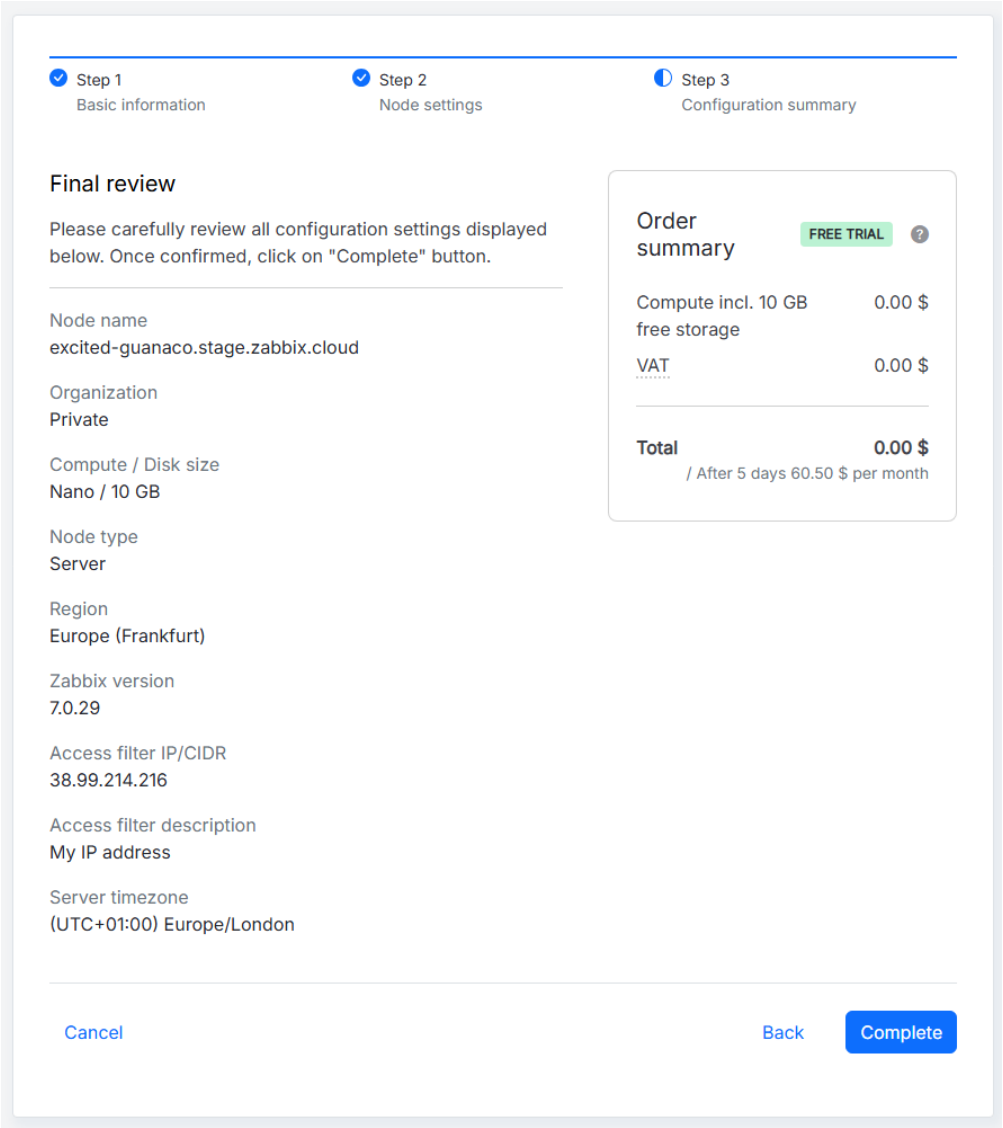
Configurar un nodo proxy

Durante la creación del nodo proxy, están disponibles los siguientes parámetros:



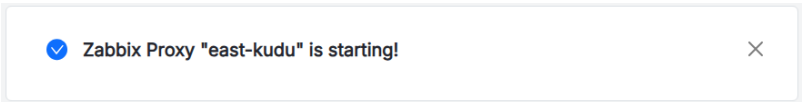
Parameter	Description
Proxy active/Proxy passive	Seleccione el modo del proxy. En el modo activo, el proxy inicia las conexiones al Zabbix server; en el modo pasivo, el Zabbix server inicia las conexiones al proxy.
Zabbix version	Seleccione la versión de Zabbix. Zabbix Cloud admite la versión LTS más reciente y aplica automáticamente las actualizaciones de versiones menores.
IP address or CIDR	Especifique el filtro de acceso IP entrante inicial para el proxy. Introduzca manualmente la dirección IP o CIDR, o complete el campo automáticamente seleccionando la opción correspondiente. El filtro define qué solicitudes de red pueden llegar al proxy y puede modificarse posteriormente en la pestaña Access filters.
Optional access filter group description	Añada una descripción opcional para el grupo de filtros de acceso, que puede contener varias direcciones IP o CIDR.
Set manually	Seleccione cómo especificar el Zabbix server. Cuando está deshabilitado, seleccione un nodo server de la lista de nodos server disponibles. Cuando está habilitado, especifique manualmente las direcciones del server.
Server address	Especifique la dirección del Zabbix server. Cuando *Set manually* está deshabilitado, seleccione un nodo server de la lista. Cuando está habilitado, especifique hasta 10 direcciones del server. Para un proxy activo, especifique la dirección IP o el nombre DNS del Zabbix server. Para un proxy pasivo, especifique la dirección IP, el nombre DNS o el CIDR del Zabbix server. Los filtros de acceso se configuran por separado y no se actualizan automáticamente al especificar una dirección del server.
Listen port	Especifique el puerto en el que el proxy escucha las conexiones entrantes. El puerto predeterminado es 10051. El puerto de escucha no puede cambiarse después de crear el nodo.

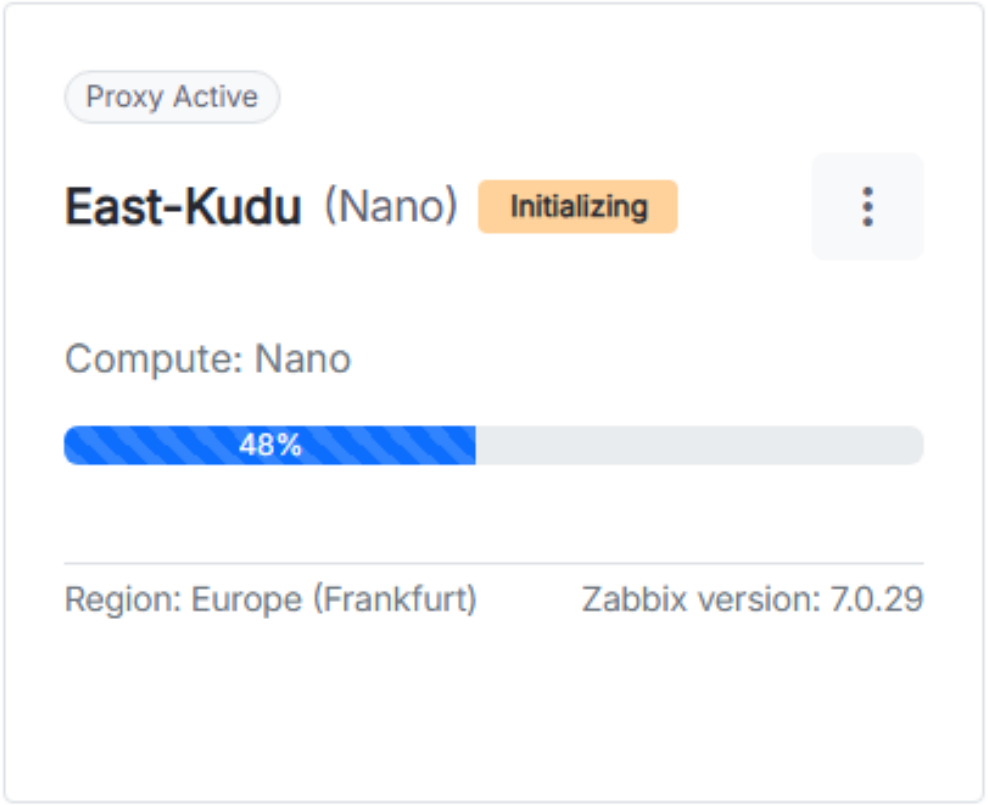
En el tercer paso, verá el resumen de la configuración del nodo.



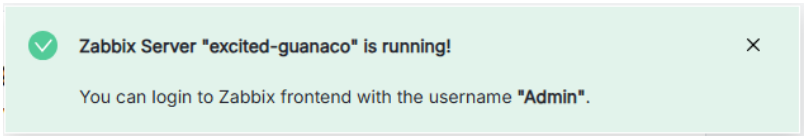
Después de revisar la configuración y confirmar que todo es correcto, haga clic en *Complete* para un nodo de prueba o en *Proceed to payment* para un nodo de pago. De lo contrario, vuelva a los pasos anteriores haciendo clic en el botón *Back*, o haga clic en *Cancel* para comenzar desde el principio.

Mientras se inicializa el nodo, se muestra el mensaje correspondiente y el nodo ya aparece en *Organizations*.





Para configurar un proxy de Zabbix Cloud, consulte Configurar un proxy de Zabbix Cloud.

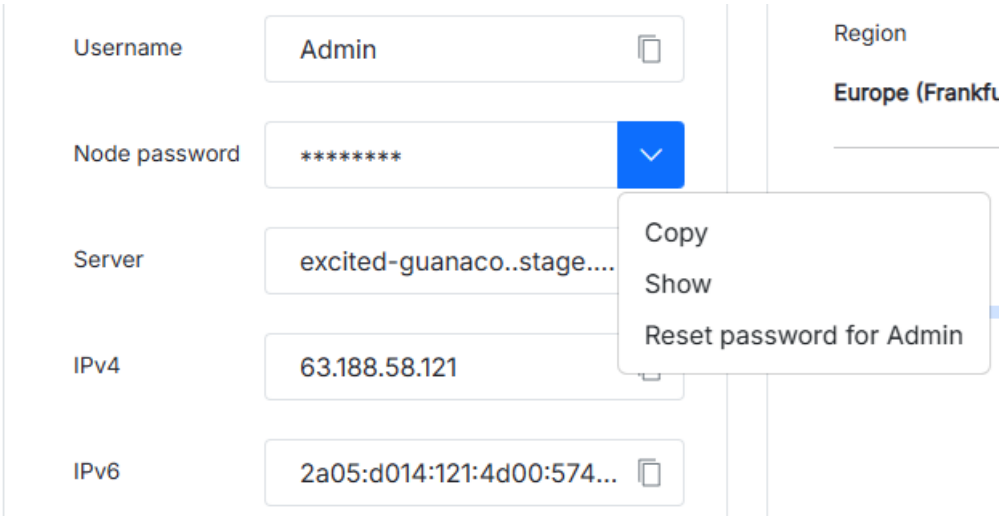


Comenzar a usar Zabbix

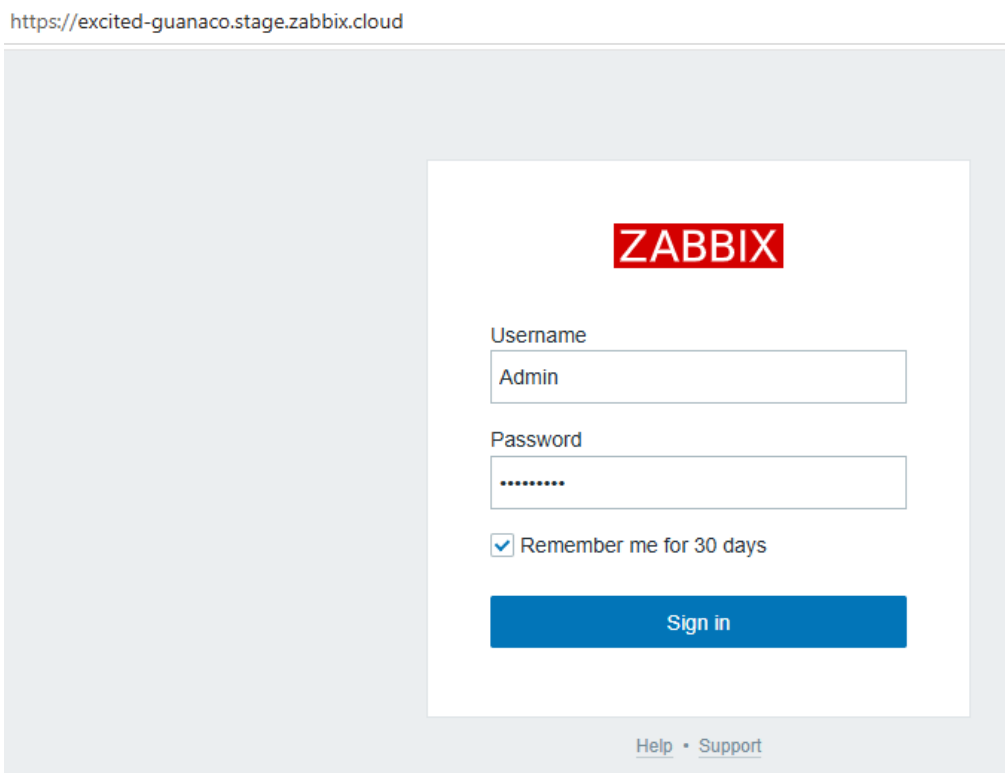
Cuando el estado del nodo server cambie de *Initializing* a *Running*, podrá iniciar sesión en el frontend de Zabbix.

Para acceder a la contraseña inicial del frontend, haga clic en el nodo creado.

Se le dirigirá a la sección settings del nodo (descripción general, filtros de acceso, cifrado, copias de seguridad, etc.). En **Connection info**, localice el campo *Node password* y seleccione *Reset password for Admin* en la lista desplegable. Después de restablecer la contraseña, haga clic en la misma lista desplegable y seleccione *Copy*.



Haga clic en la URL del frontend para abrir el frontend de Zabbix (`excited-guanaco.zabbix.cloud`).



Inicie sesión con las siguientes credenciales:

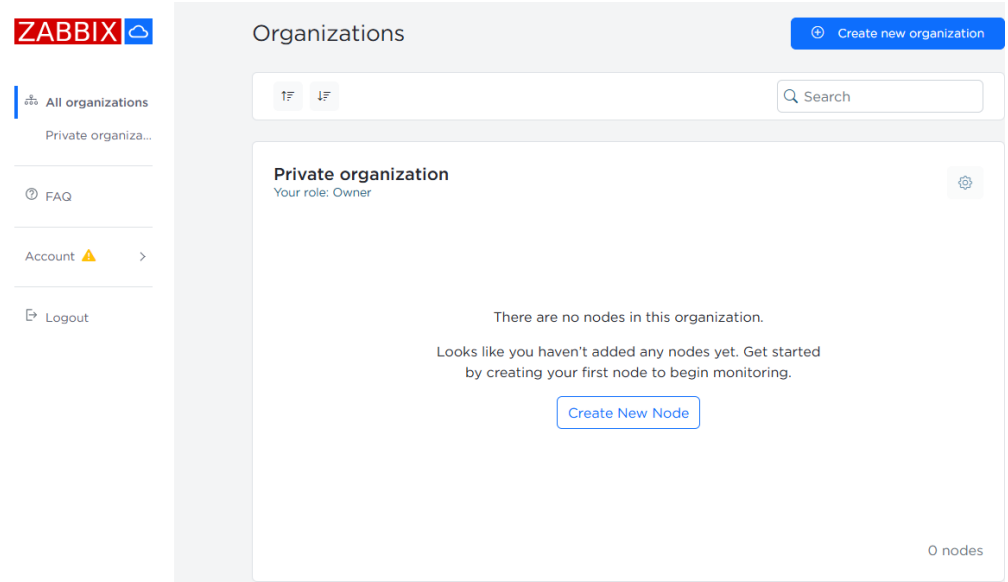
- *Username* - `Admin`
- *Password* - proporcione la contraseña copiada

Para comenzar a supervisar, debe definir access filters: la dirección IP o las direcciones IP que tienen permitido conectarse al server de Zabbix Cloud. Debe definirse al menos una.

Gestión de organizaciones Una organización es un contenedor para uno o más nodos.
Las organizaciones también le ayudan a estructurar sus recursos y delegar responsabilidades, como:

- Gestión de nodos
- Facturación

Al iniciar sesión en Zabbix Cloud, se muestra la página *Todas las organizaciones*.
De forma predeterminada, se crea una organización privada para su cuenta.



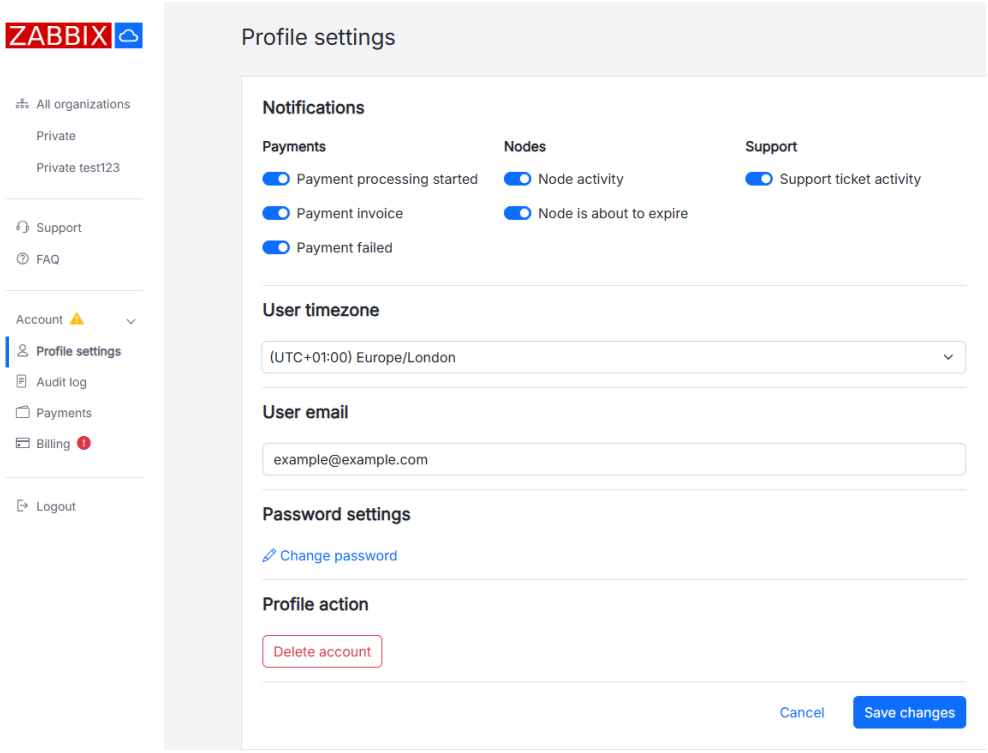
Como propietario de esta organización, puede gestionar usuarios, administradores, nodos y el nombre de la organización.

Haga clic en el botón de configuración  para acceder a las opciones de gestión.

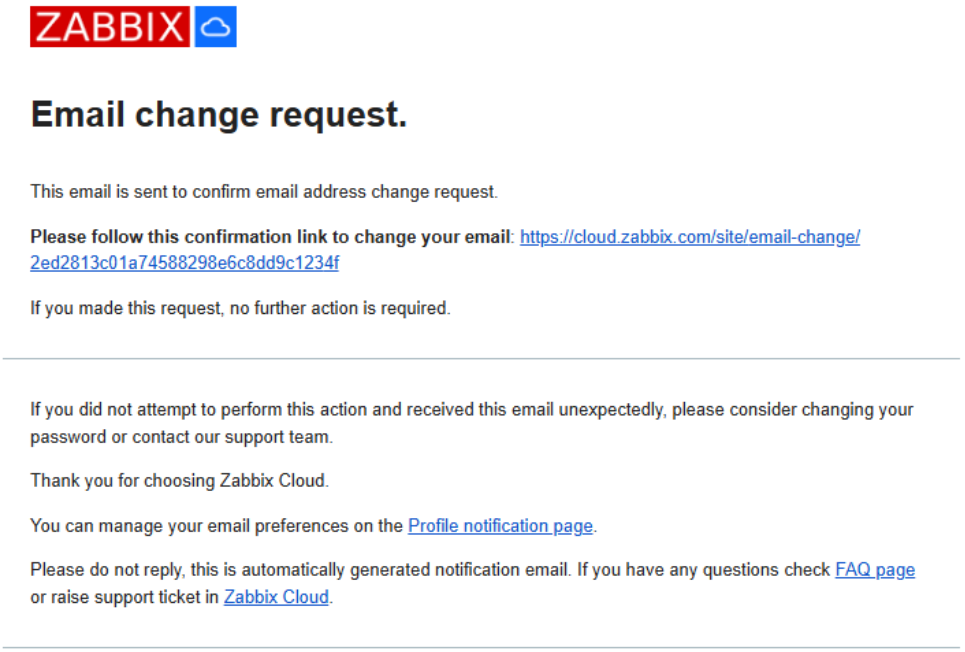
Gestión de cuentas Cambio de la dirección de correo electrónico de la cuenta

Para cambiar la dirección de correo electrónico de la cuenta, vaya a *Cuenta* > *Configuración del perfil*.

En el campo *Correo electrónico del usuario*, introduzca la nueva dirección de correo electrónico y haga clic en *Guardar cambios*.



Se envía un mensaje de confirmación a la nueva dirección de correo electrónico. Siga el enlace incluido en el mensaje para confirmar el cambio de dirección de correo electrónico.



El enlace de confirmación abre la página *Configuración del perfil* con un mensaje de confirmación y la dirección de correo electrónico actualizada en el campo Correo electrónico del usuario.

Una vez confirmado el cambio, se envía un correo electrónico de confirmación tanto a la nueva dirección como a la anterior. El mensaje enviado a la dirección anterior incluye las direcciones de correo electrónico anterior y nueva.

Configuración del nodo

Descripción general

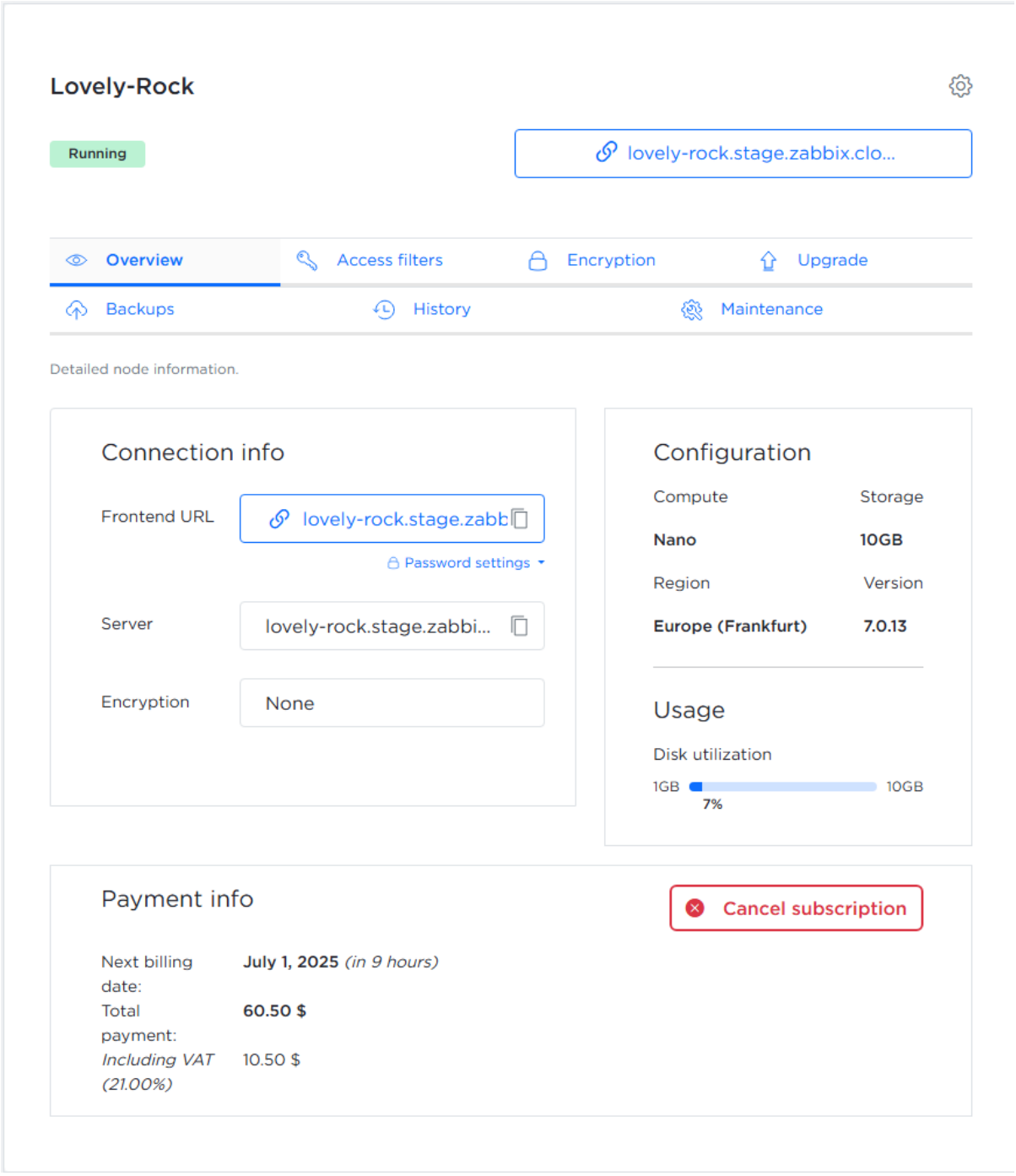
Un **nodo en la nube** es un componente de Zabbix implementado en la nube. Los nodos server incluyen el server de Zabbix, la base de datos y el web frontend, mientras que los nodos proxy incluyen el proxy de Zabbix y la base de datos. Un nodo en la nube tiene recursos informáticos dedicados (CPU, memoria) y almacenamiento en disco.

Los usuarios pueden crear uno o varios nodos en la nube. Los nodos en la nube pueden agruparse en **organizaciones** (estructurales, regionales, etc.), creando una configuración de monitorización distribuida.

Los nodos y las organizaciones en la nube se crean mediante la interfaz de usuario de Zabbix Cloud. Para acceder a la página de configuración del nodo, haga clic en el nodo desde *Todas las organizaciones*, desde su menú de organización o en el enlace *Página de configuración del nodo* que se muestra después de iniciar el nodo.

En la vista de configuración del nodo, cada pestaña permite gestionar un aspecto específico del nodo:

- Descripción general: ver los detalles del nodo, abrir su frontend y gestionar los usuarios del nodo.
- Filtros de acceso: controlar el acceso mediante una dirección IP o una subred.
- Cifrado: configurar TLS cargando certificados, claves y CRL opcionales.
- Copias de seguridad: crear, restaurar y eliminar instantáneas del nodo; ver la próxima copia de seguridad programada.
- Historial: establecer los periodos de retención para los registros de auditoría, el historial y las tendencias, con unidades predefinidas y un límite de actualización de 3 veces al día.
- Mantenimiento: definir ventanas de mantenimiento semanales por zona horaria, día y hora.
- Actualización: escalar los recursos informáticos o de almacenamiento, o gestionar el estado de prueba/pago.

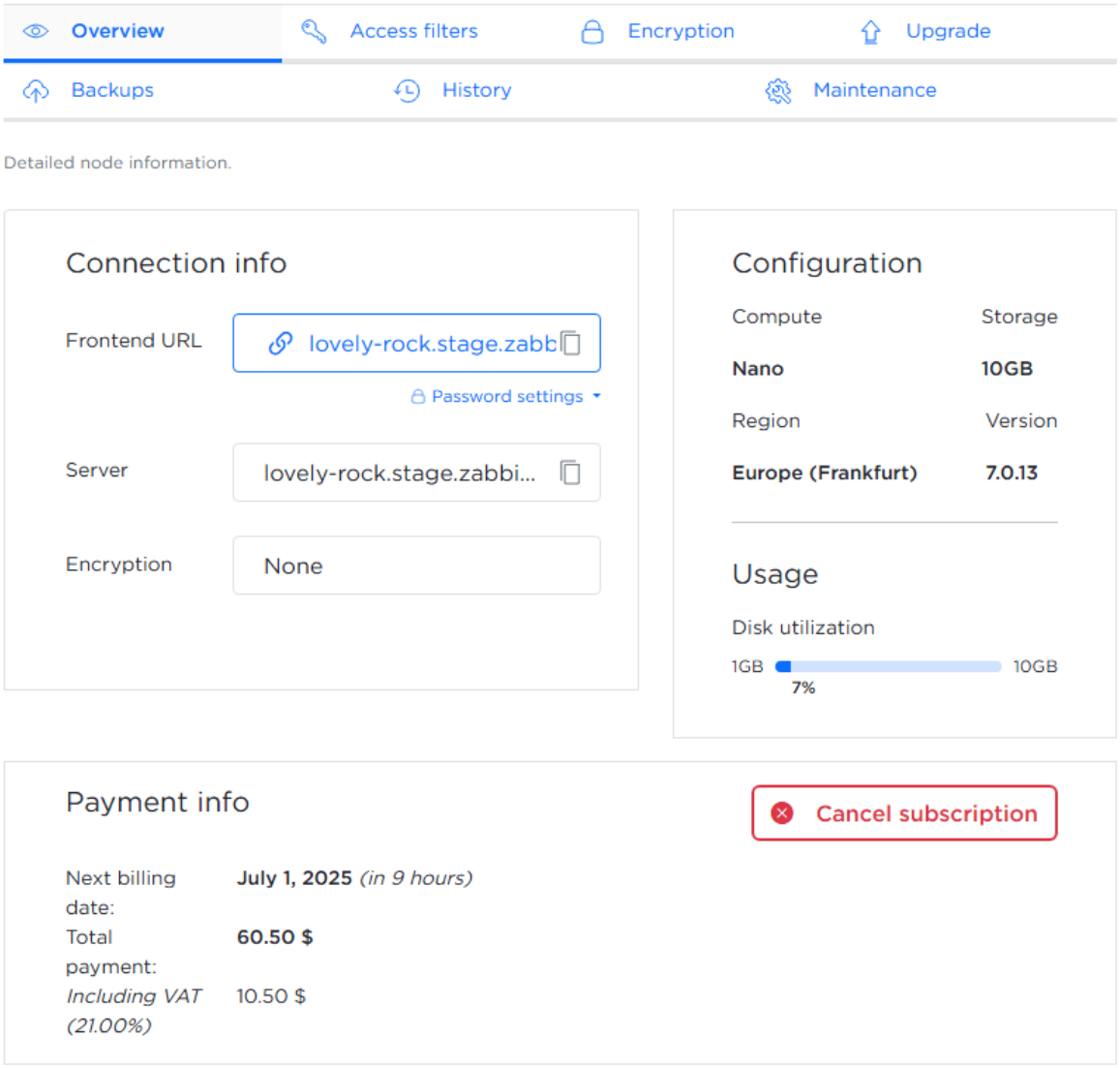


Note:
Es posible que algunas pestañas no estén disponibles según el tipo de nodo (server o proxy), el tipo de usuario que visualiza la página y el estado del nodo (de prueba/no de prueba).

Pestaña Descripción general

La pestaña *Descripción general* proporciona información detallada sobre el nodo. La información mostrada depende de si el nodo es un server o un proxy.

Nodo del server La pestaña *Overview* del nodo del server contiene las siguientes secciones:



Connection info:

- URL del frontend en la que se puede hacer clic, con una opción para copiarla al portapapeles
- Nombre de usuario
- Contraseña del nodo y su configuración
- Nombre de host del server, con una opción para copiarlo al portapapeles
- Dirección IPv4, con una opción para copiarla al portapapeles
- Dirección IPv6, con una opción para copiarla al portapapeles
- Método de cifrado actual

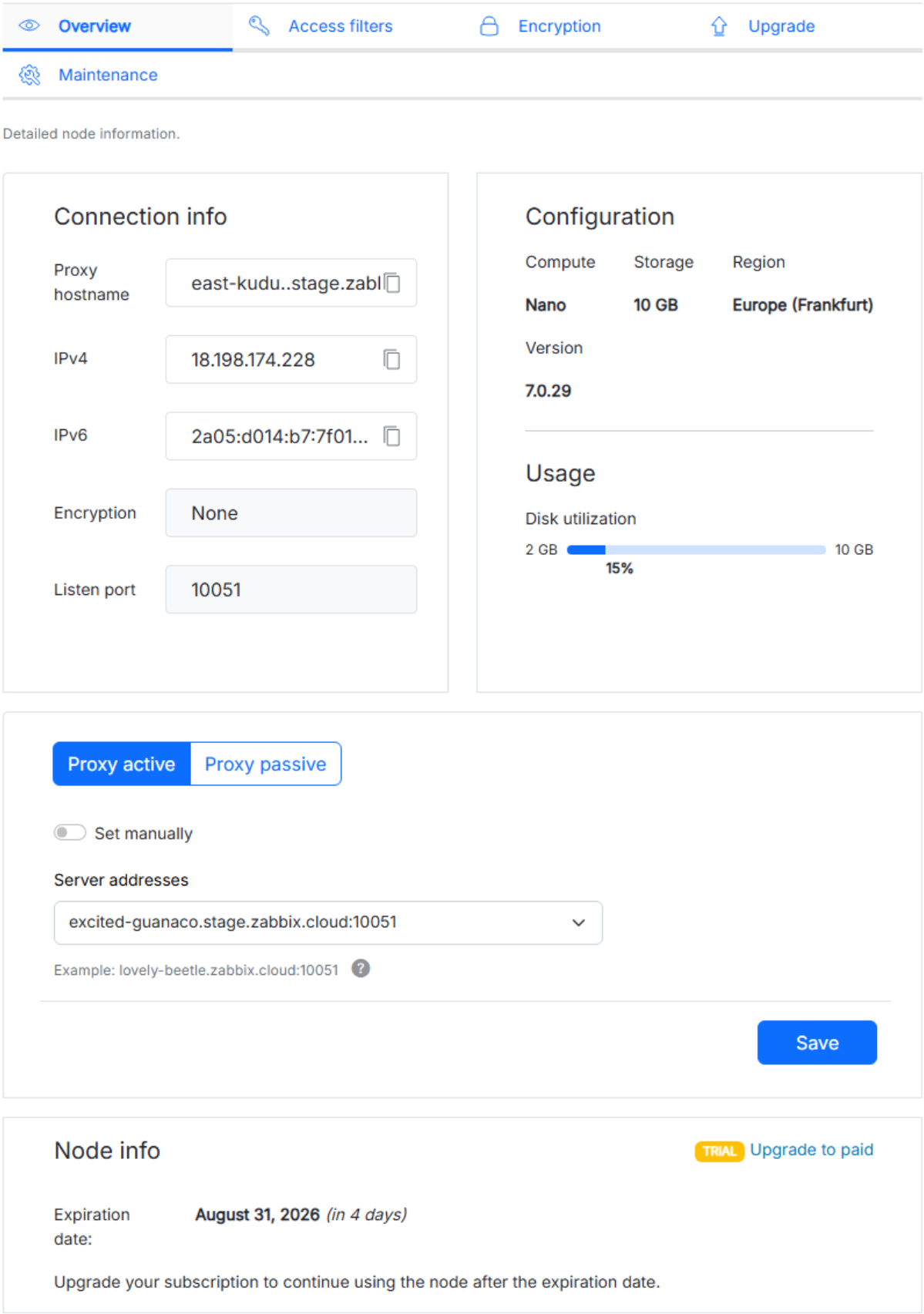
Configuration and usage:

- Nivel de cómputo, almacenamiento máximo, región
- Versión de los componentes de Zabbix
- Uso del disco

Node info/Payment Info:

- Fecha de vencimiento (para nodos de prueba y prepagados)
- Se muestra un botón específico del contexto según el tipo de nodo una vez que el nodo completa la inicialización:
  - *Upgrade to paid*: disponible para nodos en una prueba gratuita
  - *Cancel subscription*: disponible para nodos de pago con una suscripción activa
  - *Resubscribe to paid*: se muestra si se ha cancelado una suscripción
  - No se muestra ningún botón de acción para los nodos prepagados, que tienen una validez prolongada y no se pueden cancelar

Nodo proxy La pestaña *Overview* del nodo proxy contiene las siguientes secciones:



Connection info:

- Nombre de host del proxy con una opción para copiarlo al portapapeles
- Dirección IPv4 con una opción para copiarla al portapapeles
- Dirección IPv6 con una opción para copiarla al portapapeles
- Método de cifrado actual
- Puerto de escucha actual

Configuration and usage:

- Nivel de cómputo, almacenamiento máximo, región
- Versión de los componentes de Zabbix
- Uso del disco

Proxy mode:

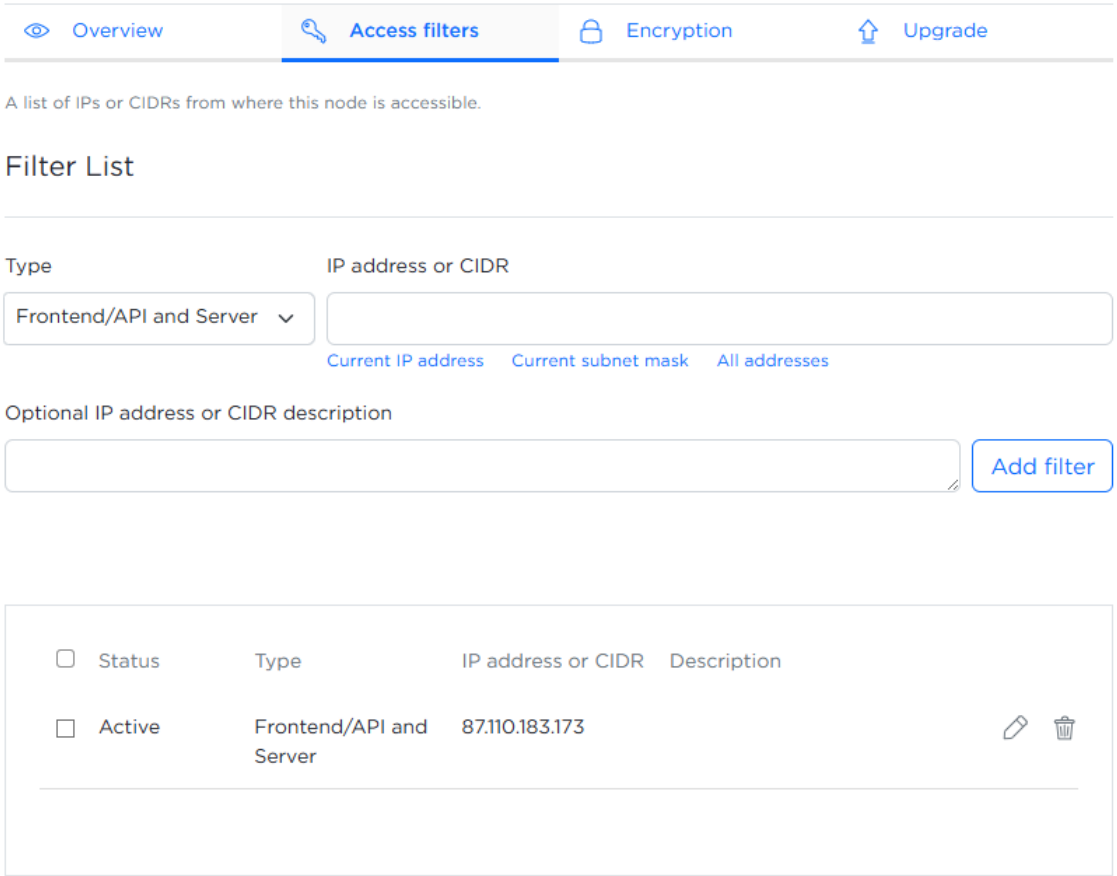
- Cambiar entre los modos de proxy activo y pasivo
- Seleccionar la dirección del server en la lista desplegable *Server addresses* o activar *Set manually* para introducir manualmente la dirección del server

Node info/Payment Info:

- Fecha de vencimiento (para nodos de prueba y prepagados)
- Se muestra un botón específico del contexto según el tipo de nodo una vez que el nodo completa la inicialización: - *Upgrade to paid*: disponible para nodos en una prueba gratuita - *Cancel subscription*: disponible para nodos de pago con una suscripción activa - *Resubscribe to paid*: se muestra si se ha cancelado una suscripción - No se muestra ningún botón de acción para los nodos prepagados, que tienen una validez prolongada y no se pueden cancelar

Pestaña de filtros de acceso

La pestaña *Filtros de acceso* permite restringir el acceso a su nodo de Zabbix por dirección IP o subred.



Para configurar un filtro de acceso, complete los siguientes campos:

- *Tipo*: para los nodos server, use la lista desplegable para elegir qué componente de Zabbix desea limitar: solo frontend/API, solo server o ambos. Para los nodos proxy, solo está disponible *IP entrante*. Define qué solicitudes de red entrantes pueden llegar al proxy.
- *Dirección IP o CIDR*: introduzca manualmente la dirección IP o el CIDR, o complete el campo automáticamente seleccionando la opción correspondiente.
- *Descripción opcional de la dirección IP o CIDR*: añada una descripción opcional para cada IP o subred.

Después de completar los campos, haga clic en *Añadir filtro*. Los filtros añadidos aparecen en la tabla siguiente, donde se muestran el estado, el tipo, la dirección IP o subred y la descripción.

Warning:
Durante la inicialización del nodo, se crea automáticamente un filtro de acceso para la dirección IP que utilizó. Si posteriormente se conecta desde una IP o red diferente, debe añadir esa dirección aquí; de lo contrario, se bloqueará el acceso.

Puede:

-  : editar un filtro.
-  : eliminar un filtro o varios filtros mediante las casillas de verificación.

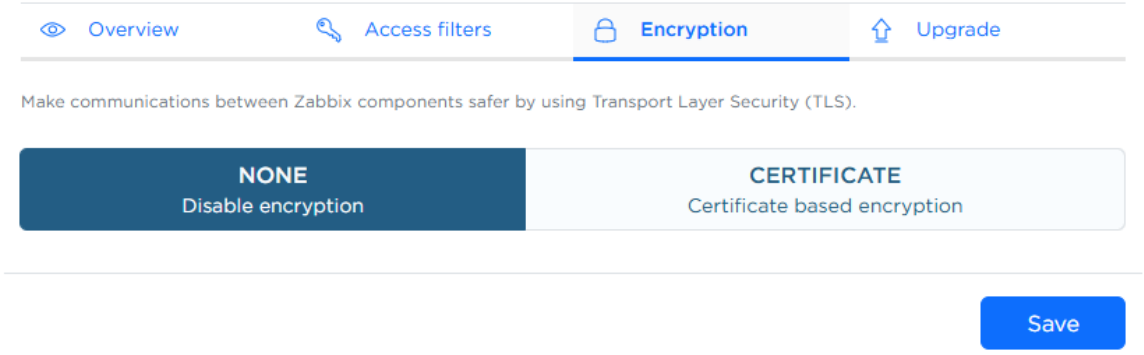
Pestaña Encryption

La pestaña *Encryption* permite configurar el cifrado TLS para la comunicación entre los componentes de Zabbix mediante la carga de los archivos de certificado necesarios.

Para habilitar la comunicación segura mediante certificados, seleccione *CERTIFICATE* y cargue los siguientes archivos desde su equipo:

- *Certificados de CA raíz* - certificados de CA de confianza utilizados para verificar los certificados de los pares. Haga clic en *Choose file* y seleccione el archivo que contiene los certificados de CA raíz de confianza (normalmente un archivo `.crt` o `.pem`, codificado en PEM).
- *Cadena de certificados* - representa la cadena de confianza de su propio certificado. Haga clic en *Choose file* y seleccione el archivo codificado en PEM (por ejemplo, `.crt` o `.pem`) que incluye su certificado y cualquier certificado de CA intermedia.
- *Clave privada* - la clave privada correspondiente a su certificado. Haga clic en *Choose file* y seleccione el archivo `.key` codificado en PEM.
- *Certificados revocados* (opcional) - si mantiene una lista de revocación de certificados (CRL), haga clic en *Choose file* y seleccione el archivo CRL para especificar los certificados que ya no deben considerarse de confianza.

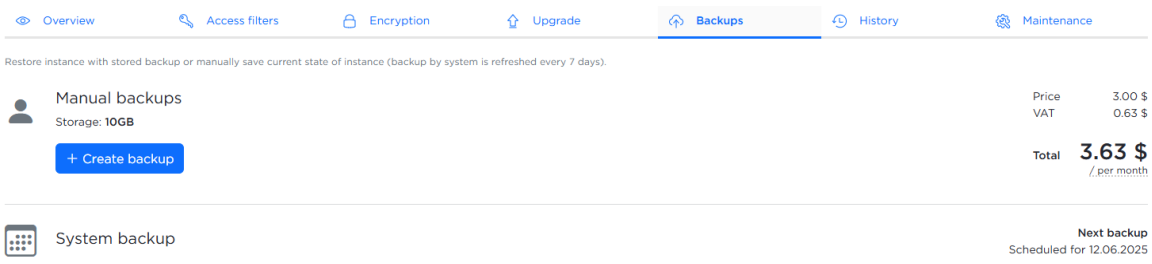
Después de seleccionar los archivos, haga clic en *Save*. Los archivos se cargarán, validarán y aplicarán para configurar el cifrado TLS entre los componentes de Zabbix.



Pestaña Backups

La pestaña *Backups* permite ver, crear, restaurar y eliminar copias de seguridad de su nodo de Zabbix.

Puede restaurar su instancia desde cualquier copia de seguridad guardada o crear una nueva copia de seguridad manual en cualquier momento.
La pestaña también muestra la fecha de la próxima copia de seguridad programada.



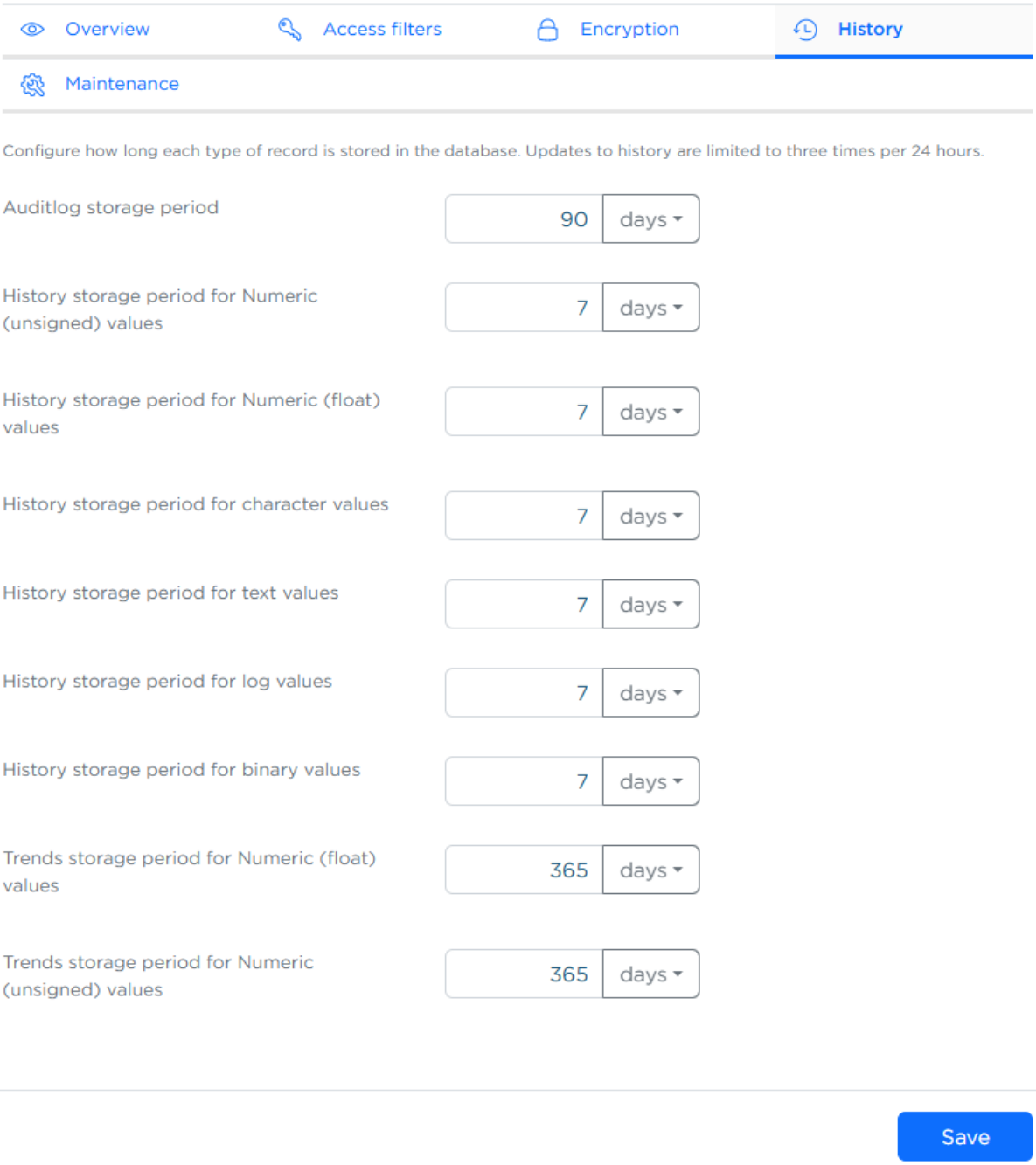
Para crear una copia de seguridad:

1. Haga clic en el botón *Create backup*.
2. Introduzca una descripción opcional en la ventana emergente.
3. Haga clic en *Create* para confirmar.

Warning:
Las copias de seguridad adicionales se cobran mensualmente en función del uso de almacenamiento.

Pestaña History

La pestaña *History* permite configurar durante cuánto tiempo se almacenan los distintos tipos de datos históricos.



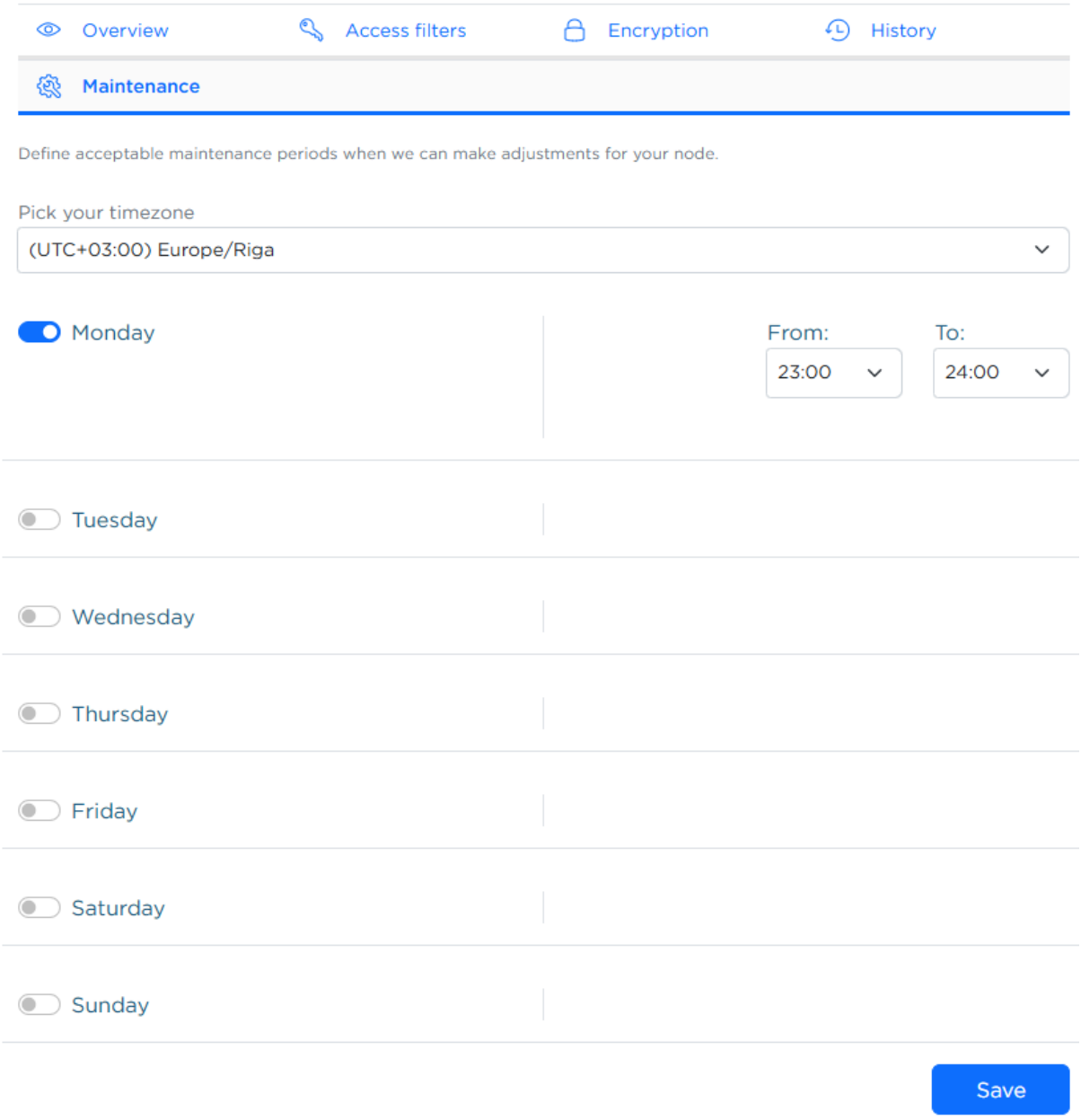
Puede establecer el período de almacenamiento para cada tipo de datos (por ejemplo, numéricos, de texto y registros) y elegir la unidad (días, meses o años). Los cambios pueden realizarse hasta tres veces cada 24 horas.

Después de actualizar los valores, haga clic en *Save* para aplicar los cambios.

Warning:
Los períodos de retención más largos pueden aumentar el uso del disco y afectar al rendimiento. Se recomienda configurar los períodos de retención en función de sus datos reales y sus necesidades de análisis.
 Zabbix Cloud utiliza la partición de la base de datos para gestionar el almacenamiento de datos en lugar del housekeeping tradicional de Zabbix. Esto permite un rendimiento más rápido y una gestión de la retención más predecible.

Pestaña Mantenimiento

La pestaña *Mantenimiento* permite definir ventanas de mantenimiento aceptables durante las cuales se pueden realizar actualizaciones o ajustes en su nodo.

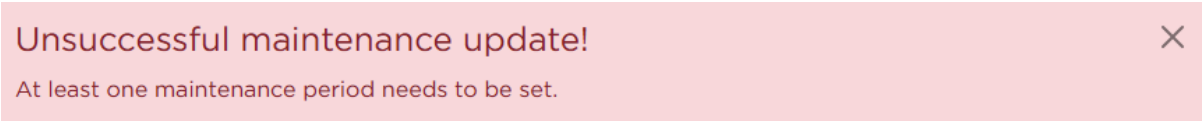


Debe:

- Seleccionar su zona horaria en la lista desplegable.
- Elegir al menos un día y una ventana horaria para el mantenimiento.

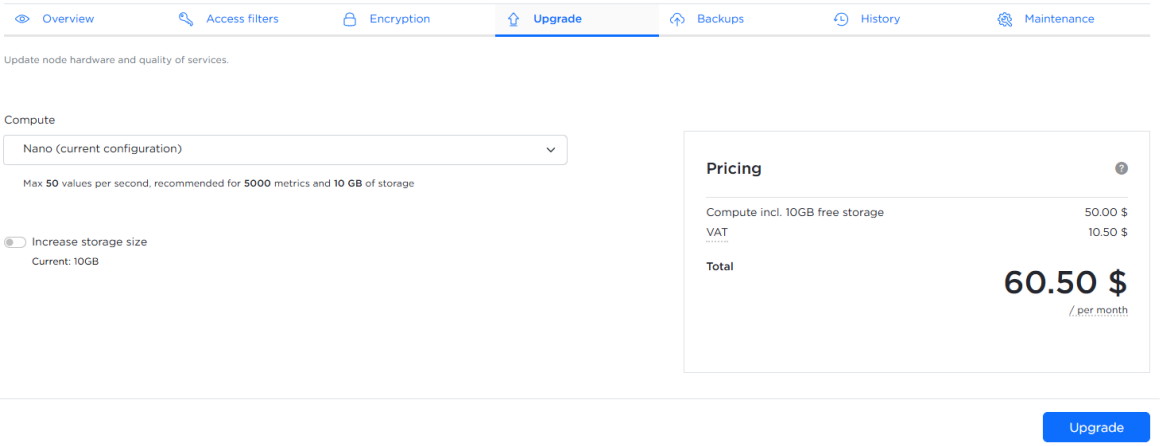
Haga clic en *Guardar* para aplicar los cambios.

Si no se selecciona ningún día ni ventana horaria, se mostrará un error.



Pestaña Upgrade

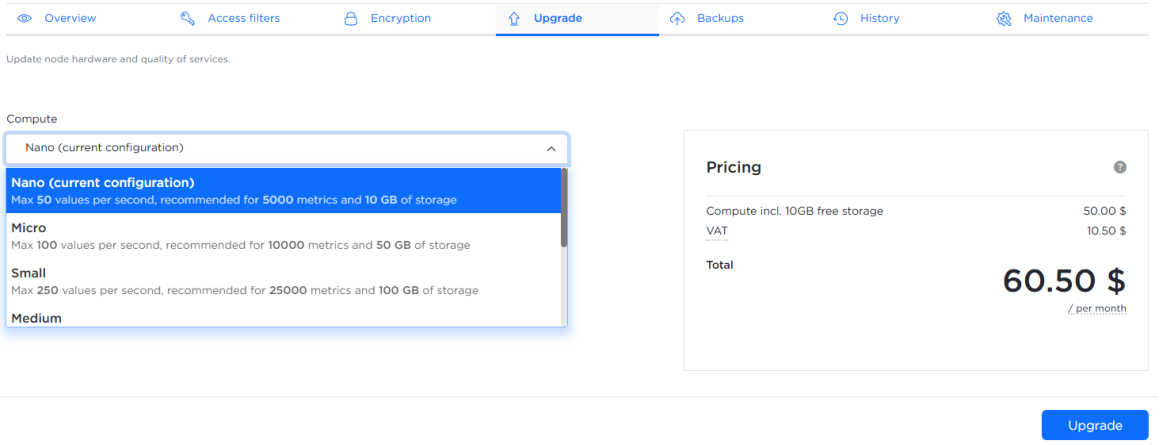
La pestaña *Upgrade* permite ampliar los recursos de su nodo o gestionar su suscripción.



Puede aumentar el nivel de cómputo o el tamaño del disco. Tenga en cuenta que el tamaño del disco solo puede aumentarse una vez cada 24 horas y no puede reducirse mediante la interfaz de usuario (póngase en contacto con el soporte de Zabbix para solicitar una reducción). Los nodos de prueba gratuita pueden convertirse en suscripciones de pago.

Para aumentar el nivel de cómputo:

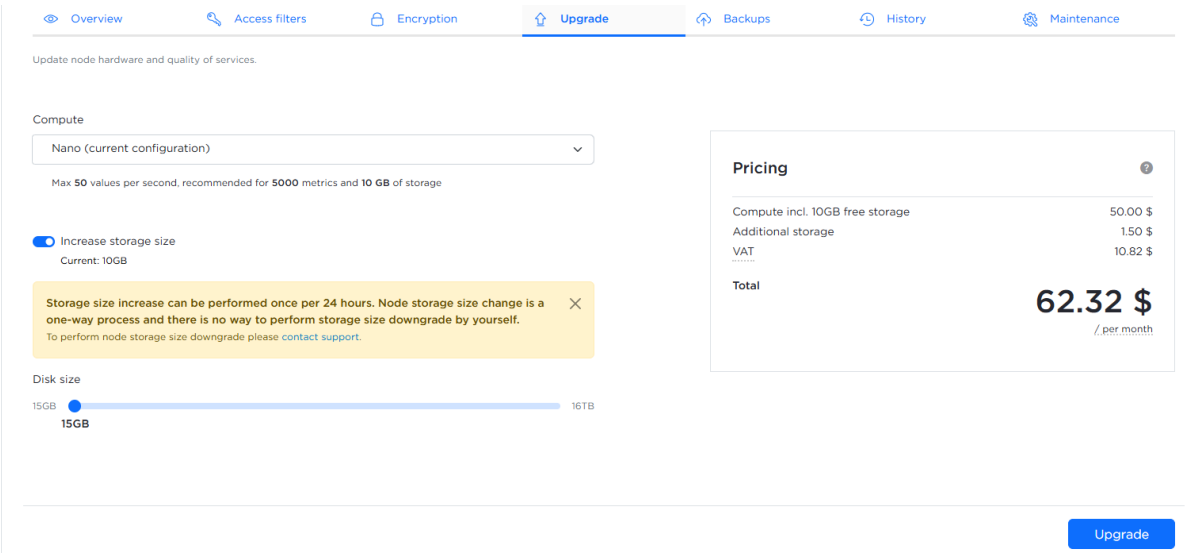
1. Haga clic en el menú desplegable *Compute*.



2. Seleccione el nivel de configuración deseado.
3. Haga clic en el botón *Upgrade*.

Para aumentar el tamaño del disco:

1. Haga clic en el botón de opción *Increase storage size*.
2. Ajuste el control deslizante *Disk size* al valor deseado.



3. Haga clic en *Upgrade*.

Note:
El precio mensual total actualizado se muestra en la tabla *Pricing* antes de confirmar los cambios.

Añadir usuarios

Introducción

Los siguientes ejemplos describen cómo añadir usuarios a su organización y nodos.

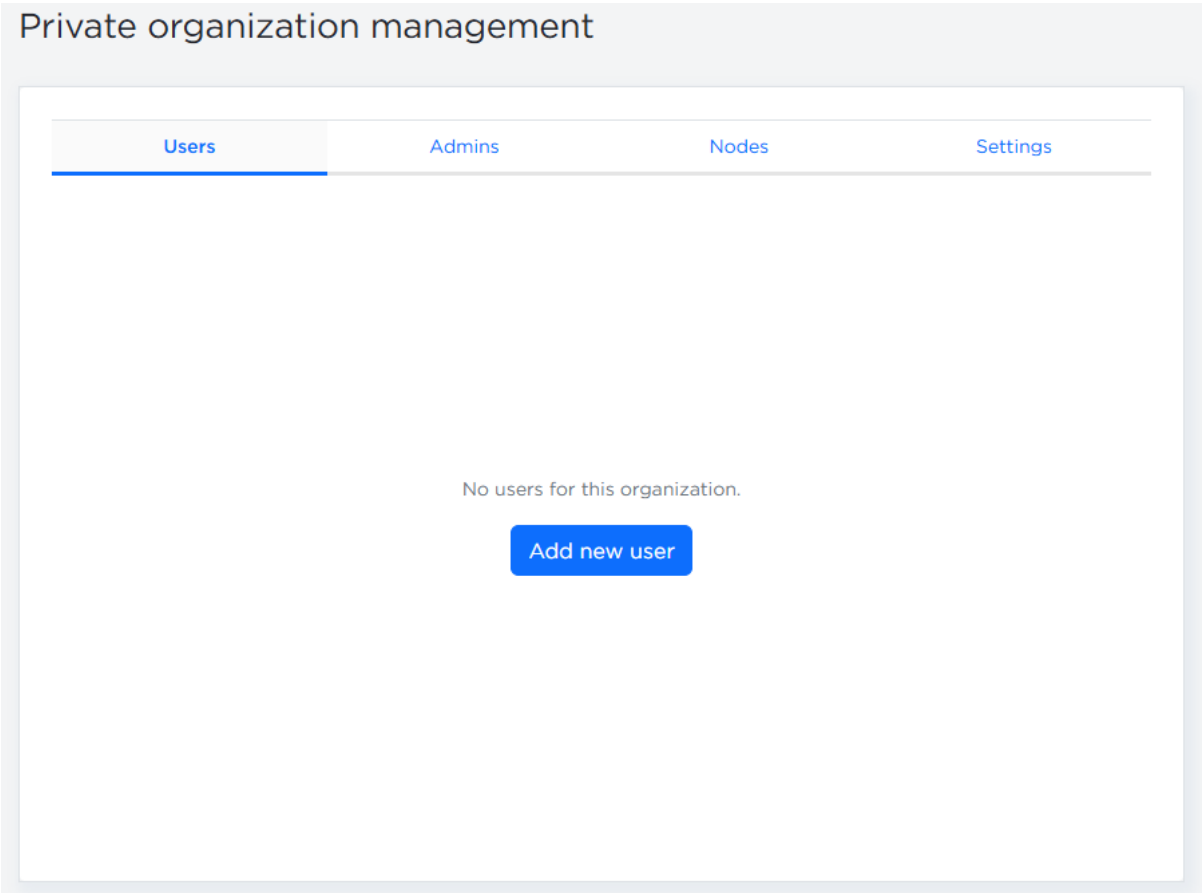
Note:
Copiar, borrar y restablecer la contraseña del frontend sólo puede ser realizado por el propietario.

Añadir usuarios a las organizaciones

De forma similar a los usuarios en Zabbix, puedes crear usuarios en Zabbix Cloud y asignarles roles para controlar lo que pueden hacer dentro de una organización y sus nodos.

Para añadir usuarios a tu organización:

1. Abre la página *Todas las organizaciones*.
2. Haz clic en el botón de configuración de la organización correspondiente para acceder a su menú de gestión.



3. En la pestaña *Usuarios* o *Administradores*, haz clic en *Añadir nuevo usuario*.
4. Introduce el correo electrónico del usuario y selecciona un rol: *Administrador* o *Usuario*.

Ten en cuenta que al seleccionar el rol *Usuario* se añade el usuario a la organización, pero no se le concede acceso a ningún nodo.

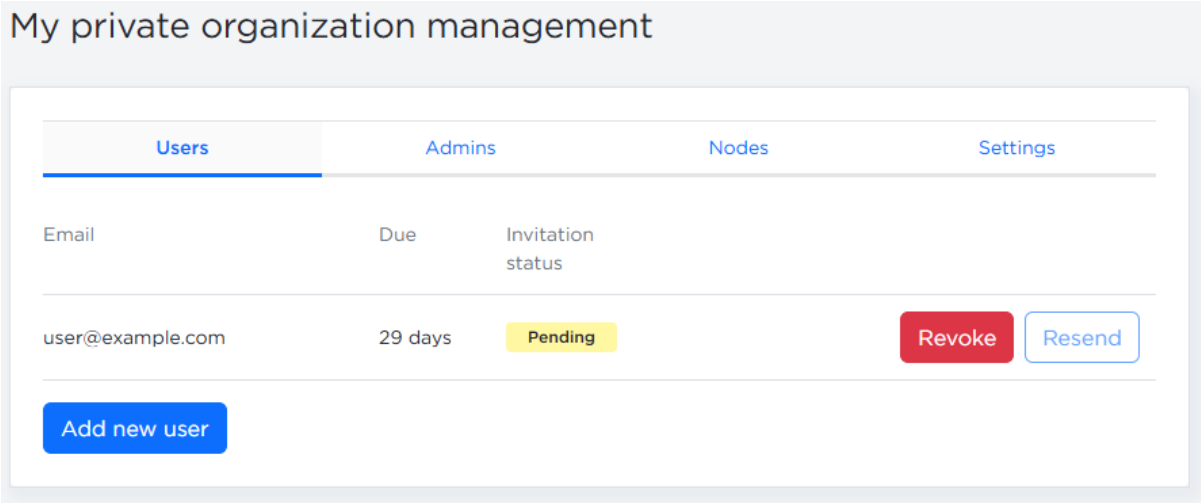
Cada rol otorga un nivel de acceso diferente:

- **Usuario**: puede acceder a los nodos a los que está asignado y realizar tareas de configuración que no generan cargos, por ejemplo:

  - Gestionar filtros de acceso
  - Configurar cifrado
  - Establecer periodos de retención de histórico/tendencias
  - Programar mantenimientos
- **Administrador**: tiene todos los permisos de Usuario y además puede realizar acciones que pueden generar cargos, por ejemplo:
  - Crear y restaurar copias de seguridad
  - Actualizar o degradar los recursos del nodo (nivel de cómputo y tamaño de disco)
  - Cancelar suscripciones
- **Propietario**: tiene control total sobre la organización. Además de todos los permisos de Administrador, el Propietario puede:
  - Crear y eliminar nodos
  - Añadir o eliminar Usuarios y Administradores
  - Renombrar o eliminar la organización
  - Gestionar los detalles de facturación

Attention:
El método de pago del Propietario se carga por todos los servicios. Si se requiere confirmación bancaria, el Propietario debe autorizar la transacción, incluso si fue iniciada por un Administrador.

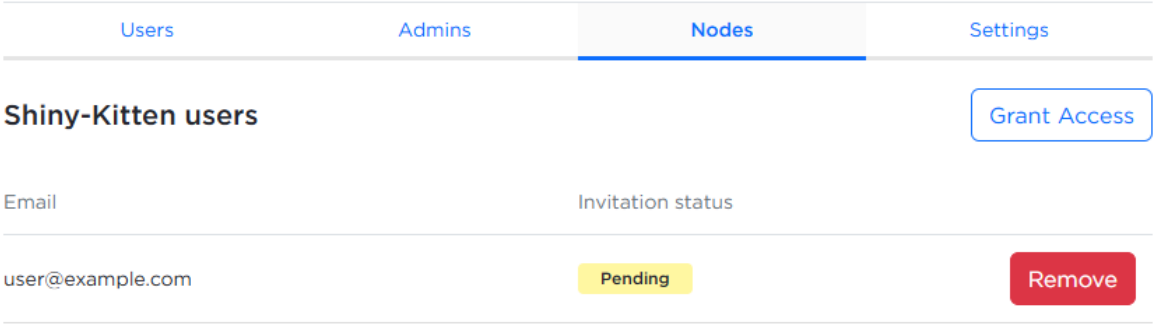
El usuario invitado se añadirá con el estado *Pendiente*. Puedes revocar la invitación o reenviarla (el reenvío está disponible una vez cada 10 minutos; el botón se desactivará si se alcanza este límite).



Añadir usuarios a los nodos

Para añadir el usuario al nodo:

1. Vaya a la pestaña *Nodos*.
2. Haga clic en *Conceder acceso*.
3. Seleccione un usuario añadido a la organización del desplegable.
4. Haga clic en *Conceder acceso*.



Alternativamente, puede conceder acceso desde la ventana de *Configuración del nodo*:

1. Haga clic en el botón de configuración  .
2. Seleccione *Añadir usuario*.
3. Elija un usuario y haga clic en *Conceder acceso*.

Los usuarios invitados deben aceptar la invitación creando una cuenta de Zabbix Cloud utilizando la misma dirección de correo electrónico a la que se envió la invitación. Si el usuario ya tiene una cuenta de Zabbix Cloud con ese correo electrónico, debe iniciar sesión y confirmar la invitación. Las invitaciones están vinculadas a una dirección de correo electrónico específica y no pueden ser aceptadas desde una cuenta diferente.

Diferencias clave de Zabbix Cloud

Introducción

Zabbix Cloud es una plataforma basada en la nube que permite a sus usuarios ejecutar instancias de Zabbix autónomas en la nube (nodos en la nube).

En comparación con una implementación de Zabbix local convencional:

- No se requiere instalación ni mantenimiento.
- Los nodos en la nube son administrados por completo por Zabbix.
- Los nodos en la nube pueden iniciarse en cuestión de minutos.

Este servicio se ofrece bajo un modelo de pago por uso (hay disponible una **prueba gratuita**).

Tabla comparativa Aunque la mayoría de las funcionalidades se solapan entre las instalaciones de Zabbix locales y los nodos en la nube, también existen diferencias.
A continuación se proporciona una tabla comparativa resumida.
En las secciones siguientes se describen detalladamente algunos elementos seleccionados.

Característica / Aspecto	Zabbix Cloud	Zabbix local
Ciclo de lanzamiento	En consonancia con las versiones LTS	Usted elige la versión de Zabbix (estable, beta, compilaciones personalizadas)
Facilidad de uso	Plataforma lista para usar, completamente administrada por Zabbix	Requiere instalación, configuración y mantenimiento
Aplicación de parches de vulnerabilidades	Aplicación automática de parches del sistema operativo y de Zabbix	Aplicación manual de parches por parte del usuario
Certificado HTTPS	Certificado válido listo para usar (sin Let's Encrypt)	Autofirmado de forma predeterminada (se puede configurar Let's Encrypt u otra CA)
Rendimiento de la base de datos (INSERT/UPDATE/SELECT)	Particionado automático para obtener la máxima velocidad; no se ejecutan DELETE de larga duración	Particionado y limpieza manuales (son posibles transacciones DELETE prolongadas)
Ajuste del rendimiento	Preajustado por el equipo de Zabbix Cloud (sin registros de cambios)	Visibilidad y control completos sobre los parámetros de ajuste
Control del firewall	Administrado por el equipo de Zabbix Cloud; los puertos de la GUI y del trapper no están expuestos públicamente	Control completo de las reglas del firewall y de la exposición de puertos
Tráfico saliente en el puerto 25	Bloqueado de forma predeterminada para evitar spam y abusos; utilice en su lugar el puerto SMTP 587 o 465	Zabbix no lo restringe; depende de su red y proveedor de alojamiento
Aislamiento y protección de la base de datos	Altamente aislada, sin acceso directo a la base de datos ni por SSH	Depende de la seguridad de su red y host
Alta disponibilidad (HA)	Se ejecuta en una plataforma en la nube de alta disponibilidad, se pueden utilizar proxies de Zabbix para una mayor resiliencia de la recopilación de datos	HA administrada por el usuario
Traps SNMP	Solo mediante un proxy dedicado (sin receptor de traps con HA ni balanceo de carga)	Directamente o mediante un proxy; es posible utilizar HA y balanceo de carga
Sondeo SNMP	Difícil sin proxies (cada dispositivo necesita NAT/puerto personalizado)	Sondeo SNMP nativo; el proxy es opcional

Característica / Aspecto	Zabbix Cloud	Zabbix local
Configuración de retención	Historial/Tendencias/Auditoría únicamente mediante la interfaz de usuario (sin API ni archivo de configuración)	Configurable en el archivo de configuración del server de Zabbix o mediante la API
Scripts personalizados	No se admiten `AlertScriptsPath`, `ExternalScripts`, módulos del frontend ni módulos de la comunidad	Compatibilidad completa (ruta de scripts, módulos, integraciones)
Supervisión ODBC	Solo PostgreSQL (controlador `{postgresql}`); el plugin de MariaDB está presente como marcador de posición	ODBC para PostgreSQL, MySQL, Oracle, etc.; totalmente configurable
Limitación de llamadas ODBC	No es posible (solo `StartODBCPollers=1`); las consultas de sincronización pesadas bloquean otras tareas	Cantidad y programación de pollers totalmente ajustables
Carga de certificados SAML	No compatible	Compatible mediante la interfaz de usuario o la API
Informes programados	Debe crear su propio medio Email (los medios de tipo script no son compatibles)	Admite medios de tipo script y Email listos para usar
Interfaz del host para comprobaciones activas	Dirección IP asignada por la plataforma (creada automáticamente)	Interfaces administradas por usted; las direcciones IP están bajo su control

Diferencias de características Trampas SNMP

Las trampas SNMP solo son compatibles a través de un proxy Zabbix dedicado. Si se requiere la monitorización de trampas SNMP, no se puede utilizar el balanceo de carga automatizado ni la alta disponibilidad para el proxy porque las trampas SNMP deben enviarse a una única dirección IP.

Sondeo SNMP

Sin proxies, el sondeo SNMP requiere que cada dispositivo esté expuesto a través de NAT y puertos personalizados.

Configuración de retención

Los períodos de retención de historial, tendencias y registro de auditoría solo se pueden configurar a través de la interfaz web en la nube. Estas configuraciones no se pueden modificar a través de `zabbix_server.conf` ni de la API.

No se admiten anulaciones manuales de historial por item (la partición se controla globalmente). Las URL de los nodos en la nube no aceptan los mismos parámetros de consulta que las instalaciones on-premise.

Scripts personalizados

Los siguientes tipos de scripts personalizados no son compatibles con Zabbix Cloud:

- Scripts de alerta (`AlertScriptsPath`)
- Scripts externos (`ExternalScripts`)
- Scripts de frontend en Zabbix Server

No se pueden instalar módulos de frontend desarrollados por la comunidad.

Monitorización ODBC

Zabbix Cloud admite la monitorización ODBC solo para PostgreSQL. Utilice la plantilla ODBC oficial de Zabbix y defina la cadena de conexión en la plantilla utilizando:

```
Driver={postgresql}
```

También se instala un plugin de MariaDB, pero actualmente no es funcional. Si se utiliza, defina:

```
Driver={mysql}
```

Attention:
No existe una configuración funcional conocida para Oracle MySQL 8.0 en Zabbix Cloud. Si bien las consultas simples (como `SELECT 1`) pueden tener éxito, las consultas más complejas devuelven `SQL_ERROR`.

 No es posible limitar las llamadas ODBC (`StartODBCPollers=1` solamente). Los informes síncronos pesados pueden afectar el rendimiento, y solo se puede ejecutar un único `SELECT` a la vez.

Acceso a la infraestructura

Zabbix Cloud no proporciona acceso SSH a los nodos subyacentes, ni permite conexiones directas a la base de datos (por ejemplo, al puerto 3306). Todas las actividades de configuración, monitoreo y solución de problemas deben realizarse a través de la interfaz de usuario o la API de la nube, asegurando que las capas del sistema operativo y la base de datos permanezcan aisladas y seguras.

Tráfico saliente en el puerto 25

Para evitar el spam y el uso indebido, las conexiones salientes desde los nodos de Zabbix Cloud al puerto TCP 25 están bloqueadas de forma predeterminada. Esto afecta a:

- Los tipos de medio Email que envían mensajes a través de un servidor SMTP en el puerto 25. Dado que 25 es el *puerto del servidor SMTP* predeterminado para un nuevo tipo de medio Email, cámbielo a 587 (con *Seguridad de la conexión* configurada como STARTTLS) o a 465 (SSL/TLS), según lo que admita su servidor SMTP. El proveedor de correo electrónico *Office365 relay* no se puede utilizar, ya que se conecta al endpoint de relay de Microsoft en el puerto 25.
- Los informes programados, que requieren un tipo de medio Email.
- Las comprobaciones que el propio nodo de la nube realiza en el puerto 25, como `net.tcp.service[smtp]`, comprobaciones simples o comprobaciones SMTP en las reglas de detección de red. Estas informan de que el servicio SMTP está inactivo incluso cuando está en ejecución; realice dichas comprobaciones mediante un proxy de Zabbix o un agent de Zabbix en su propia red.

El tipo de medio *Cloud Email* predeterminado no se ve afectado.

Interfaz de comprobación activa

Las comprobaciones activas en la nube crean automáticamente una interfaz de host con una IP no relacionada con su red local. De forma predeterminada, esta IP puede ser externa. Para mantener la coherencia, puede eliminar o ajustar manualmente la interfaz de host después de su creación.

Carga de certificado SAML

La autenticación SAML no es compatible porque pocos proveedores aceptan mensajes SAML sin cifrar o sin firmar.

Informes programados

El tipo de medio *Cloud Email* predeterminado utiliza un transporte mediante script y no se puede usar para informes programados. Para enviar informes por correo electrónico:

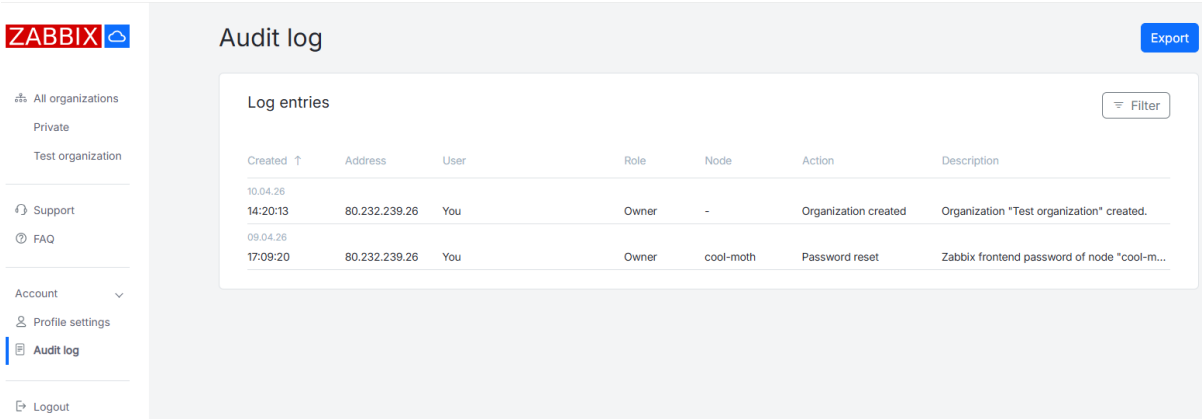
1. Cree un nuevo tipo de medio de correo electrónico.
2. Asigne el medio de correo electrónico a su usuario en *Configuración del usuario* > *Medios*.
3. Seleccione ese tipo de medio al configurar los informes programados.

Attention:
No utilice el puerto 25 como *puerto del server SMTP* para este tipo de medio; consulte Tráfico saliente en el puerto 25.

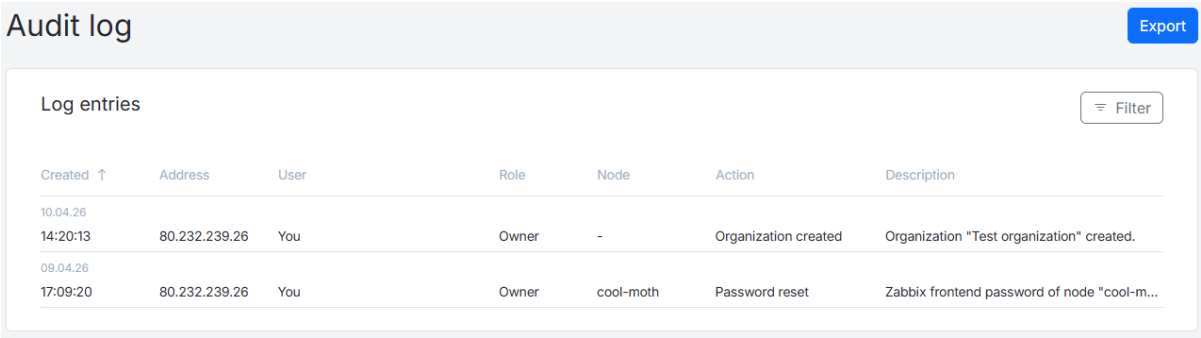
Registro de auditoría

Descripción general En Zabbix Cloud, el registro de auditoría registra la actividad de usuarios y del sistema.

Después de iniciar sesión en una cuenta de Zabbix Cloud, vaya a *Account* > *Audit log* para ver los registros del registro de auditoría. Las entradas del registro de auditoría disponibles para un usuario dependen del rol de organización del usuario.



Visualización de registros del log de auditoría Los registros del log de auditoría se muestran en una tabla.



La tabla contiene la siguiente información:

Column	Description
Created	Fecha y hora en que se creó la entrada del log. La hora mostrada corresponde a la zona horaria del usuario seleccionada en la lista desplegable *User timezone* en *Account > Profile settings*.
User	Nombre del usuario de la organización que realizó la acción.
Role	El rol del usuario que realizó la acción.
Node	Nombre del nodo relacionado con la acción.
Action	Nombre de la acción realizada.
Description	Descripción de la acción realizada.

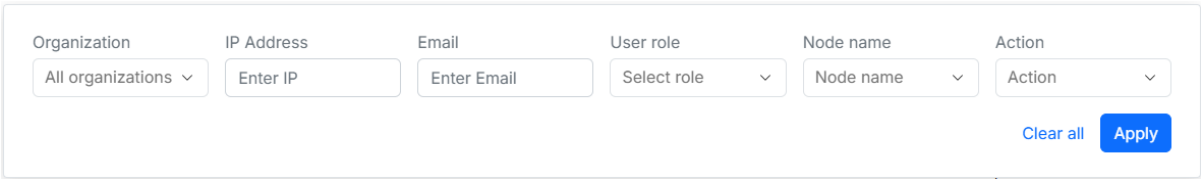
Se puede hacer clic en los nombres de las columnas para ordenar los registros en orden ascendente o descendente.

Los registros del log de auditoría filtrados se pueden exportar en formato `.csv` mediante el botón *Export*.

Uso de filtros

Utilice el botón *Filter* para abrir el filtro.

Para mostrar los resultados filtrados, haga clic en *Apply* después de seleccionar uno o más parámetros de filtrado. Para borrar el filtro, haga clic en *Clear all*.



Es posible filtrar los registros por uno o más parámetros. Los siguientes parámetros de filtrado están disponibles:

Parameter	Description
Organization	Seleccione todas las organizaciones o una específica.
IP Address	Especifique una dirección IP.
Email	Especifique una dirección de correo electrónico asociada al usuario.
User role	Seleccione todos los roles o uno específico: Owner, Admin o User.
Node name	Seleccione todos los nodos o uno específico.
Action	Seleccione un tipo de acción específico.

El registro de auditoría realiza un seguimiento de las siguientes acciones:

- *Nodo creado*
- *Nodo eliminado*
- *Nodo actualizado*
- *Versión del nodo actualizada*
- *Contraseña restablecida*
- *Mantenimiento actualizado*
- *Configuración de historial actualizada*
- *Fecha de expiración del nodo ampliada*
- *Configuración del proxy actualizada*

- *Copia de seguridad creada*
- *Copia de seguridad eliminada*
- *Copia de seguridad restaurada*
- *Copia de seguridad actualizada*
- *Filtro de acceso añadido*
- *Filtro de acceso actualizado*
- *Filtro de acceso eliminado*
- *Cifrado habilitado*
- *Cifrado deshabilitado*
- *Suscripción cancelada*
- *Suscripción renovada*
- *Pago de la suscripción fallido*
- *El pago requiere confirmación*
- *Organización creada*
- *Organización eliminada*
- *Admin invitado*
- *Usuario invitado*
- *Admin se unió*
- *Usuario se unió*
- *Admin eliminado de la organización*
- *Usuario eliminado de la organización*
- *Nodo añadido a la organización*
- *Nodo eliminado de la organización*
- *Acceso al nodo concedido*
- *Acceso al nodo revocado*
- *Configuración del usuario actualizada*
- *Pago creado*
- *Pago completado*
- *Pago fallido*
- *Cargo de IVA aplicado*
- *Validación del IVA fallida*
- *Información de facturación actualizada*
- *Método de pago actualizado*
- *Método de pago eliminado*
- *Nodo prepago creado*
- *Nodo prepago actualizado*
- *Incidente iniciado*
- *Incidente finalizado*

Configurar un proxy de Zabbix Cloud

Descripción general Esta página describe cómo configurar un proxy de Zabbix implementado en Zabbix Cloud para recopilar datos de monitorización de hosts en un entorno remoto.

El procedimiento está destinado principalmente a un server y un proxy de Zabbix creados en Zabbix Cloud.

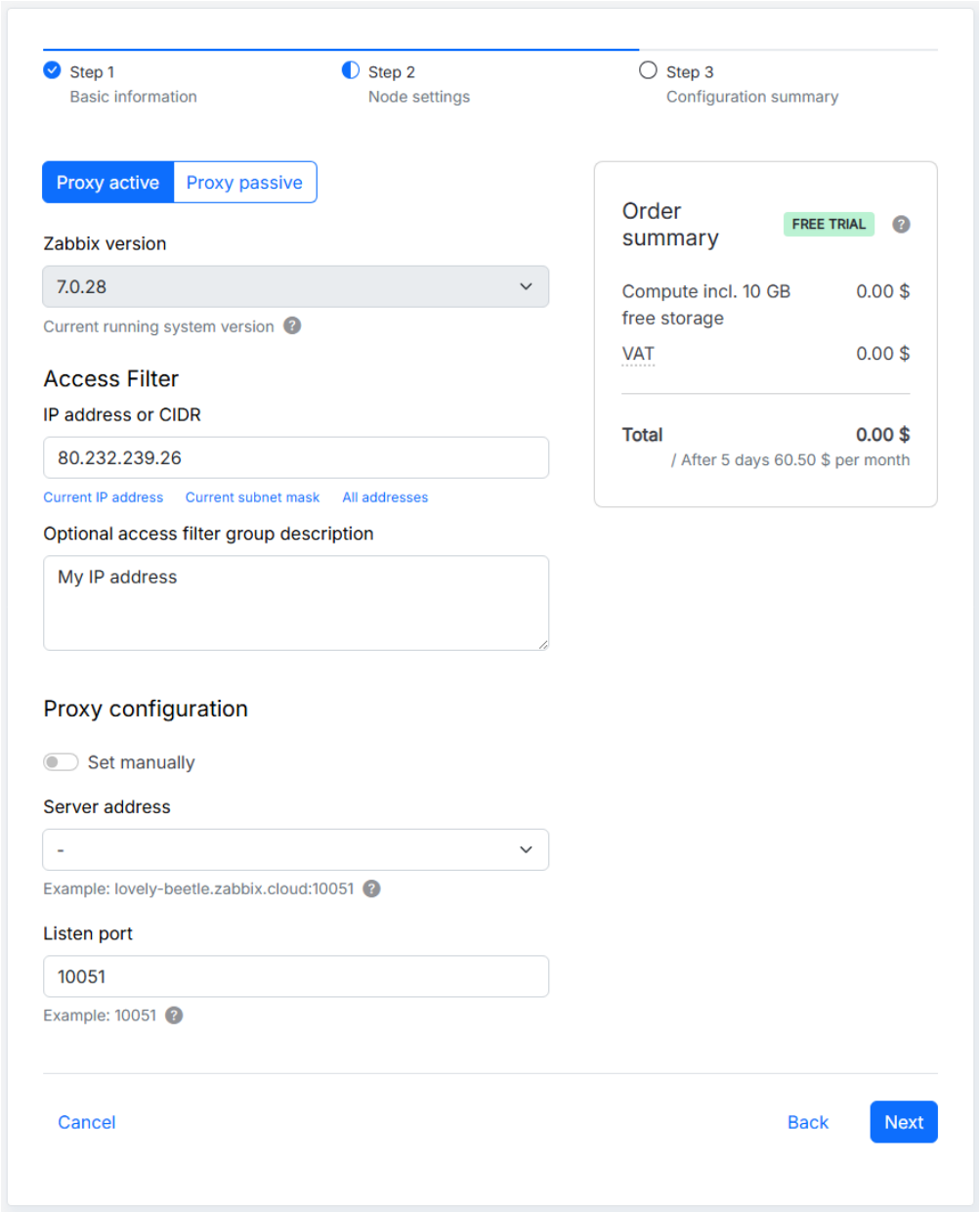
El procedimiento también describe la configuración necesaria cuando el server de Zabbix no está alojado en Zabbix Cloud.

Requisitos previos Un server de Zabbix está instalado, configurado y en ejecución, o un nodo de server de Zabbix está en ejecución en Zabbix Cloud.

Crear un nodo proxy Cree un nodo proxy de Zabbix Cloud.

Durante la creación del nodo:

1. Seleccione *Proxy*.
2. Seleccione *Proxy active* o *Proxy passive* según cómo deba comunicarse el proxy con el server de Zabbix.
3. Configure la dirección IP del nodo o los filtros de acceso CIDR (puede hacerlo después de crear el nodo).
4. Especifique el server de Zabbix al que se conectará el proxy.



5. Complete el proceso de creación del nodo.

Para obtener más información sobre cómo crear un nodo proxy, consulte Implementar Zabbix en la nube.

Configurar filtros de acceso Zabbix Cloud utiliza filtros de acceso para controlar qué direcciones IP o subredes pueden conectarse a un nodo de la nube.

Durante la inicialización del nodo, se crea automáticamente un filtro de acceso para la dirección IP utilizada para acceder al nodo. Esto no incluye necesariamente las direcciones de los hosts monitorizados ni de otros componentes de Zabbix. Añada manualmente las direcciones necesarias.

Consulte Configuración del nodo — Filtros de acceso para obtener más información.

Nodo proxy

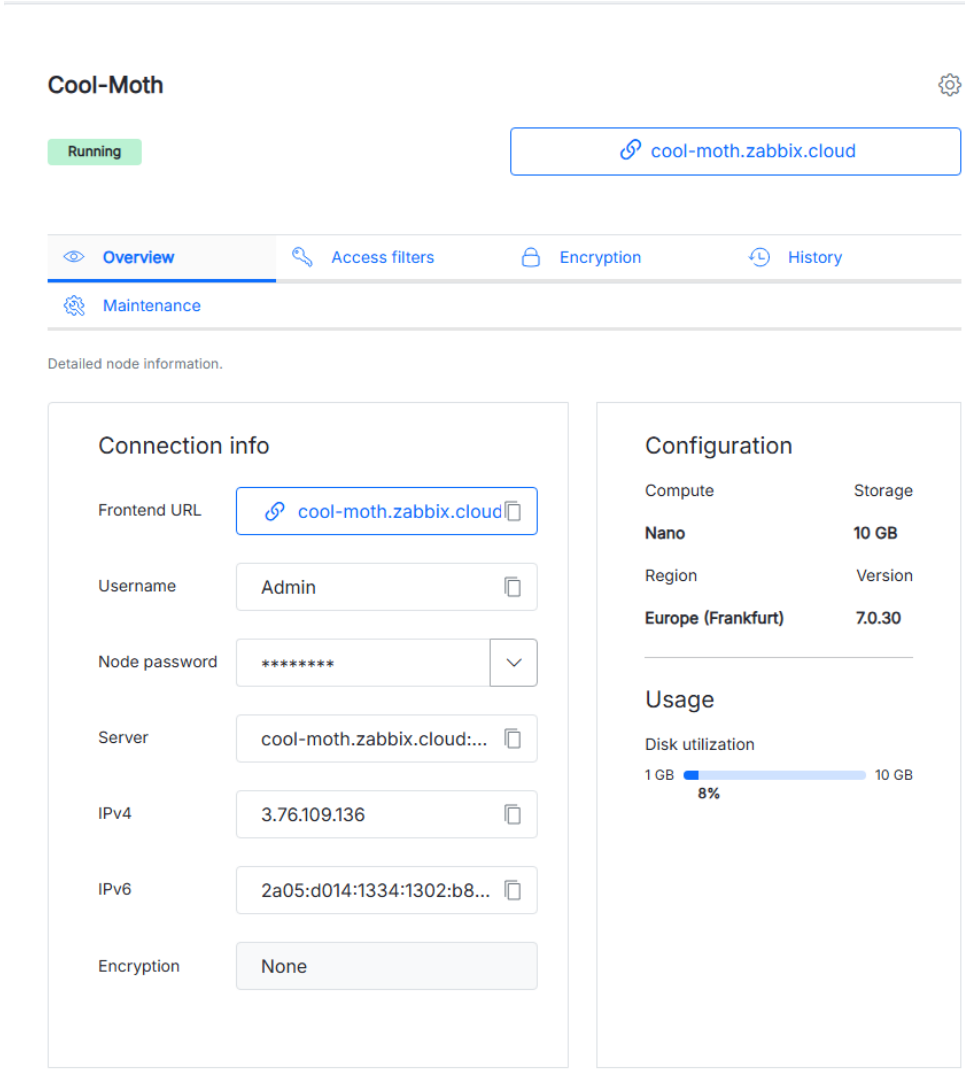
Añada las direcciones IP o subredes desde las que se esperan conexiones al proxy.

Si los hosts monitorizados utilizan comprobaciones activas del agent, los agents inician las conexiones al proxy, por lo que la dirección IP o subred de los hosts monitorizados debe estar permitida por los filtros de acceso del nodo proxy.

Si se estableció *Proxy Passive* como modo del proxy, el server de Zabbix se conecta al proxy, por lo que la dirección IP o subred del server de Zabbix debe estar permitida por los filtros de acceso del nodo proxy.

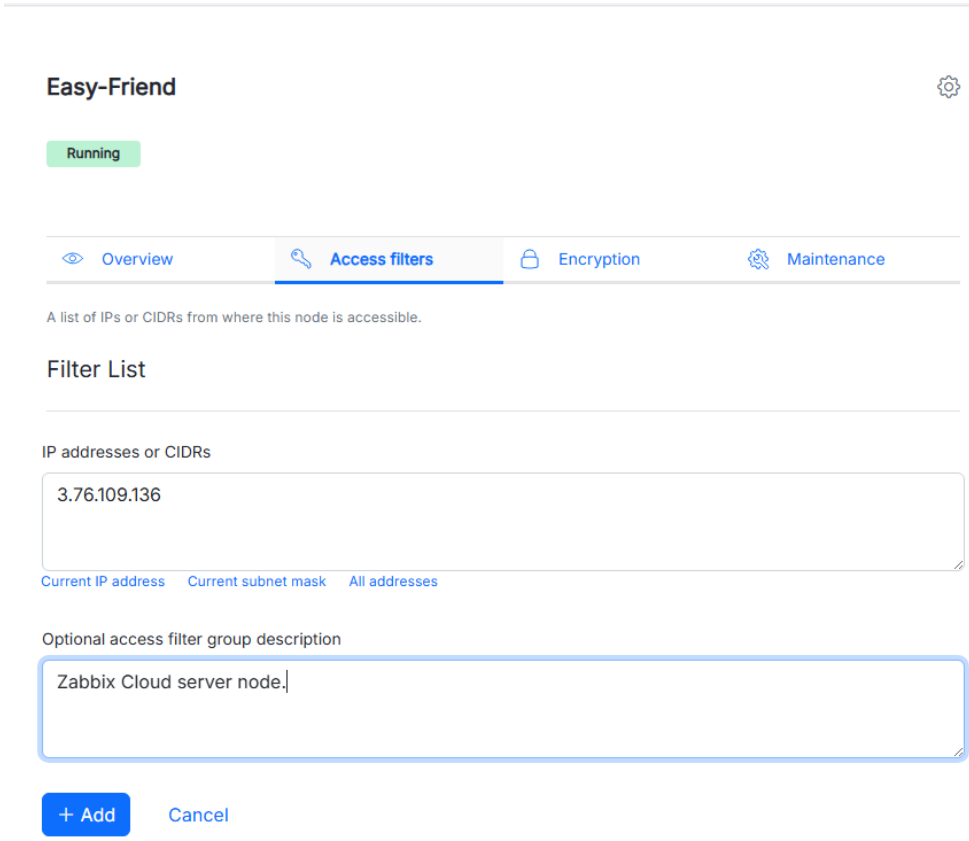
Si se seleccionó *Proxy passive* y se utiliza un nodo server:

1. Vaya a la pestaña *Overview* de su nodo server y copie la dirección IPv4 o IPv6 del nodo.

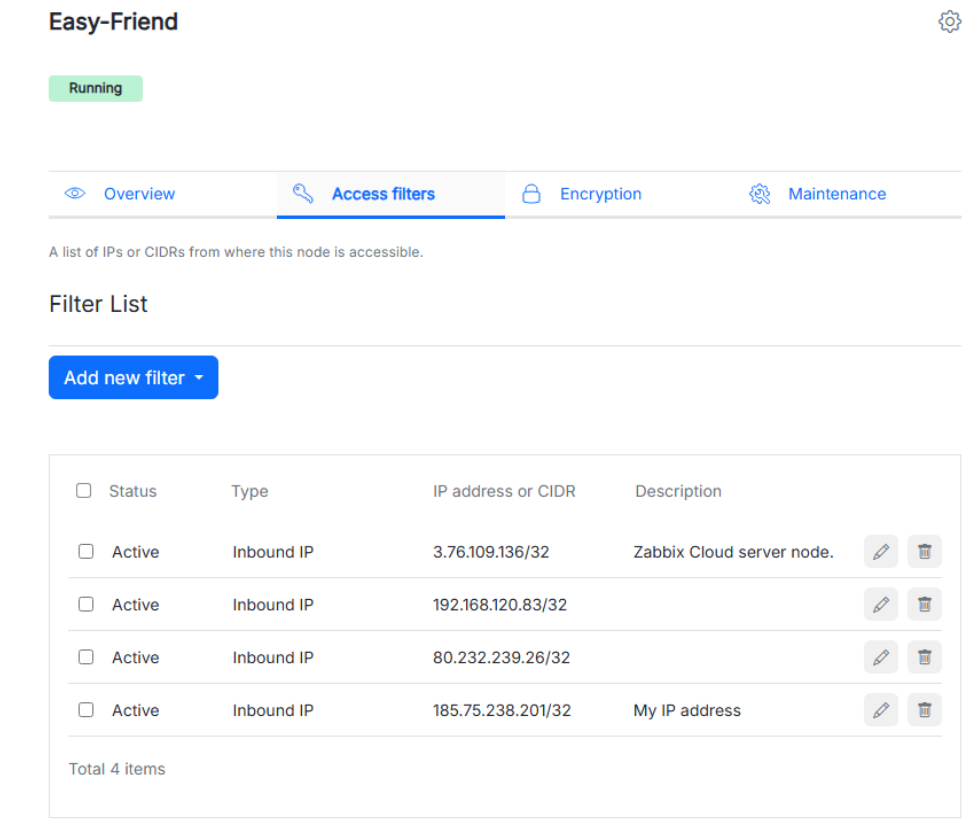


2. Vaya a la pestaña *Access filters* de su nodo proxy y haga clic en *Add new filter* > *Add*.

3. Pegue la dirección copiada en el campo *IP addresses or CIDRs* y añada una descripción en el campo *Optional access filter group description*.



4. Haga clic en *Add*.



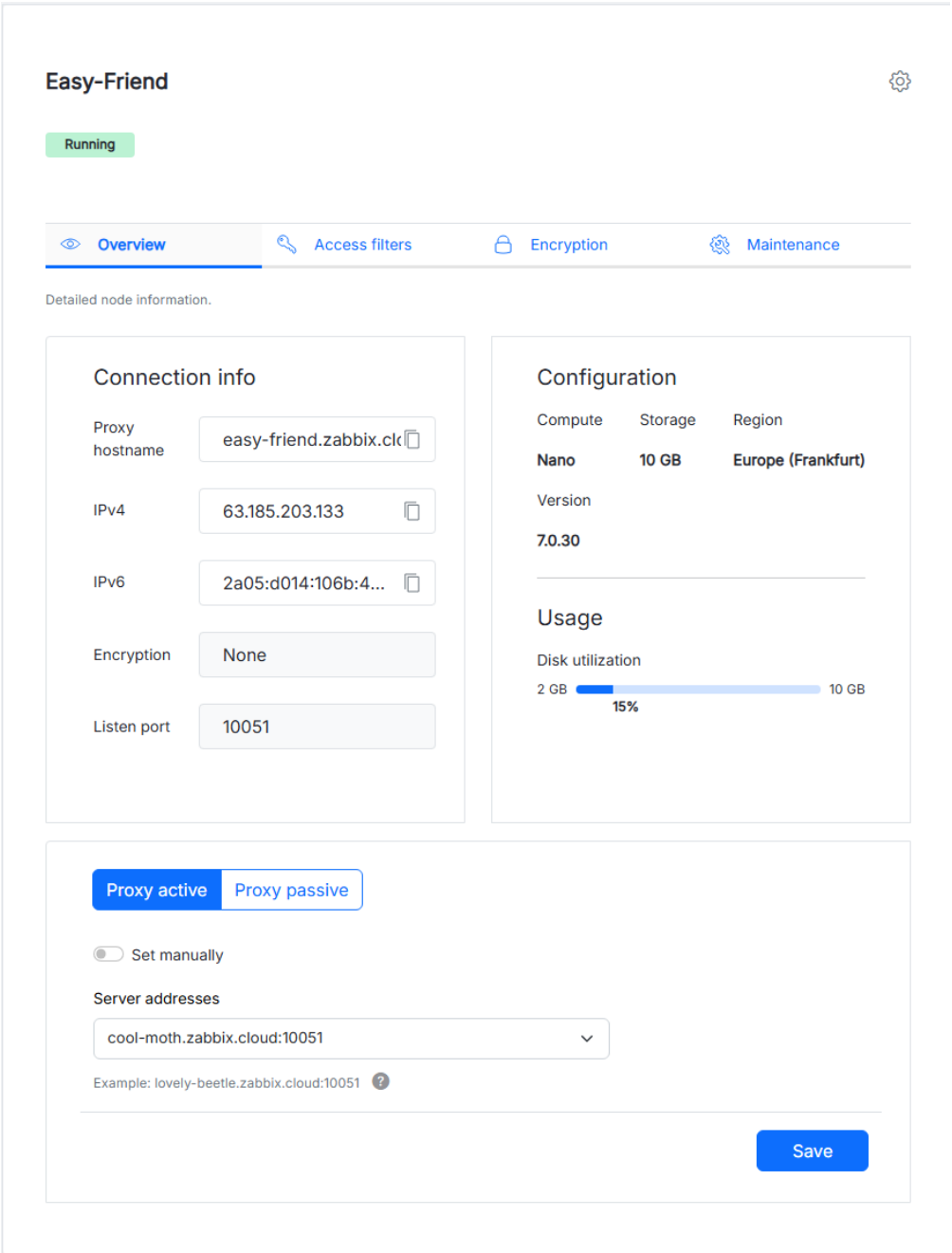
Repita los pasos 2-4 para la dirección IP de un agent de Zabbix que utilice comprobaciones activas.

Nodo del server

Si el server de Zabbix está alojado en Zabbix Cloud, configure también los filtros de acceso en el nodo del server.

Para un proxy en modo activo, añada la dirección IP o el CIDR del proxy al nodo del server en los filtros de acceso, ya que el proxy inicia la conexión con el server.

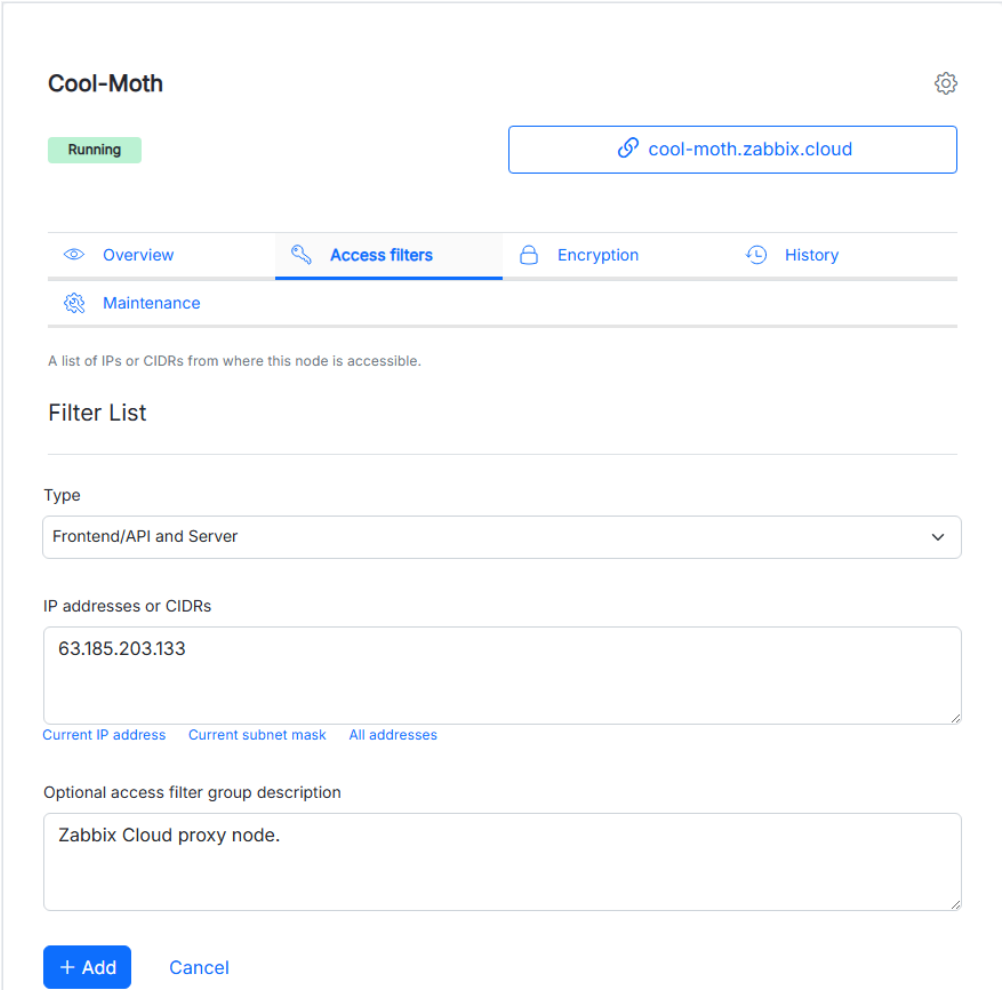
1. Vaya a la pestaña *Overview* de su nodo proxy y copie la dirección IPv4 o IPv6 del nodo.



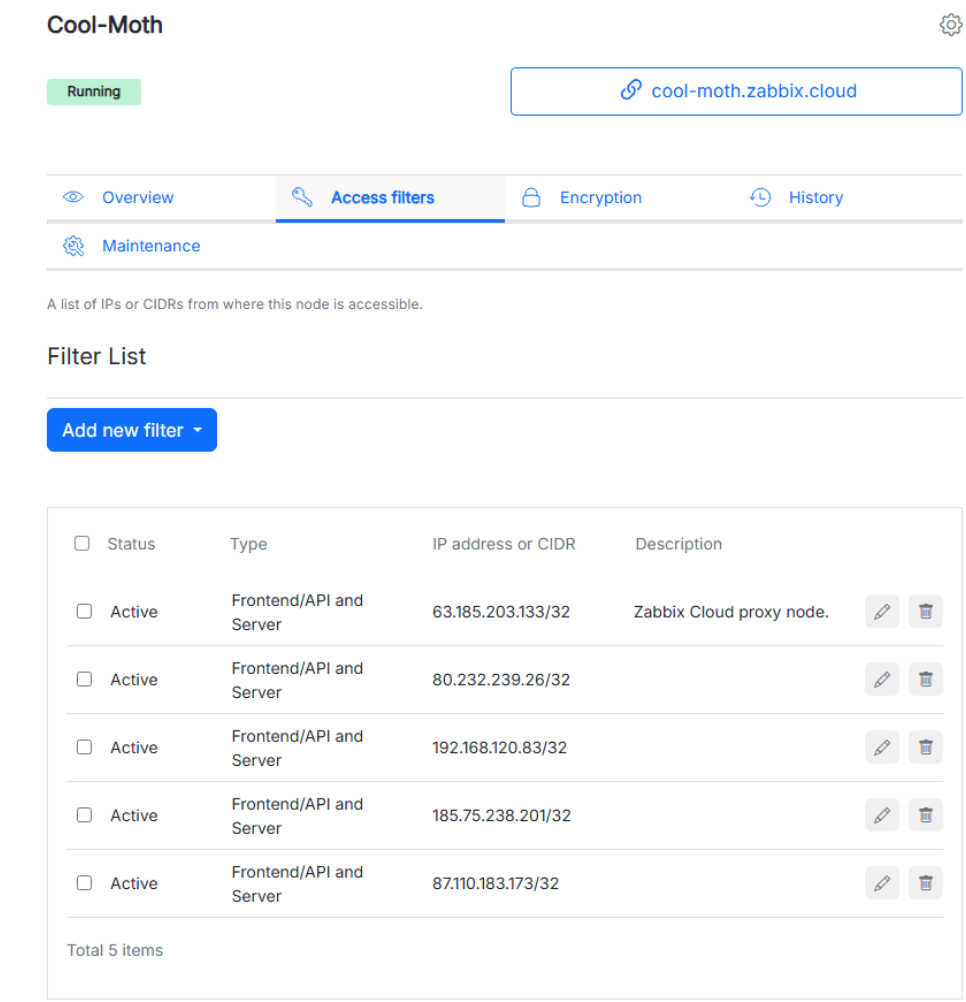
2. Vaya a la pestaña *Access filters* del nodo del server y haga clic en *Add new filter* > *Add*.

3. Pegue la dirección copiada en el campo *IP addresses or CIDRs* y añada una descripción en el campo *Optional access filter group description*.

4. Seleccione el tipo de filtro adecuado.



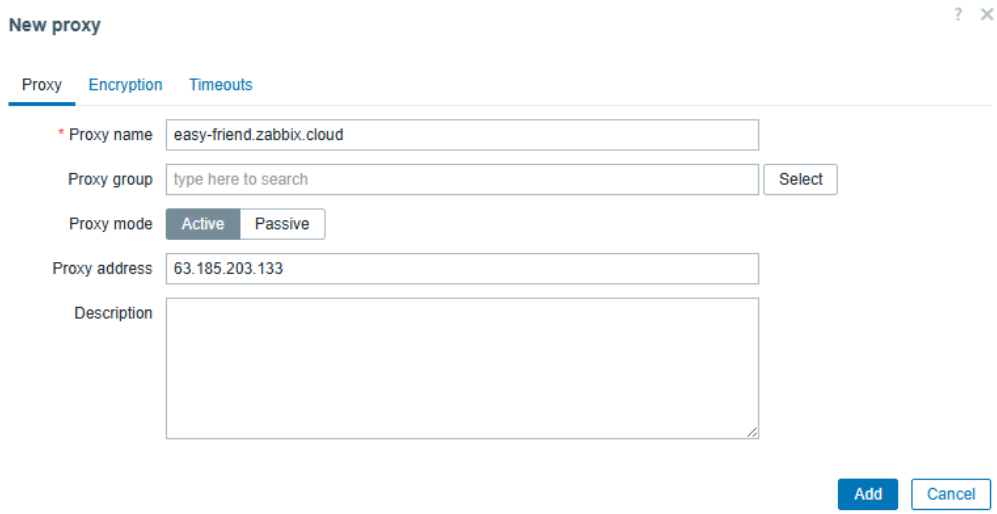
5. Haga clic en *Add*.



Configurar el proxy El proxy debe configurarse en el frontend de Zabbix para comunicarse con el server de Zabbix.

Abra *Administración* > *Proxies* en el frontend de Zabbix, haga clic en *Crear proxy* y configure el proxy según sea necesario:

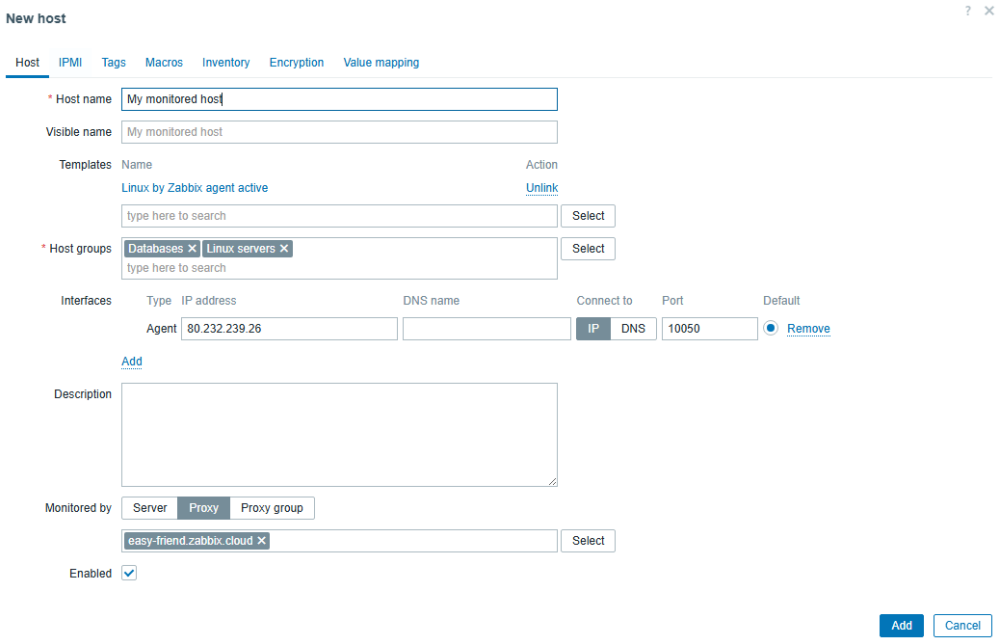
1. Añada el nombre del proxy en el campo *Nombre del proxy*. Debe coincidir con el nombre de host del proxy configurado para el proxy. Introdúzcalo o cópielo de la pestaña *Descripción general* de su proxy.
2. Seleccione el modo del proxy correspondiente al seleccionado durante la creación del proxy.
3. Para un proxy activo, introduzca la dirección IP o el nombre DNS del proxy.
4. Haga clic en *Añadir*.



Para obtener más información, consulte Supervisión distribuida mediante proxies.

Configurar hosts monitorizados Añada los hosts que deben monitorizarse a través del proxy.

1. Vaya a *Recopilación de datos* > *Hosts*.
2. Cree un host o abra uno existente.
3. En *Monitorizado por*, seleccione el *Proxy* y escriba el nombre del proxy para buscarlo y seleccionarlo, o pulse el botón *Seleccionar* para abrir una lista de proxies entre los que elegir.
4. Configure la interfaz del host y los demás parámetros.
5. Haga clic en *Añadir* o *Actualizar*.



Un host configurado para ser monitorizado por un proxy envía o recibe datos de monitorización a través de ese proxy, según el tipo de item y el modo del agent.

Configurar el agent de Zabbix Cuando un host supervisado utiliza Zabbix agent o Zabbix agent 2, configure el agent para que se comunique con el componente de Zabbix correspondiente.

La configuración depende de si el agent utiliza comprobaciones *pasivas* o *activas*.

Comprobaciones pasivas del agent

Con las comprobaciones pasivas, el proxy de Zabbix inicia las conexiones al agent y solicita datos de monitorización.

Abra el archivo de configuración del agent de Zabbix y actualice el parámetro `Server` para permitir conexiones desde la dirección IP del proxy de Zabbix Cloud.

Por ejemplo:

```
Server=<proxy_IP>
```

Asegúrese de que el host monitorizado permita conexiones TCP entrantes desde el proxy en el puerto del agent, que de forma predeterminada es 10050.

El proxy debe poder acceder al host monitorizado a través de la red.

Comprobaciones activas del agent

Con las comprobaciones activas, el agent inicia conexiones al proxy de Zabbix y envía los datos de monitorización recopilados.

Abra el archivo de configuración del agent de Zabbix y actualice el parámetro `ServerActive` con la dirección del proxy de Zabbix Cloud.

Por ejemplo:

`ServerActive=<proxy_IP>:<proxy_listen_port>`

Sustituya <proxy_IP> por la dirección IP del proxy de Zabbix Cloud y <proxy_listen_port> por el puerto de escucha configurado cuando se creó el nodo proxy. El puerto de escucha predeterminado es 10051.

Si se especifica un proxy en ServerActive, el agent obtiene del proxy la configuración de las comprobaciones activas en lugar de obtenerla directamente del server de Zabbix.

Asegúrese de que el host monitorizado pueda establecer una conexión TCP saliente con el proxy en el puerto de escucha configurado.

Para obtener más información sobre las comprobaciones pasivas y activas del agent, consulte Comprobaciones pasivas y activas del agent.

Servidor Zabbix externo Un proxy de Zabbix Cloud también puede comunicarse con un servidor Zabbix implementado fuera de Zabbix Cloud.

En este caso, el servidor Zabbix externo debe poder comunicarse con el proxy de nube según el modo de proxy seleccionado.

Para un proxy en modo activo, el proxy inicia la conexión con el servidor Zabbix externo. Configure el servidor Zabbix externo para aceptar conexiones del proxy y asegúrese de que el proxy pueda acceder al server a través de la red.

Para un proxy pasivo, el server de Zabbix inicia la conexión con el proxy.
Agregue la dirección IP o la subred del servidor Zabbix externo a los filtros de acceso del nodo del proxy y asegúrese de que el server pueda acceder al proxy a través de la red.

El servidor Zabbix externo no utiliza los filtros de acceso de Zabbix Cloud. En su lugar, configure las reglas de firewall y de red necesarias en el entorno externo.

Los hosts monitorizados pueden encontrarse en el mismo entorno que el servidor Zabbix externo, en otra red o en una red accesible para el proxy.

Cuando los agents utilizan comprobaciones activas, configure su parámetro ServerActive para que apunte al proxy de Zabbix Cloud. Cuando los agents utilizan comprobaciones pasivas, configure su parámetro Server para permitir conexiones desde el proxy.

Verificar la configuración Después de configurar el proxy, los agentes, los filtros de acceso y la conectividad de red, verifique que el proxy se esté comunicando con el server de Zabbix.

1. Vaya a *Administración* > *Proxies*.
2. Compruebe el *Estado* del proxy.
3. Vaya a *Monitoreo* > *Últimos datos* y compruebe que haya valores recientes de los items disponibles.

Si el proxy no se está comunicando con el server, verifique:

- el modo del proxy;
- la dirección y el puerto del server;
- los filtros de acceso del proxy;
- los filtros de acceso del server, cuando el server está alojado en Zabbix Cloud;
- la configuración del agent;
- las reglas del firewall y de enrutamiento;
- la dirección configurada en Server o ServerActive.

Aplicación móvil

Aplicación móvil

La aplicación móvil de Zabbix le permite supervisar su entorno de Zabbix desde su dispositivo móvil. Consulte los problemas, compruebe sus detalles, actualícelos y reciba notificaciones push cuando algo requiera atención.

Obtener la aplicación (próximamente)

Configurar Zabbix para la aplicación

Configurar el adaptador de puente de Zabbix Instale el adaptador de puente de Zabbix para permitir que los dispositivos móviles se conecten a Zabbix.

Añadir dispositivos Añada su dispositivo móvil en el web frontend de Zabbix para conectarlo a su Zabbix server.

Configurar notificaciones push Reciba alertas en su dispositivo móvil cuando se produzca un problema.

Más información sobre la aplicación

Descripción general de la aplicación Aprenda los conceptos básicos del uso de la aplicación móvil de Zabbix.

Acceso móvil seguro Proteja la aplicación, sus dispositivos móviles y el adaptador de puente de Zabbix, o revoque el acceso desde dispositivos perdidos o comprometidos.

Configurar Zabbix para la aplicación

Para usar la aplicación móvil de Zabbix (próximamente), primero debe:

1. Configurar el adaptador de puente de Zabbix.
2. Configurar las notificaciones push.
3. Añadir su dispositivo móvil en la interfaz web de Zabbix.

Las secciones siguientes describen cada paso.

Después de la configuración, consulte Acceso móvil seguro para mantener segura la aplicación y el adaptador de puente de Zabbix.

Configurar el adaptador de puente de Zabbix Para permitir que su Zabbix server se comunique con la aplicación móvil, debe configurar el adaptador de puente de Zabbix.

El adaptador de puente puede instalarse en el mismo sistema que su Zabbix server o en un sistema diferente mediante paquetes:

> **Attention:**
> Los pasos siguientes corresponden a Ubuntu 24.04. Para otros sistemas operativos, tome los comandos de instalación del repositorio y de los paquetes (por ejemplo, `wget`, `dpkg`, `apt`) de la página Obtener Zabbix correspondiente a su sistema operativo y utilícelos para instalar el paquete `zabbix-bridge-adapter`.

1. Inicie una nueva sesión de shell con privilegios de root:

```
sudo -s
```

2. Instale el repositorio de Zabbix:

```
wget https://repo.zabbix.com/zabbix/8.0/release/ubuntu/pool/main/z/zabbix-release/zabbix-release_latest_8.
dpkg -i zabbix-release_latest_8.0+ubuntu24.04_all.deb
apt update
```

3. Instale el adaptador de puente de Zabbix:

```
apt install zabbix-bridge-adapter
```

4. Edite el archivo de configuración de Zabbix server:

```
EnableMobileDevices=1
BridgeAdapterURL=https://bridge-adapter.example.com:10054/rpc
```

5. Reinicie Zabbix server, por ejemplo (consulte otras opciones):

```
systemctl restart zabbix-server
```

6. Copie el ID de Zabbix server desde *Informes* > *Información del sistema*; lo utilizará en el paso siguiente.

7. Edite el archivo de configuración del adaptador de puente de Zabbix:

```
ZabbixAPIURL=https://example.com/zabbix/api_jsonrpc.php
ServerId=019fb1f9-22bd-74ea-bc03-51efb8b76aba
DataDir=/var/lib/zabbix-bridge-adapter
```

> **Attention:**
> El directorio especificado en `DataDir` debe tener permisos 0700, y todos los archivos almacenados en él deben tener permisos 0600.

8. Revise y acepte el acuerdo de licencia de usuario final (EULA) del adaptador de puente de Zabbix:

```
zabbix_bridge_adapter license show
sudo -u zabbix-bridge-adapter zabbix_bridge_adapter -c /etc/zabbix/zabbix_bridge_adapter.conf license acce
```

9. Inicie el adaptador de puente de Zabbix:

```
systemctl start zabbix-bridge-adapter
```

> **Attention:**
> Al iniciarse, el adaptador de puente de Zabbix utiliza el ID del server para registrar su Zabbix server en el Zabbix Bridge alojado de forma centralizada. El estado del registro se almacena en un archivo de estado dentro de `DataDir`. Si reinstala el adaptador de puente y se pierde el archivo de estado, es posible que no pueda registrar el mismo Zabbix server. Para evitarlo, haga una copia de seguridad del archivo de estado antes de reinstalar el adaptador de puente y restáurela después.

Configurar notificaciones push Para recibir alertas en su dispositivo móvil cuando algo requiera atención en su entorno supervisado:

1. Cree un tipo de medio de notificación push (o habilite el predeterminado).
2. Cree una acción (o habilite la acción de trigger predeterminada *Informar de problemas a los administradores de Zabbix*).
3. Añada un dispositivo para añadir automáticamente el tipo de medio a su usuario (o añada medios de usuario manualmente).

Añadir dispositivos Para utilizar la aplicación móvil de Zabbix en su dispositivo, debe añadirlo en la interfaz web de Zabbix. Para obtener instrucciones, consulte:

- *Usuarios* > *Dispositivos* (como *Super admin*, añada sus propios dispositivos móviles o los de otros usuarios)
- *Configuración del usuario* > *Dispositivos* (como cualquier usuario, añada sus propios dispositivos móviles)

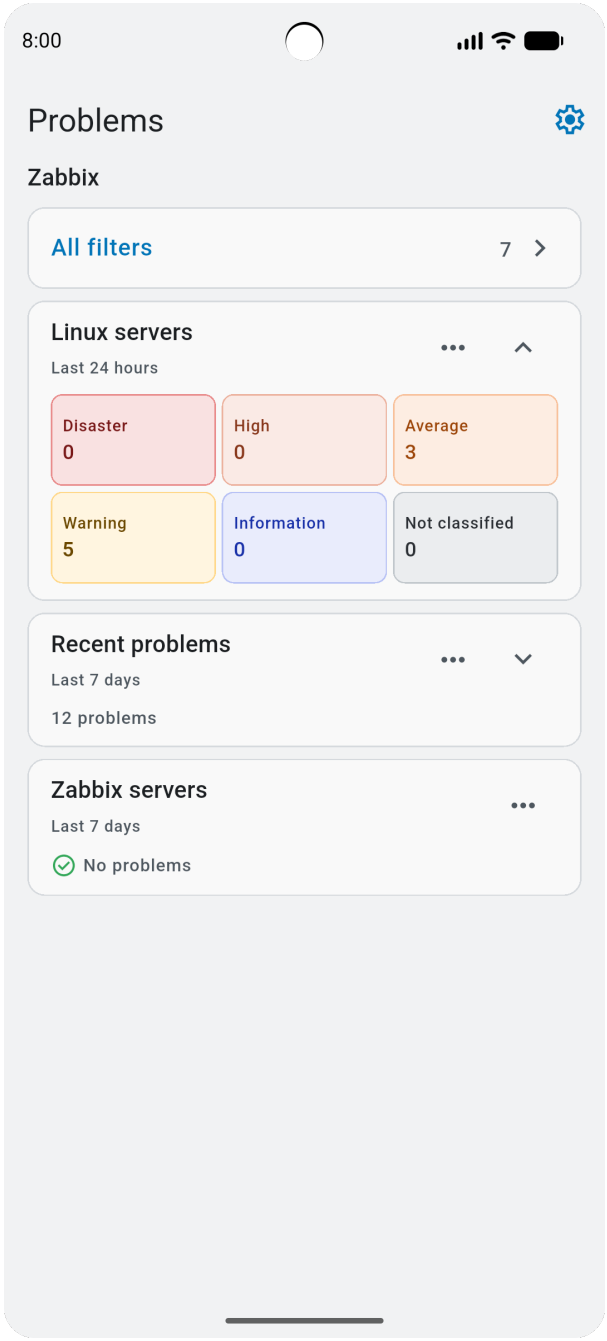
Descripción general de la aplicación

La aplicación móvil de Zabbix (próximamente) le permite supervisar su entorno de Zabbix desde su dispositivo móvil.
Vea los problemas, consulte sus detalles, actualícelos y reciba notificaciones push cuando algo requiera atención.

Esta página ofrece una descripción general de la aplicación móvil y explica cómo utilizarla.

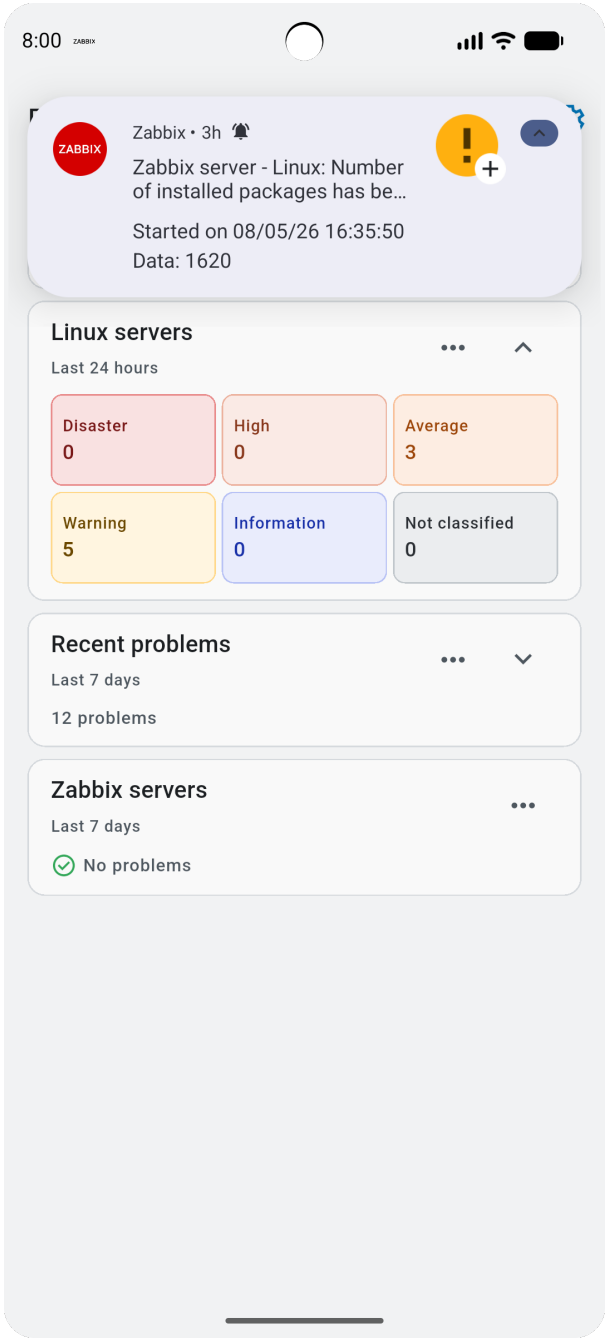
Problemas La pantalla *Problemas* muestra una descripción general de los problemas según tus filtros fijados. Contiene:

- Icono de *Configuración*: tócalo para ver la configuración de la aplicación.
- *Todos los filtros*: tócalo para ver y administrar los filtros.
- Nombre del Zabbix server (o *Zabbix*, si no se configuró durante la instalación de la interfaz web de Zabbix).
- Filtros fijados: toca su nombre o una gravedad para abrir la pantalla *Lista de problemas*. Toca el menú de tres puntos para administrar el filtro (editarlo, duplicarlo, quitarlo de los fijados o eliminarlo), o toca la flecha para expandirlo o contraerlo. Ten en cuenta que los filtros fijados se ordenan alfabéticamente y que el filtro *Problemas recientes* no se puede editar ni eliminar.



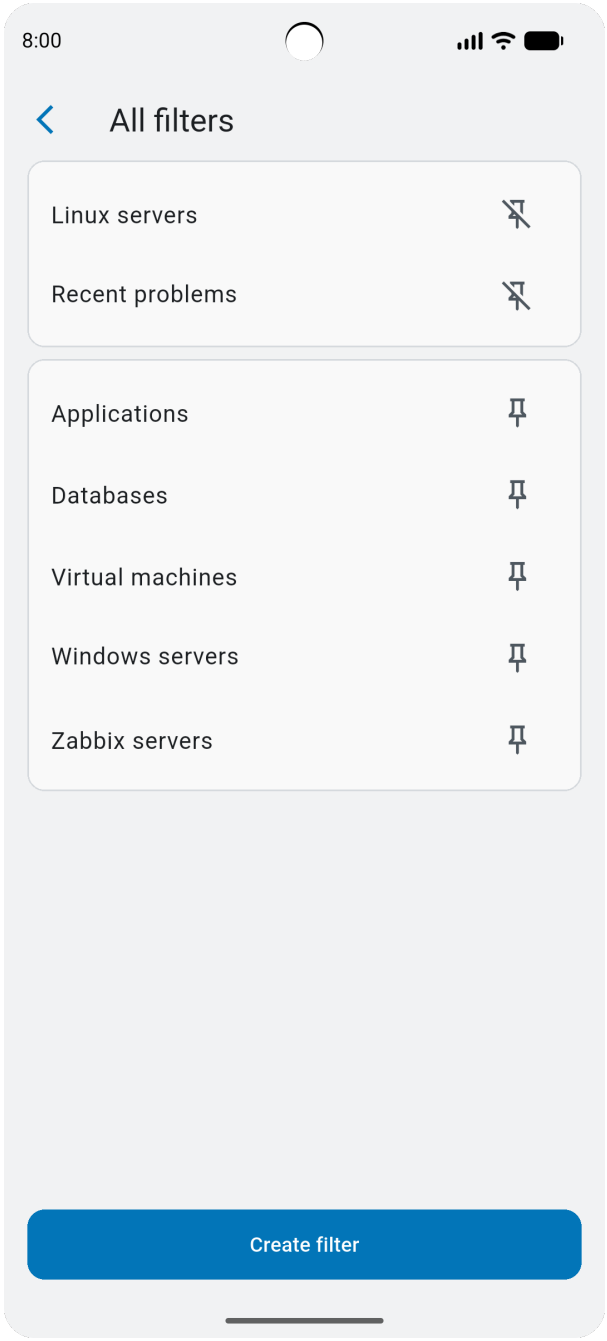
Notificaciones Cuando el server de Zabbix detecta un problema, o cuando un problema se actualiza o se resuelve, puede enviar una notificación a su dispositivo móvil. La notificación contiene el host del problema, su nombre y un mensaje generado a partir de las plantillas de mensajes configuradas para el tipo de medio de notificación push.

Toque la notificación para abrir la pantalla *Detalles del problema*.



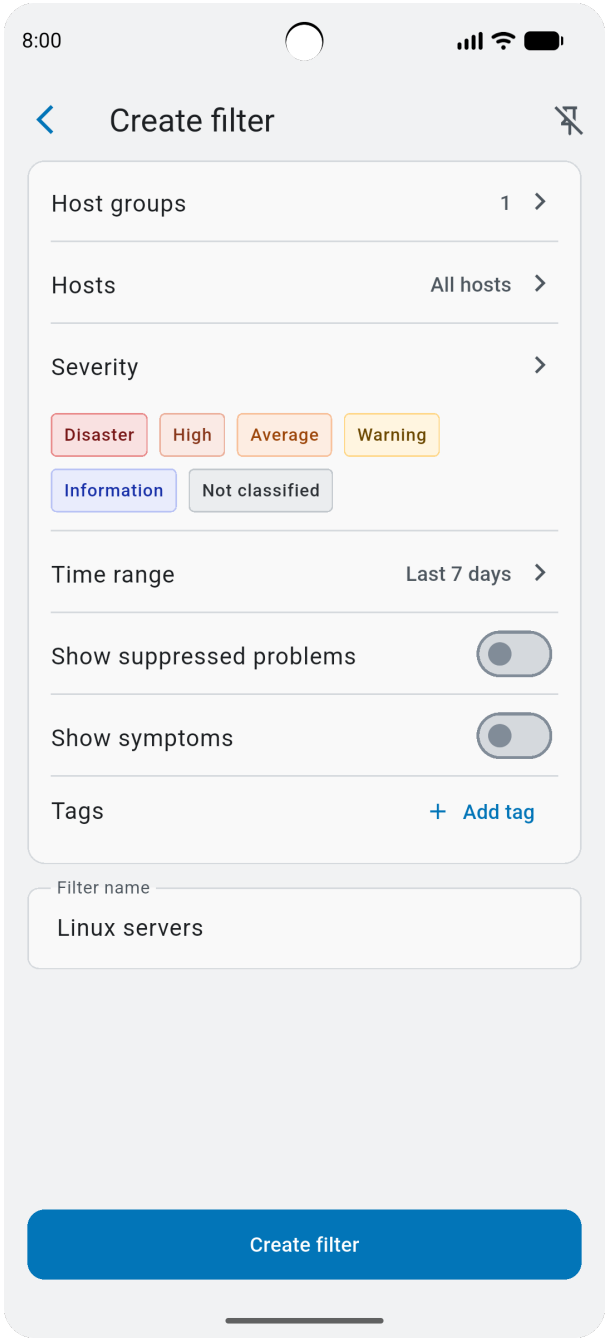
Todos los filtros La pantalla *Todos los filtros* permite gestionar los filtros. Contiene:

- Nombres de filtros: toque para editar el filtro.
- Iconos de chincheta: toque para fijar o desfijar el filtro en la pantalla *Problemas*.
- *Crear filtro*: toque para crear un filtro nuevo.



Crear filtro La pantalla *Crear filtro* permite crear un filtro con opciones similares a las de *Monitorización* > *Problemas* (consulte los detalles). También puede establecer un *Intervalo de tiempo* para determinar durante cuánto tiempo se mostrarán los problemas sin resolver.

Pulse *Crear filtro* para guardar el filtro. Los filtros nuevos se fijan en la pantalla *Problemas*; pulse el icono de fijación para cambiar esta configuración.



Lista de problemas La pantalla *Lista de problemas* muestra los problemas que coinciden con el filtro (por ejemplo, *Problemas recientes*).

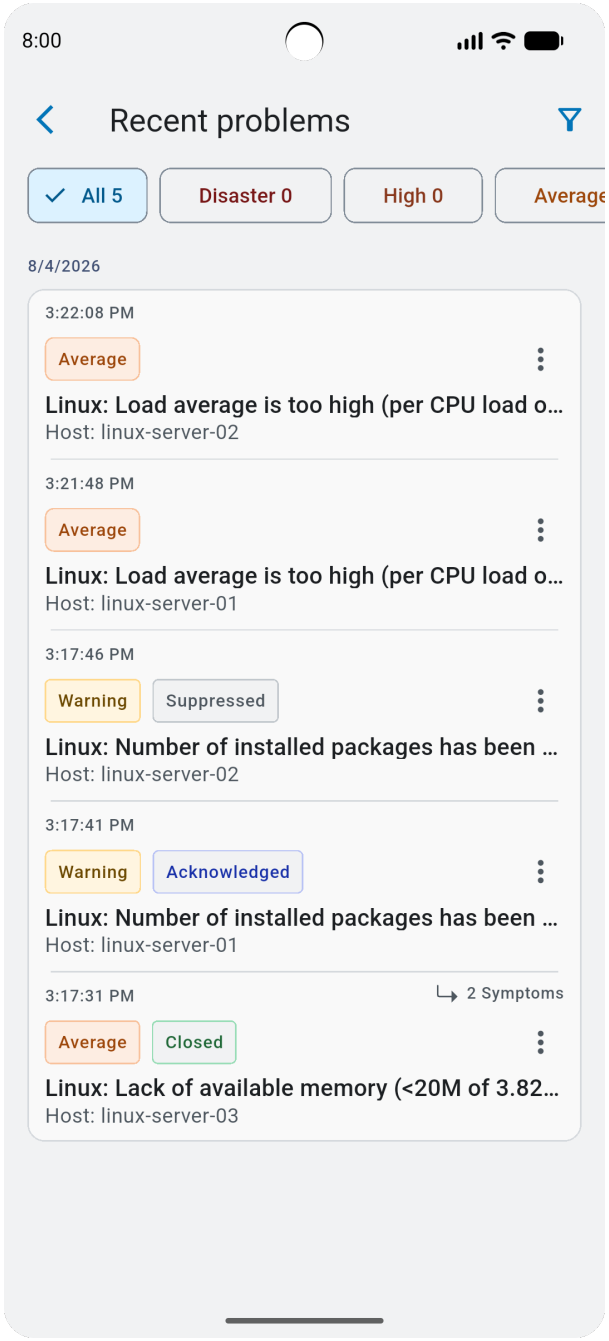
Toque el icono de filtro o una gravedad de problema para restringir aún más la lista.

Para cada problema, puede ver:

- La hora y la fecha de inicio.
- La gravedad, el nombre y el host.
- Si se suprimió (se muestra si *Mostrar problemas suprimidos* está habilitado).
- Si se confirmó o se cerró recientemente (se resolvió automáticamente o se cerró manualmente).
- Si tiene síntomas; toque para ver la lista de problemas de síntomas.

Toque un problema para abrir la pantalla *Detalles del problema*.

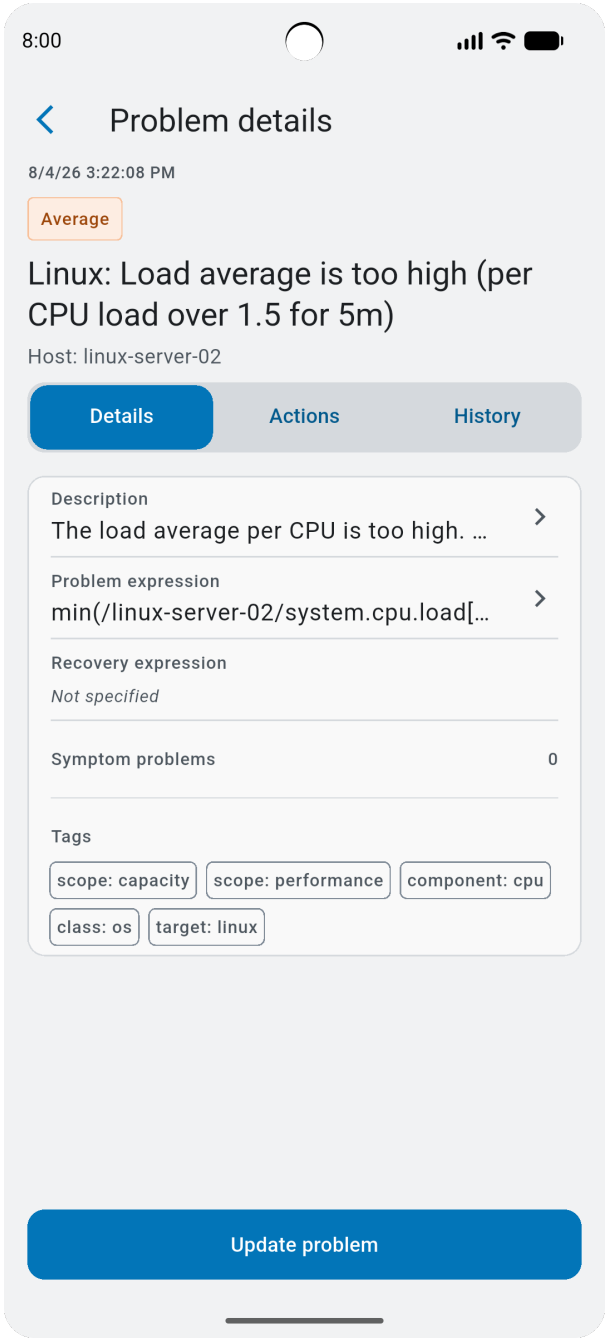
Toque el menú de tres puntos para confirmar rápidamente el problema o suprimirlo durante el tiempo que elija.



Detalles del problema La pantalla *Detalles del problema* muestra los detalles, las acciones y el historial del problema.

- Pestaña *Detalles*: muestra la descripción del problema, las expresiones, los problemas sintomáticos (toque para ver la lista de problemas) y las etiquetas.
- Pestaña *Acciones*: muestra la cronología del problema: cuándo comenzó o se reconoció, si se envió una notificación y otras acciones.
- Pestaña *Historial*: muestra la antigüedad, la duración y el tiempo de recuperación del problema (o *No disponible* si el problema continúa activo).

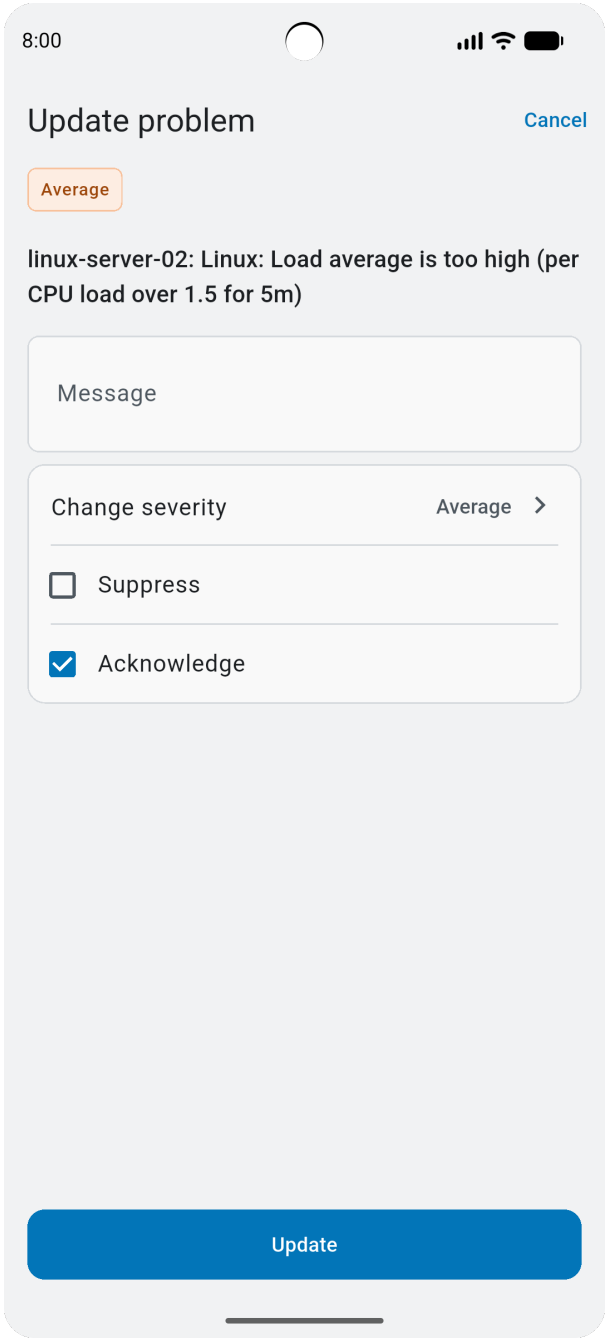
Toque *Actualizar problema* para actualizar el problema.



Actualizar problema La pantalla *Actualizar problema* permite actualizar un problema:

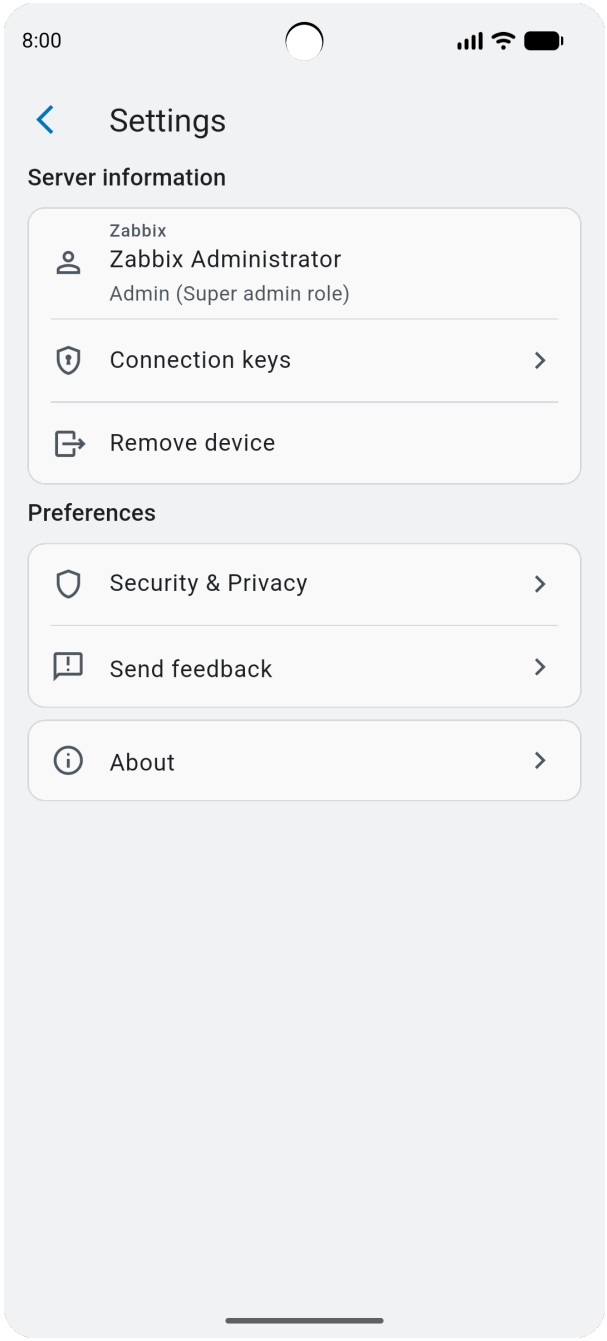
- Añadir un mensaje.
- Cambiar la gravedad del problema.
- Suprimir el problema durante el tiempo que elija.
- Confirmar o cancelar la confirmación del problema.
- Cerrar el problema.

Pulse *Actualizar* para guardar los cambios.



Configuración La pantalla *Configuración* permite acceder a la configuración de la aplicación. Contiene:

- Nombre del server de Zabbix (o *Zabbix*, si no se configuró durante la instalación de la interfaz web de Zabbix).
- Nombre, tipo y rol del usuario al que se agregó el dispositivo móvil.
- *Claves de conexión*: abrir para ver y copiar el ID del server, el ID del dispositivo y otros detalles.
- *Eliminar dispositivo*: tocar para eliminar el dispositivo móvil de Zabbix.
- *Seguridad y privacidad*: abrir para activar el bloqueo de la aplicación y administrar el uso compartido de datos de diagnóstico.
- *Enviar comentarios*: abrir para enviar comentarios a Zabbix.
- *Acerca de*: abrir para ver la versión de la aplicación y otros detalles sobre ella.



Acceso móvil seguro

Esta página contiene las prácticas recomendadas para mantener seguras la aplicación móvil de Zabbix y el adaptador Zabbix bridge.

Protege los dispositivos móviles Para evitar que otras personas abran la aplicación en tu dispositivo móvil, activa el bloqueo de la aplicación en *Configuración*.

Si un dispositivo se ha perdido o ha sido comprometido, elimínalo en la interfaz web de Zabbix. Esto revoca y elimina sus credenciales de acceso, por lo que el dispositivo ya no puede acceder a Zabbix. Según los permisos de tu rol de usuario, puedes eliminar dispositivos en las siguientes secciones:

- *Configuración del usuario > Dispositivos*: cualquier usuario puede eliminar sus propios dispositivos (requiere el permiso *Administrar dispositivos propios*).
- *Usuarios > Dispositivos*: *Super admin* puede eliminar los dispositivos de cualquier usuario (requiere el permiso *Administrar dispositivos de usuarios*).

También puedes limitar qué partes de Zabbix pueden utilizarse desde un dispositivo móvil. Para ello, asigna a los usuarios un rol de usuario que otorgue únicamente los permisos que necesita la aplicación. Los permisos mínimos son los siguientes:

Sección	Configuración
Acceso a elementos de la interfaz de usuario	Desmarca todas las casillas.
Acceso a servicios	Establece *Acceso de lectura y escritura a los servicios* y *Acceso de solo lectura a los servicios* en `None`.
Acceso a módulos	Desmarca todas las casillas.
Acceso a la API	Desmarca la casilla *Habilitado*.
Acceso a acciones	Desmarca todas las casillas. Para permitir que los usuarios actualicen problemas en la aplicación, marca también las acciones que necesiten: - *Añadir comentarios a los problemas* - *Cambiar la gravedad* - *Confirmar problemas* - *Suprimir problemas* - *Cerrar problemas* - *Crear y editar medios propios* Ten en cuenta que los problemas que los usuarios pueden ver en la aplicación también dependen de su acceso a los hosts.
Dispositivos	Marca la casilla *Habilitado*. Para permitir que los usuarios añadan y eliminen sus propios dispositivos móviles, marca también la casilla *Administrar dispositivos propios*; de lo contrario, un *Super admin* debe administrar sus dispositivos en *Usuarios > Dispositivos*.

Proteger el adaptador de puente de Zabbix El adaptador de puente de Zabbix almacena sus datos (el archivo de estado y las claves de cifrado) en el directorio especificado en el parámetro `DataDir` del archivo de configuración del adaptador de puente de Zabbix. Asegúrese de que el directorio tenga permisos 0700 y que todos los archivos almacenados en él tengan permisos 0600.

El adaptador de puente de Zabbix transfiere las solicitudes de los dispositivos móviles a la API de Zabbix. Para restringir los métodos de la API que se pueden llamar a través del adaptador de puente de Zabbix, utilice su parámetro `AllowedAPIMethods`. Por ejemplo, permita únicamente el conjunto mínimo de métodos que requiere la aplicación móvil, o incluso menos si no desea que se utilicen algunas funciones de la aplicación móvil (por ejemplo, no permita `event.acknowledge` si los usuarios de la aplicación móvil no deben actualizar los problemas).

Centro de desarrolladores

Centro para desarrolladores

Comience a desarrollar extensiones personalizadas para Zabbix con orientación sobre plugins, módulos y widgets.

Construir sobre Zabbix

Plugins Una visión general sobre cómo desarrollar y gestionar plugins de Zabbix para ampliar la funcionalidad o integrarse con sistemas externos.

Biblioteca de Python para Zabbix Recursos para usar la biblioteca oficial `zabbix_utils` para interactuar con la API de Zabbix y crear integraciones.

Módulos del frontend Guías y referencias para crear módulos personalizados que amplían o modifican el frontend de Zabbix para adaptarlo a casos de uso específicos.

Widgets Un desglose de la estructura y la lógica de los widgets, con instrucciones para crear elementos de dashboard personalizados adaptados a sus necesidades.

Creación de un plugin

Esta página está destinada a quienes estén interesados en desarrollar su propio plugin (recopilador de datos independiente) para Zabbix agent 2.

Un plugin personalizado amplía la funcionalidad de Zabbix agent 2 más allá de los plugins integrados y los plugins cargables proporcionados.

Cada plugin es un paquete de *Go* que define la estructura e implementa una o varias interfaces (*Exporter*, *Configurator*, *Runner*). Para obtener más información, consulte Interfaces de plugins y Diagrama de conexión.

Esta guía le ayudará a crear un plugin cargable personalizado.

Para obtener orientación adicional, consulte también el repositorio de:

- Plugin de ejemplo
- Plugins cargables existentes

Lo que crearás Este es un tutorial paso a paso para crear un plugin cargable simple llamado **MyIP**. Este plugin implementará una única clave de item (`myip`) que devuelve la dirección IP externa del host del agent de Zabbix.

Paso 1: Configuración

1. Un plugin es un módulo Go estándar. Comience inicializando el archivo `go.mod` en el directorio del plugin para realizar el seguimiento de las dependencias del plugin:

```
cd path/to/plugins/myip # Cambie a su directorio del plugin
go mod init myip
```

2. Instale la dependencia obligatoria Zabbix Go SDK (`golang.zabbix.com/sdk`):

```
go get golang.zabbix.com/sdk@$LATEST_COMMIT_HASH
```

Sustituya `$LATEST_COMMIT_HASH` por el hash del commit `HEAD` más reciente del repositorio `golang.zabbix.com/sdk` en la rama de versión correspondiente. Por ejemplo:

```
go get golang.zabbix.com/sdk@af85407
```

Tenga en cuenta que actualmente no se admite el control de versiones de `golang.zabbix.com/sdk`, pero esto puede cambiar en el futuro.

Las dependencias adicionales pueden instalarse según sea necesario mediante `go get`.

3. Cree un archivo `main.go` vacío para el código fuente del plugin:

```
touch main.go
```

Ahora la configuración inicial está completa y el plugin está listo para el desarrollo.

Paso 2: Estructura del plugin El módulo `golang.zabbix.com/sdk`, instalado en el paso anterior, proporciona el framework necesario para desarrollar plugins y garantiza que todos los plugins tengan una estructura coherente.

1. Configurar el flujo de ejecución básico.

Comience definiendo el flujo de ejecución principal del plugin. Añada el siguiente código a `main.go`:

```
package main

func main() {
    err := run()
    if err != nil {
        panic(err)
    }
}

func run() error {
    return nil
}
```

Esto establece el flujo de ejecución básico del plugin. La función run contendrá posteriormente la lógica principal del plugin.

2. Explorar las interfaces del plugin.

Un plugin de Zabbix agent 2 debe estar representado por una estructura que implemente interfaces del paquete `golang.zabbix.com/sdk/p`

- *Accessor*: define los métodos esenciales que todos los plugins deben implementar, como establecer el nombre del plugin y gestionar los tiempos de espera de las claves de los items.
- Una o más de las siguientes interfaces funcionales del plugin:
  - *Exporter*: realiza un sondeo y devuelve un valor (o varios valores), nada o un error; suele utilizarse junto con la interfaz *Collector*.
  - *Collector*: gestiona la recopilación periódica de datos.
  - *Runner*: define los procedimientos de inicio y apagado del plugin.
  - *Watcher*: permite realizar sondeos independientes de métricas, omitiendo el planificador interno del agent; resulta útil para la monitorización basada en traps o en eventos.
  - *Configurator*: define cómo el plugin lee y aplica sus opciones de configuración.

Puede implementar estas interfaces usted mismo o utilizar las implementaciones predeterminadas proporcionadas por Zabbix Go SDK y modificarlas según sea necesario. En este tutorial se utilizan las implementaciones predeterminadas.

3. Crear la estructura del plugin.

Ahora, importe el paquete *plugin* y cree una estructura `myIP` que incluya la estructura `plugin.Base`:

```
import "golang.zabbix.com/sdk/plugin"

type myIP struct {
    plugin.Base
}
```

Actualmente, la estructura myIP satisface la interfaz Accessor. Más adelante en el tutorial se añadirá un método para implementar una de las interfaces funcionales del plugin, *Exporter*.

Paso 3: Definir claves de item Su plugin necesita las claves de item para recopilar datos y proporcionarlos al server o proxy de Zabbix.

1. Importe el paquete *errs* para el manejo de errores:

```
import "golang.zabbix.com/sdk/errs"
```

2. Registre las claves de item mediante la función `plugin.RegisterMetrics()` dentro de la función `run()`:

```
func run() error {
    p := &myIP{}

    // Register the `myip` item key.
    err := plugin.RegisterMetrics(
        p,
        "MyIP",                             // Plugin name
        "myip",                             // Item key name
        "Returns the host's IP address.", // Item key description
    )
    if err != nil {
        return errs.Wrap(err, "failed to register metrics")
    }

    return nil
}
```

Para registrar varias claves de item, repita los parámetros *nombre de métrica* y *descripción* para cada métrica. Por ejemplo:

```
plugin.RegisterMetrics(&impl, "Myip", "metric.one", "Metric one description.", "metric.two", "Metric two d
```

Paso 4: Configurar el handler El handler facilita la comunicación entre el agent y el plugin.

1. Importe el paquete *container*:

```
import "golang.zabbix.com/sdk/plugin/container"
```

2. Dentro de la función `run()`, añada código para crear y configurar un handler:

```
func run() error {
    p := &myIP{}

    // Register the `myip` item key.
```

```
    err := plugin.RegisterMetrics(
        p,
        "MyIP",                           // Plugin name
        "myip",                           // Item key name
        "Returns the host's IP address.", // Item key description
    )
    if err != nil {
        return errs.Wrap(err, "failed to register metrics")
    }

    // Create a new handler.
    h, err := container.NewHandler("MyIP") // Plugin name
    if err != nil {
        return errs.Wrap(err, "failed to create new handler")
    }

    // Setup logging to forward logs from the plugin to the agent.
    // Available via p.Logger.Infof, p.Logger.Debugf, etc.
    p.Logger = h

    // Start plugin execution.
    // Blocks until a termination request is received from the agent.
    err = h.Execute()
    if err != nil {
        return errs.Wrap(err, "failed to execute plugin handler")
    }

    return nil
}
```

Paso 5: Implementar la recopilación de datos La recopilación de datos se realiza mediante la interfaz Exporter, que describe el método Export:

```
func Export(
  key string,             // La clave del item que se va a recopilar.
  params []string,        // Argumentos pasados a la clave del item (`myip[arg1, arg2]`).
  context ContextProvider // Metadatos sobre la recopilación de datos de la clave del item.
) (any, error)
```

1. Importe los paquetes necesarios para solicitudes HTTP y lectura de respuestas:

```
import (
    "io"
    "net/http"
)
```

2. Implemente el método Export para la estructura myIP:

```
func (p *myIP) Export(
    key string, params []string, context plugin.ContextProvider,
) (any, error) {
    // El plugin puede usar una lógica de recopilación de datos diferente según el parámetro `key`.
    // Esta implementación solo verifica que la `key` proporcionada sea compatible.
    if key != "myip" {
        return nil, errs.Errorf("unknown item key %q", key)
    }

    // El registro se reenviará al registro del agent 2.
    p.Infof(
        "received request to handle %q key with %d parameters",
        key,
        len(params),
    )
```

```
    // Recopile los datos y devuélvalos.

    resp, err := http.Get("https://api.ipify.org")
    if err != nil {
        return nil, errs.Wrap(err, "failed to get IP address")
    }

    defer resp.Body.Close()

    body, err := io.ReadAll(resp.Body)
    if err != nil {
        return nil, errs.Wrap(err, "failed to read response body")
    }

    return string(body), nil
}
```

Paso 6: Compilar y configurar el plugin

1. Para compilar el plugin, ejecute:

```
go mod tidy
go build
```

Esto debería crear un ejecutable `myip` en el directorio actual.

2. Configure Zabbix agent 2 para usar el plugin:

```
echo "Plugins.MyIP.System.Path=$PATH_TO_THE_MYIP_PLUGIN_EXECUTABLE" > /etc/zabbix/zabbix_agent2.d/plugins.
```

Reemplace `$PATH_TO_THE_MYIP_PLUGIN_EXECUTABLE` por la ruta al `myip` creado en el paso 5.

El nombre del plugin en el nombre del parámetro de configuración (*MyIP* en este tutorial) debe coincidir con el nombre del plugin definido en la función *plugin.RegisterMetrics()*.

3. Para probar el plugin y su item `myip`, ejecute:

```
zabbix_agent2 -c /etc/zabbix/zabbix_agent2.conf -t myip
```

La salida debería contener una dirección IP externa de su host y verse similar a esto:

```
myip                                            [s|192.0.2.1]
```

Con esto, ha creado un plugin cargable sencillo para Zabbix agent 2. ¡Felicidades!

```
package main

import (
    "io"
    "net/http"

    "golang.zabbix.com/sdk/errs"
    "golang.zabbix.com/sdk/plugin"
    "golang.zabbix.com/sdk/plugin/container"
)

var _ plugin.Exporter = (*myIP)(nil)

type myIP struct {
    plugin.Base
}

func main() {
    err := run()
    if err != nil {
        panic(err)
```

```
    }
}

func run() error {
    p := &myIP{}

    // Registra la clave de item `myip`.
    err := plugin.RegisterMetrics(
        p,
        "MyIP",                          // Nombre del plugin
        "myip",                          // Nombre de la clave de item
        "Devuelve la dirección IP del host.", // Descripción de la clave de item
    )
    if err != nil {
        return errs.Wrap(err, "failed to register metrics")
    }

    // Crea un nuevo controlador.
    h, err := container.NewHandler("MyIP") // Nombre del plugin
    if err != nil {
        return errs.Wrap(err, "failed to create new handler")
    }

    // Configura el registro para reenviar los logs del plugin al agent.
    // Disponible mediante p.Logger.Infof, p.Logger.Debugf, etc.
    p.Logger = h

    // Inicia la ejecución del plugin.
    // Se bloquea hasta que se recibe una solicitud de finalización del agent.
    err = h.Execute()
    if err != nil {
        return errs.Wrap(err, "failed to execute plugin handler")
    }

    return nil
}

func (p *myIP) Export(
    key string, params []string, context plugin.ContextProvider,
) (any, error) {
    // El plugin puede usar una lógica de recopilación de datos diferente según el parámetro `key`.
    // Esta implementación solo verifica que la `key` proporcionada sea compatible.
    if key != "myip" {
        return nil, errs.Errorf("unknown item key %q", key)
    }

    // El log se reenviará al log 2 del agent.
    p.Infof(
        "received request to handle %q key with %d parameters",
        key,
        len(params),
    )

    // Recopila los datos y los devuelve.

    resp, err := http.Get("https://api.ipify.org")
    if err != nil {
        return nil, errs.Wrap(err, "failed to get IP address")
    }

    defer resp.Body.Close()
```

```
    body, err := io.ReadAll(resp.Body)
    if err != nil {
        return nil, errs.Wrap(err, "failed to read response body")
    }

    return string(body), nil
}
```

Código fuente completo

Interfaces de plugin

Esta sección describe las interfaces de plugin disponibles.

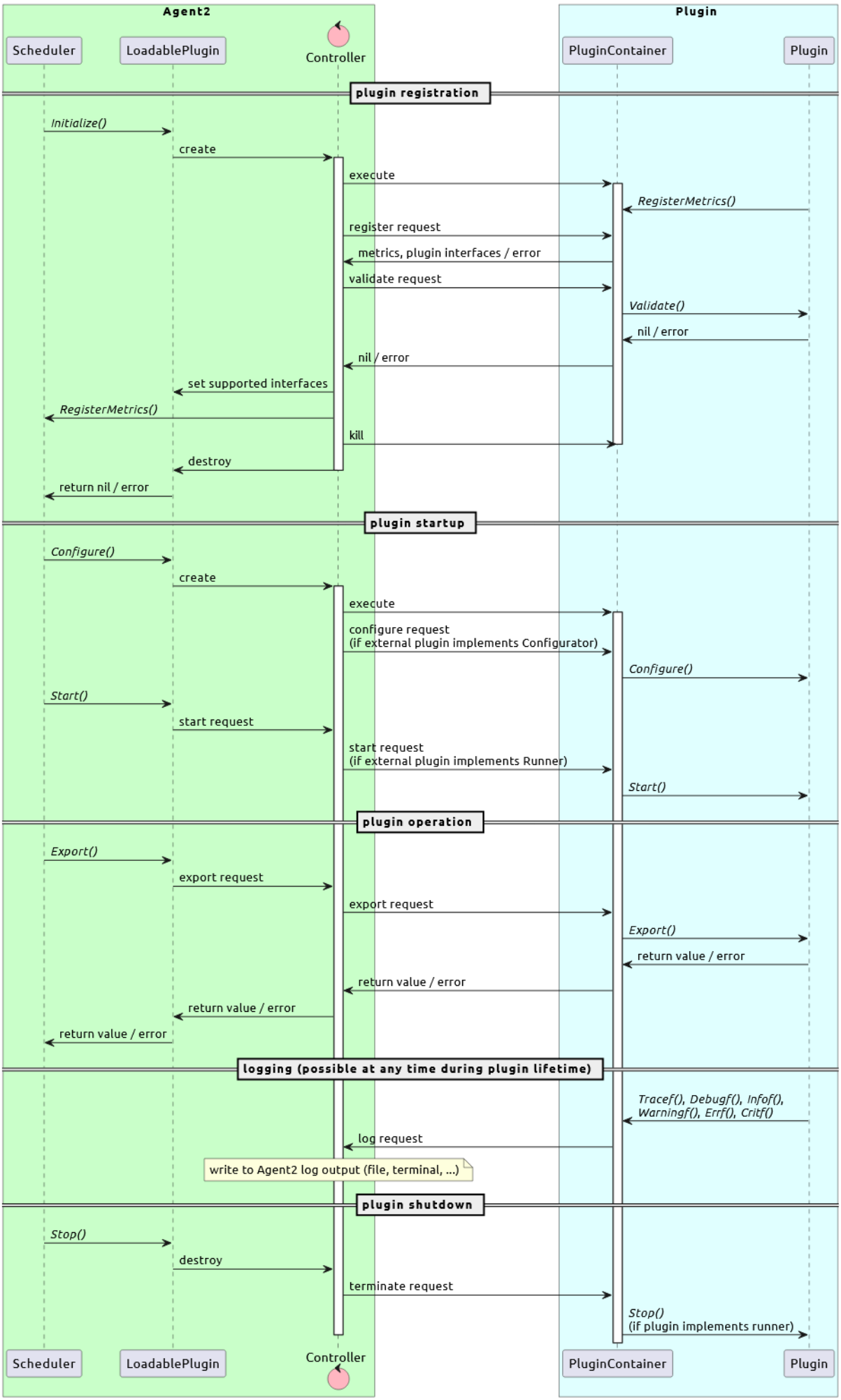
plugin.Exporter *Exporter* es la interfaz más simple que realiza una consulta y devuelve un valor (valores), nada o un error. Acepta una clave de métrica preanalizada, parámetros y contexto. El acceso a todas las demás interfaces de plugin es exclusivo y no se puede llamar a ningún método si un plugin ya está realizando una tarea. Además, existe un límite de 100 llamadas *Export()* concurrentes como máximo por plugin, que puede reducirse según los requisitos de cada plugin.

plugin.Configurator La interfaz *Configurator* proporciona parámetros de configuración del plugin a partir de los archivos de configuración del agente Zabbix 2.

plugin.Runner La interfaz *Runner* proporciona los medios para realizar la inicialización cuando un plugin se inicia (activa) y la desinicialización cuando un plugin se detiene (desactiva). Por ejemplo, un plugin puede iniciar/detener alguna *goroutine* en segundo plano implementando la interfaz Runner.

Diagrama de conexión Zabbix agent 2 se conecta bidireccionalmente a los plugins mediante sockets UNIX en Linux y Named Pipes en Windows.

El diagrama de conexión siguiente ilustra el proceso de comunicación entre Zabbix agent 2 y un plugin cargable, así como el proceso de recopilación de métricas.


Agent2
Plugin
Scheduler
LoadablePlugin
Controller
PluginContainer
Plugin
plugin registration
Initialize()
create
execute
RegisterMetrics()
register request
metrics, plugin interfaces / error
validate request
Validate()
nil / error
nil / error
set supported interfaces
RegisterMetrics()
kill
destroy
return nil / error
plugin startup
Configure()
create
execute
configure request
(if external plugin implements Configurator)
Configure()
Start()
start request
start request
(if external plugin implements Runner)
Start()
plugin operation
Export()
export request
export request
Export()
return value / error
return value / error
return value / error
return value / error
logging (possible at any time during plugin lifetime)
Tracef(), Debugf(), Infof(),
Warningf(), Errf(), Critf()
log request
write to Agent2 log output (file, terminal, ...)
plugin shutdown
Stop()
destroy
terminate request
Stop()
(if plugin implements runner)
Scheduler
LoadablePlugin
Controller
PluginContainer
Plugin

Biblioteca Python para Zabbix

zabbix_utils es la biblioteca oficial de Python para Zabbix.

Con zabbix_utils puedes:

- Usar la API de Zabbix para gestionar objetos de Zabbix (crear hosts, actualizar items, obtener eventos, etc.).
- Recopilar datos del agent de Zabbix (de forma similar a Zabbix get).
- Enviar datos al server o proxy de Zabbix (de forma similar a Zabbix sender).

La biblioteca admite la ejecución de scripts tanto síncronos como asíncronos, lo que le permite manejar flujos de trabajo simples y operaciones paralelas a gran escala, como la recopilación masiva de datos, exportaciones masivas, procesamiento de alertas y escalado eficiente del sistema.

La biblioteca es compatible con las versiones soportadas del server de Zabbix y las versiones modernas de Python.

Para comenzar, instala la biblioteca y luego sigue la guía de inicio rápido.

Instalación

zabbix_utils puede instalarse utilizando cualquiera de los métodos descritos a continuación.

Requisitos:

- Zabbix 6.0 o posterior (probado en 6.0, 7.0, 7.2, 7.4)
- Python 3.8 o posterior (probado en 3.8–3.13)
- Framework HTTP asíncrono aiohttp (requerido solo para el modo asíncrono)

Python Package Index (PyPI)

Este es el método más común para la mayoría de los entornos:

```
pip install zabbix_utils
pip install zabbix_utils[async] # Sólo para soporte asíncrono
```

Repositorio de Zabbix

Utilice este método si prefiere gestionar las dependencias a través del gestor de paquetes de su sistema.

Descargue los paquetes de Zabbix para su distribución de Linux y luego ejecute los siguientes comandos.

En RHEL y derivados:

```
dnf install python3-zabbix-utils
dnf install epel-release # Solo para soporte async
dnf install python3-aiohttp # Solo para soporte async
```

En Debian/Ubuntu y derivados:

```
apt install python3-zabbix-utils
apt install python3-aiohttp # Solo para soporte async
```

Fuente (GitHub)

Utilice este método si prefiere la última versión de desarrollo:

```
git clone https://github.com/zabbix/python-zabbix-utils
cd python-zabbix-utils/
pip install -r requirements.txt # Solo para soporte asíncrono
python3 setup.py install
```

Para instalar las dependencias necesarias para el modo asíncrono, también puede utilizar uno de los métodos descritos anteriormente.

Guía de inicio rápido

Después de haber instalado zabbix_utils, puedes usarlo en tu script.

Esta guía de inicio rápido te muestra cómo:

- Recuperar el nombre de host desde el agent de Zabbix.
- Crear un host y un item trapper usando la API de Zabbix.
- Enviar un valor al item.

La guía presenta el script paso a paso, explicando cada parte a medida que se introduce. El script completo se proporciona al final de la página.

La guía también asume que tu server, agent y API de Zabbix se están ejecutando localmente.

Recuperar el nombre del host desde el agent de Zabbix

Comience recuperando el nombre del host del sistema donde se está ejecutando el agent de Zabbix. Necesitará este nombre de host para crear un host usando la API de Zabbix.

1. Importe la clase `Getter` de zabbix_utils. Esta clase funciona como Zabbix get y le permite solicitar datos al agent de Zabbix.

2. Cree una instancia de `Getter` que se conecte al agent de Zabbix local en `127.0.0.1` en el puerto 10050.

3. Llame al método `get()` en la instancia de `Getter` para solicitar el nombre del host al agent de Zabbix.

- El método `get()` toma una clave de item como parámetro y envía una solicitud al agent de Zabbix para ese item.
- El método `get()` devuelve un objeto, y el nombre del host se almacena dentro del atributo `value` de ese objeto.
- `system.hostname` es una clave incorporada del agent de Zabbix que devuelve el nombre del host.

```
from zabbix_utils import Getter

agent = Getter(host='127.0.0.1', port=10050)
hostname = agent.get('system.hostname').value
```

Conectarse e iniciar sesión en la API de Zabbix

A continuación, conéctese a la API de Zabbix. Esto permite que su script gestione objetos de Zabbix como hosts e items.

1. Importe la clase `ZabbixAPI` desde zabbix_utils.

2. Cree una instancia de `ZabbixAPI` y proporcione la URL de su interfaz web de Zabbix.

3. Llame al método `login()` en la instancia de `ZabbixAPI` y proporcione su nombre de usuario y contraseña. La cuenta debe tener permiso para acceder a la API de Zabbix.

4. Llame al método `logout()` para cerrar la sesión después de las operaciones de la API.

```
from zabbix_utils import ZabbixAPI

api = ZabbixAPI(url='127.0.0.1/zabbix')
api.login(user='Admin', password='zabbix')

### Las operaciones de la API van aquí

api.logout()
```

Crear un host en Zabbix

Ahora que tienes el nombre de host del agent de Zabbix y estás conectado a la API de Zabbix, puedes crear un nuevo host en Zabbix.

1. Llama al método de la API `host.create()` en la instancia de `ZabbixAPI` y proporciona los detalles del host:

- `host` - establece la variable `hostname` que contiene el nombre de host que recuperaste del agent de Zabbix.
- `interfaces` - contiene los detalles de conexión para el agent de Zabbix que se está ejecutando en el host.
- `groups` - contiene al menos un grupo de hosts al que debe pertenecer el host.

2. Dado que la API de Zabbix devuelve el ID del host recién creado, guarda este ID en la variable `host_id` para su uso posterior.

3. Imprime un mensaje para confirmar que el host fue creado.

```
api.host.create(
    host=hostname,
    interfaces=[{
        'type': 1,
        'main': 1,
```

```
        'useip': 1,
        'ip': '127.0.0.1',
        'dns': '',
        'port': '10050',
    }],
    groups=[{'groupid': '2'}]
)

host_id = host['hostids'][0]

print(f"Host '{hostname}' creado con ID {host_id}")
```

Crear un nuevo item en Zabbix

Después de crear un host, puede añadirle un item.

1. Defina una clave única para el item que se va a crear.

2. Llame al método `item.create()` en la instancia de `ZabbixAPI` y proporcione los detalles del item:

   - `hostid` - establezca el valor de la variable `host_id` que contiene el ID del host que acaba de crear.
   - `name` - un nombre para el item.
   - `key_` - establezca el valor de la variable `item_key` que acaba de definir.
   - `type` - establezca el valor 2 (item trapper), necesario para recibir los valores enviados desde su script en los siguientes pasos.
   - `value_type` - establezca el valor 3 (numérico sin signo), el tipo de datos que almacena este item.

3. Dado que la API de Zabbix devuelve el ID del item recién creado, guarde este ID en la variable `item_id` para su uso posterior.

4. Imprima un mensaje para confirmar que el item se ha creado.

```
item_key = 'app.myservice.heartbeat'

item = api.item.create(
    hostid=host_id,
    name='App heartbeat',
    key_=item_key,
    type=2,
    value_type=3,
)

item_id = item['itemids'][0]

print(f"Item '{item_key}' creado con ID {item_id}")
```

Enviar valor al host

Ahora que ha creado su host y su item, puede enviarle datos.

1. Importe la clase `time` y espere unos segundos antes de enviar los datos. Esto asegura que Zabbix haya procesado completamente su nuevo host y su item.

2. Importe la clase `Sender` de zabbix_utils. Esta clase funciona como Zabbix sender y permite que su script envíe datos a Zabbix.

3. Cree una instancia de `Sender` que se conecte al servidor Zabbix local en `127.0.0.1` en el puerto `10051`.

4. Llame al método `send_value()` en la instancia de `Sender` para enviar un valor al item de su host y proporcione estos detalles:

   - `hostname` - establezca la variable `hostname` que contiene el nombre del host que recuperó del agent de Zabbix.
   - `item_key` - establezca la variable `item_key` que acaba de definir anteriormente en su script.
   - `1` - el valor a enviar.

4. Imprima un mensaje para confirmar que se envió el valor.

```
import time
time.sleep(10)

from zabbix_utils import Sender

sender = Sender(server='127.0.0.1', port=10051)
```

```
response = sender.send_value(hostname, item_key, 1)

print(f"Sender response: {response}")
```

Después de que este código se ejecute correctamente, debería ver un mensaje de éxito que indica que Zabbix recibió su valor:

```
Sender response: {"processed": 1, "failed": 0, "total": 1, "time": "0.000151", "chunk": 1}
```

Ahora también puede comprobar la interfaz web de Zabbix (*Monitorización* > *Últimos datos*) para ver el valor.

Script completo

A continuación se muestra el script completo que combina todos los pasos: recuperar el nombre del host, crear el host y el item usando la API de Zabbix y enviar datos a Zabbix.

```
import time
from zabbix_utils import Getter, ZabbixAPI, Sender

### Recuperar el nombre del host desde el agent de Zabbix
agent = Getter(host='127.0.0.1', port=10050)
hostname = agent.get('system.hostname').value

### Conectar e iniciar sesión en la API de Zabbix
api = ZabbixAPI(url='127.0.0.1/zabbix')
api.login(user='Admin', password='zabbix')

### Crear el host en Zabbix
host = api.host.create(
    host=hostname,
    interfaces=[{
        'type': 1,
        'main': 1,
        'useip': 1,
        'ip': '127.0.0.1',
        'dns': '',
        'port': '10050',
    }],
    groups=[{'groupid': '2'}]
)

host_id = host['hostids'][0]

print(f"Host '{hostname}' creado con ID {host_id}")

### Crear el item en Zabbix
item_key = 'app.myservice.heartbeat'

item = api.item.create(
    hostid=host_id,
    name='App heartbeat',
    key_=item_key,
    type=2,
    value_type=3
)

item_id = item['itemids'][0]

print(f"Item '{item_key}' creado con ID {item_id}")

### Cerrar sesión en la API
api.logout()

### Esperar a que Zabbix procese el nuevo host
time.sleep(10)
```

```
### Enviar valor al host
sender = Sender(server='127.0.0.1', port=10051)
response = sender.send_value(hostname, item_key, 1)

print(f"Respuesta del sender: {response}")
```

Usar la API de Zabbix

zabbix_utils le permite utilizar la API de Zabbix para gestionar objetos de Zabbix, incluyendo la creación de hosts, actualización de items, recuperación de eventos y más.

Las solicitudes a la API pueden realizarse en modo síncrono o asíncrono:

- En modo síncrono, su script de Python envía una solicitud y espera una respuesta antes de continuar, lo que es adecuado para operaciones simples, secuenciales y predecibles.
- En modo asíncrono, el script envía solicitudes sin esperar cada respuesta, permitiendo que otras operaciones se realicen en paralelo; esto es más eficiente para solicitudes lentas o grandes lotes de datos.

Los ejemplos de esta página se centran en el modo síncrono, aunque el modo asíncrono sigue patrones similares. Hay ejemplos adicionales disponibles en el repositorio de GitHub zabbix_utils.

Importar

Para usar zabbix_utils para trabajar con la API de Zabbix, importe la clase `ZabbixAPI` en su script:

```
from zabbix_utils import ZabbixAPI
```

> **Note:**
> Si ya utiliza una librería Python de la comunidad, normalmente puede reemplazar esa importación por `ZabbixAPI` de zabbix_utils.

Iniciar sesión

Antes de realizar solicitudes a la API, debe crear una instancia de `ZabbixAPI` e iniciar sesión en la API de Zabbix.

Elija el método que mejor se adapte a la forma en que gestiona las credenciales. Puede utilizar su nombre de usuario y contraseña o tokens de API.

Durante la inicialización

Al crear una instancia de `ZabbixAPI`, puede proporcionar la URL de la interfaz web de Zabbix y sus credenciales de una sola vez:

```
### Usuario y contraseña:
api = ZabbixAPI(url="127.0.0.1/zabbix", user="Admin", password="zabbix")

### Token de API:
api = ZabbixAPI(url="127.0.0.1/zabbix", token="your_api_token")
```

Si prefiere agrupar los parámetros de conexión, puede pasarlos como un diccionario de Python y desempaquetarlos usando la sintaxis de argumentos de palabras clave de Python:

```
ZABBIX_AUTH = {
    "url": "127.0.0.1/zabbix",
    "user": "Admin",
    "password": "zabbix"
}

api = ZabbixAPI(**ZABBIX_AUTH)
```

Usando `login()`

Primero puede crear una instancia de `ZabbixAPI` solo con la URL de la interfaz web de Zabbix y luego usar el método `login()` con sus credenciales:

```
### Nombre de usuario y contraseña:
api = ZabbixAPI(url="127.0.0.1/zabbix")
api.login(user="Admin", password="zabbix")
```

```
### Token de API:
api = ZabbixAPI(url="127.0.0.1/zabbix")
api.login(token="your_api_token")
```

Uso de variables de entorno

Primero puede almacenar sus credenciales como variables de entorno, por ejemplo, usando una terminal:

```
### Nombre de usuario y contraseña:
export ZABBIX_USER="Admin"
export ZABBIX_PASSWORD="zabbix"

### Token:
export ZABBIX_TOKEN="your_api_token"
```

Luego, si inicia su script desde la misma sesión de terminal, la instancia de `ZabbixAPI` en el script solo requiere la URL de la interfaz web de Zabbix:

```
api = ZabbixAPI(url="127.0.0.1/zabbix")
```

También puede almacenar la URL en una variable de entorno:

```
### Nombre de usuario y contraseña:
export ZABBIX_USER="Admin"
export ZABBIX_PASSWORD="zabbix"
export ZABBIX_URL="https://127.0.0.1/zabbix"

### Token:
export ZABBIX_TOKEN="your_api_token"
export ZABBIX_URL="https://127.0.0.1/zabbix"
```

Cuando la URL y las credenciales están configuradas como variables de entorno, la instancia de `ZabbixAPI` en el script no requiere ningún argumento:

```
api = ZabbixAPI()
```

Cerrar sesión

Si ha iniciado sesión utilizando un nombre de usuario y una contraseña, llame al método `logout()` después de completar sus operaciones de API:

```
from zabbix_utils import ZabbixAPI

api = ZabbixAPI(url="127.0.0.1/zabbix")
api.login(user="Admin", password="zabbix")

### Las acciones de la API van aquí

api.logout()
```

Note:
No es necesario llamar al método `logout()` cuando se utilizan tokens de API.

Solicitudes de API

Después de iniciar sesión, puede realizar cualquier solicitud de API llamando a los métodos descritos en la referencia de métodos de la API de Zabbix.

Los métodos de la API se llaman utilizando el siguiente formato:

```
api_instance.zabbix_object.method(parameters)
```

Note:
Algunos nombres de métodos u objetos de la API de Zabbix utilizan palabras que son palabras reservadas en Python (por ejemplo, `import`). Para evitar errores de Python, agregue un guion bajo al nombre del método u objeto cuando lo utilice en Python (por ejemplo, `api.configuration.import_`).

Por ejemplo, para ver una lista de hosts con `host.get`:

```
### 1. Importe ZabbixAPI desde zabbix_utils:
from zabbix_utils import ZabbixAPI

### 2. Cree la instancia de ZabbixAPI e inicie sesión:
api = ZabbixAPI(url="127.0.0.1/zabbix")
api.login(user="Admin", password="zabbix")

### 3. Recupere los hosts que coincidan con un filtro:
hosts = api.host.get(
    filter={
        "host": [
            "Zabbix server",
            "Linux server"
        ]
    }
)

### 4. Itere sobre los hosts devueltos e imprima los detalles de cada host:
for host in hosts:
    print(host)

### 5. Cierre la sesión de la API para cerrar la sesión:
api.logout()
```

Modo asíncrono

El modo asíncrono permite que tu script envíe solicitudes a la API sin esperar a que cada una termine. Esto puede hacer que tu script sea más eficiente cuando necesita realizar muchas solicitudes a la API o cuando algunas solicitudes tardan mucho tiempo.

Al utilizar el modo asíncrono, existen diferencias importantes en comparación con el modo síncrono:

- Importa el módulo `asyncio` de Python (primero debes instalar las dependencias requeridas).
- Importa `AsyncZabbixAPI` en lugar de `ZabbixAPI`.
- Escribe tu código dentro de una función `async`.
- Utiliza `await` al llamar a los métodos de la API, incluyendo `login()` y `logout()`.

> **Attention:**
> Al crear una instancia de `AsyncZabbixAPI`, no puedes iniciar sesión durante la inicialización.

Por ejemplo, para ver una lista de hosts con `host.get` en modo asíncrono:

```
### 1. Importa asyncio para el modo asíncrono y AsyncZabbixAPI desde zabbix_utils:
import asyncio
from zabbix_utils import AsyncZabbixAPI

### 2. Define la función principal async donde se ejecutarán todas las operaciones de la API:
async def main():

    # 3. Crea una instancia de AsyncZabbixAPI e inicia sesión (debes usar await):
    api = AsyncZabbixAPI(url="127.0.0.1")
    await api.login(user="User", password="zabbix")

    # 4. Recupera los hosts que coincidan con un filtro (debes usar await):
    hosts = await api.host.get(
        filter={
            "host": [
                "Zabbix server",
                "Linux server"
            ]
        }
    )

    # 5. Itera sobre los hosts devueltos e imprime los detalles de cada host:
```

```
    for host in hosts:
        print(host)

    # 6. Cierra la sesión en la API para cerrar la sesión (debes usar await):
    await api.logout()

### 7. Ejecuta la función async main() usando el bucle de eventos de asyncio:
asyncio.run(main())
```

Ejemplos

Los siguientes ejemplos muestran tareas comunes de la API de Zabbix utilizando la biblioteca zabbix_utils.

Hay ejemplos adicionales disponibles en el directorio `examples` del repositorio zabbix_utils en GitHub.

Recopilar datos del agent de Zabbix

Descripción general

zabbix_utils le permite recopilar datos del agent de Zabbix (de manera similar a Zabbix get).

Los datos se pueden recopilar en modo síncrono o asíncrono:

- En modo síncrono, su script de Python solicita y espera los datos antes de continuar, lo que es adecuado para operaciones simples, secuenciales y predecibles.
- En modo asíncrono, el script solicita datos sin esperar cada respuesta, lo que permite que otras operaciones continúen en paralelo; esto es más eficiente para solicitudes lentas o grandes lotes de datos.

Los ejemplos de esta página se centran en el modo síncrono, aunque el modo asíncrono sigue patrones similares. Hay ejemplos adicionales disponibles en el repositorio de GitHub zabbix_utils.

Importar

Para usar zabbix_utils para recopilar valores de item, importe la clase `Getter` en su script de Python:

```
from zabbix_utils import Getter
```

Datos de solicitud

Para solicitar el valor de un item:

1. Cree una instancia de `Getter`, especificando la dirección IP y el puerto de su agent de Zabbix.
2. Llame al método `get()` en la instancia de `Getter`, especificando la clave del item que desea recuperar.

Por ejemplo, para solicitar datos del item `system.uname`:

```
agent = Getter(host='192.0.2.1', port=10050)
response = agent.get('system.uname')
```

Uso de una IP no predeterminada

Si el server que ejecuta su script tiene varias direcciones IP, puede especificar una `source_ip` para que `Getter` la use al conectarse al agent de Zabbix:

```
agent = Getter(
    host='192.0.2.1',
    port=10050,
    source_ip='10.10.7.1'
)
```

Uso del tiempo de espera

Puede establecer un `timeout` de respuesta para `Getter` para controlar cuánto tiempo debe esperar su script una respuesta del agent de Zabbix antes de desistir:

```
agent = Getter(
    host='192.0.2.1',
    port=10050,
    timeout=30
)
```

Uso de cifrado

Getter no incluye compatibilidad integrada con cifrado, pero puede proporcionarla creando un wrapper con bibliotecas de terceros:

```
def psk_wrapper(sock, tls):
    # ...
    # Implementación del wrapper TLS PSK para el socket
    # ...

agent = Getter(
    host='192.0.2.1',
    port=10050,
    socket_wrapper=psk_wrapper
)
```

Respuesta

La respuesta del agent de Zabbix es procesada por la biblioteca y devuelta como un objeto AgentResponse:

```
print(response)
### {
###     "error": null,
###     "raw": "Linux zabbix_server 5.15.0-3.60.5.1.el9uek.x86_64",
###     "value": "Linux zabbix_server 5.15.0-3.60.5.1.el9uek.x86_64"
### }

print(response.value)
### Linux zabbix_server 5.15.0-3.60.5.1.el9uek.x86_64

print(response.error)
### None
```

Modo asíncrono

El modo asíncrono permite que tu script recopile valores sin esperar a que llegue cada uno. Esto puede hacer que tu script sea más eficiente cuando necesita recopilar muchos valores o cuando algunos valores tardan mucho en recopilarse.

Al usar el modo asíncrono, hay diferencias importantes en comparación con el modo síncrono:

- Importa el módulo asyncio de Python (primero debes instalar las dependencias requeridas).
- Importa AsyncGetter en lugar de Getter.
- Escribe tu código dentro de una función async.
- Usa await al llamar al método get().

Por ejemplo, para recopilar un solo valor usando el modo asíncrono:

```
### 1. Importa asyncio para el modo asíncrono, y AsyncGetter desde zabbix_utils:
import asyncio
from zabbix_utils import AsyncGetter

### 2. Define la función principal async donde se ejecutarán todas las solicitudes de datos:
async def main():
    agent = AsyncGetter(host='192.0.2.1', port=10050)

    # 3. Obtén el valor system.uname desde el agent de Zabbix (debe usarse await):
    response = await agent.get('system.uname')

    # 4. Imprime el valor devuelto por el agent de Zabbix:
    print(response.value)

### 5. Ejecuta la función main() async usando el event loop de asyncio:
asyncio.run(main())
```

Enviar datos al servidor o proxy de Zabbix

zabbix_utils le permite enviar valores de item a un item trapper en el servidor o proxy de Zabbix (de forma similar a Zabbix sender).

Puede enviar un único valor, varios valores o incluso dirigirse a varios clústeres de Zabbix.

Los datos pueden enviarse en modo síncrono o asíncrono:

- En modo síncrono, su script de Python envía valores y espera una respuesta antes de continuar; esto es adecuado para operaciones simples, secuenciales y predecibles.
- En modo asíncrono, el script envía valores sin esperar cada respuesta, lo que permite que otras operaciones continúen en paralelo; esto es más eficiente para solicitudes lentas o grandes lotes de datos.

Los ejemplos de esta página se centran en el modo síncrono, aunque el modo asíncrono sigue patrones similares. Hay ejemplos adicionales disponibles en el repositorio de GitHub de zabbix_utils.

Importar

Para utilizar zabbix_utils para enviar valores de item, importe la clase `Sender` en su script:

```
from zabbix_utils import Sender
```

Para enviar múltiples valores, también puede importar la clase `ItemValue`:

```
from zabbix_utils import Sender, ItemValue
```

Enviar un único valor

Para enviar el valor de un item:

1. Cree una instancia de `Sender`, especificando la dirección IP y el puerto de su Zabbix server o proxy.
2. Llame al método `send_value()` en la instancia de `Sender` utilizando el siguiente formato:

```
sender_instance.send_value('host', 'item.key', 'value', optional_timestamp, optional_nanoseconds)
```

Por ejemplo, para enviar 1 al item trapper `service.status` en el host `Linux server`:

```
sender = Sender(server='127.0.0.1', port=10051)
response = sender.send_value('Linux server', 'service.status', 1)
```

Usar una IP no predeterminada

Si el servidor que ejecuta su script tiene varias direcciones IP, puede especificar una `source_ip` para que el `Sender` la utilice al enviar valores al servidor o proxy de Zabbix:

```
sender = Sender(
    server='127.0.0.1',
    port=10051,
    source_ip='10.10.7.1'
)
```

Usando timeout

Puede establecer un `timeout` de respuesta para el `Sender` para controlar cuánto tiempo debe esperar su script una respuesta del server o proxy de Zabbix antes de rendirse:

```
sender = Sender(
    server='127.0.0.1',
    port=10051,
    timeout=30
)
```

Uso del archivo de configuración del agent

Puede permitir que `zabbix_utils` lea los parámetros `Server` o `ServerActive` desde un archivo de configuración local del agent o agent 2 de Zabbix. En estos casos, no es necesario especificar los parámetros de conexión al crear una instancia de `Sender`:

```
sender = Sender(
    use_config=True,
    config_path='/etc/zabbix/zabbix_agent2.conf'
)
```

Attention:
Si ServerActive contiene uno o más clústeres de Zabbix con varias instancias de server, Sender envía los datos al primer server disponible de cada clúster. Si no se establece ServerActive, se utiliza la dirección de Server con el puerto predeterminado (10051).

Usando cifrado

El Sender no incluye soporte de cifrado incorporado, pero puede proporcionarlo creando un wrapper usando librerías de terceros:

```
def psk_wrapper(sock, tls):
    # ...
    # Implementación de un wrapper TLS PSK para el socket
    # ...

sender = Sender(
    server='127.0.0.1',
    port=10051,
    socket_wrapper=psk_wrapper
)
```

Respuesta para un solo valor

La respuesta devuelta por el server o proxy de Zabbix es procesada por la biblioteca y devuelta como un objeto TrapperResponse:

```
print(response)
### {"processed": 1, "failed": 0, "total": 1, "time": "0.000123", "chunk": 1}

print(response.processed)
### 1

print(response.failed)
### 0

print(response.total)
### 1
```

Modo asíncrono

El modo asíncrono permite que tu script de Python envíe valores sin esperar una respuesta del servidor o proxy de Zabbix. Esto puede hacer que tu script sea más eficiente cuando necesita enviar muchos valores o cuando algunos valores tardan mucho en enviarse.

Al utilizar el modo asíncrono, hay algunas diferencias importantes en comparación con el modo síncrono:

- Importa el módulo asyncio de Python (primero debes instalar las dependencias requeridas).
- Importa AsyncSender en lugar de Sender.
- Escribe tu código dentro de una función async.
- Usa await al llamar al método send_value().

Por ejemplo, para enviar un solo valor utilizando el modo asíncrono:

```
### 1. Importa asyncio para el modo asíncrono y AsyncSender desde zabbix_utils:
import asyncio
from zabbix_utils import AsyncSender

### 2. Define la función principal async donde se ejecutarán todas las operaciones de envío de datos (debe
async def main():
    sender = AsyncSender(server='127.0.0.1', port=10051)
    response = await sender.send_value('Linux server', 'service.status', 1)

    # 3. Imprime la respuesta devuelta por el servidor o proxy de Zabbix:
    print(response)

### 4. Ejecuta la función async main() usando el bucle de eventos de asyncio:
asyncio.run(main())
```

Enviar múltiples valores

Para enviar múltiples valores:

1. Prepare un array de objetos `ItemValue`, cada uno utilizando el mismo formato que el método `send_value()`.
2. Cree una instancia de `Sender`, especificando la dirección IP y el puerto de su Zabbix server o proxy.
3. Llame al método `send()` (en lugar de `send_value()`) en la instancia de `Sender`, especificando el array de objetos con los valores a enviar.

Por ejemplo, para enviar cinco valores a diferentes hosts:

```
items = [
    ItemValue('server-de', 'service.status', 'up', 1770887205, 100),
    ItemValue('server-fr', 'service.status', 'up', 1770887205, 100),
    ItemValue('server-uk', 'service.status', 'up', 1770887205, 100),
    ItemValue('server-nl', 'service.status', 'up', 1770887205, 100),
    ItemValue('server-pl', 'service.status', 'up', 1770887205, 100),
]

sender = Sender(server='127.0.0.1', port=10051)
response = sender.send(items)
```

Usar un tamaño de fragmento personalizado

Si necesita enviar más valores de los que un item trapper puede aceptar en una sola solicitud, puede dividirlos en fragmentos.

Por defecto, el tamaño del fragmento es de 250 valores. Puede cambiarlo estableciendo el parámetro `chunk_size` al crear una instancia de `Sender`.

Por ejemplo, para enviar cinco valores en tres fragmentos (2-2-1), establezca el parámetro `chunk_size` en 2:

```
items = [
    ItemValue('server-de', 'service.status', 'up'),
    ItemValue('server-fr', 'service.status', 'up'),
    ItemValue('server-uk', 'service.status', 'up'),
    ItemValue('server-nl', 'service.status', 'up'),
    ItemValue('server-pl', 'service.status', 'up'),
]

sender = Sender(server='127.0.0.1', port=10051, chunk_size=2)
response = sender.send(items)
```

Enviar valores a varios clústeres de Zabbix

Para enviar valores a varios clústeres de Zabbix:

1. Prepare una matriz de clústeres de Zabbix. Si un clúster tiene varios nodos, el valor se enviará al primer nodo **disponible** de cada clúster.
2. Cree un `Sender`, especificando su matriz de clústeres de Zabbix.
3. Llame al método `send_value()` en la instancia de `Sender` utilizando el mismo formato que el método `send_value()`.

Por ejemplo, para enviar un valor al primer nodo disponible en cada clúster:

```
zabbix_clusters = [
    ['zabbix.cluster1.node1', 'zabbix.cluster1.node2:10051'],
    ['zabbix.cluster2.node1:10051', 'zabbix.cluster2.node2', 'zabbix.cluster2.node3']
]

sender = Sender(clusters=zabbix_clusters)
response = sender.send_value('Linux server', 'service.status', 1)
```

Respuesta para múltiples valores

Por defecto, `Sender` devuelve un resultado agregado del envío de valores a través de todos los hosts o clústeres:

```
print(response)
### {"processed": 2, "failed": 0, "total": 2, "time": "0.000108", "chunk": 2}
```

Si necesita información más detallada, puede inspeccionar los resultados de cada clúster y cada chunk utilizando el atributo `response.details`:

```
print(response)
### {"processed": 2, "failed": 0, "total": 2, "time": "0.000108", "chunk": 2}
```

```
if response.failed == 0:
    print(f"Value sent successfully in {response.time}")
else:
    print(response.details)
    # {
    #     127.0.0.1:10051: [
    #         {
    #             "processed": 1,
    #             "failed": 0,
    #             "total": 1,
    #             "time": "0.000051",
    #             "chunk": 1
    #         }
    #     ],
    #     zabbix.example.local:10051: [
    #         {
    #             "processed": 1,
    #             "failed": 0,
    #             "total": 1,
    #             "time": "0.000057",
    #             "chunk": 1
    #         }
    #     ]
    # }

    for node, chunks in response.details.items():
        for resp in chunks:
            print(f"Processed {resp.processed} of {resp.total} at {node.address}:{node.port}")
            # Processed 1 of 1 at 127.0.0.1:10051
            # Processed 1 of 1 at zabbix.example.local:10051
```

Registro de depuración

Para solucionar problemas al trabajar con la API de Zabbix, recopilar datos del agent de Zabbix o enviar datos al server o proxy de Zabbix, puede habilitar el registro de depuración.

La biblioteca zabbix_utils utiliza el módulo estándar `logging` de Python.

Para ver mensajes de depuración detallados de la biblioteca:

1. Importe el módulo `logging` de Python en su script.
2. Configure el sistema de registro para que muestre los mensajes en un formato legible y establezca el nivel de registro en DEBUG.

```
import logging
from zabbix_utils import Getter

logging.basicConfig(
    format=u'[%(asctime)s] %(levelname)s %(message)s',
    level=logging.DEBUG
)

agent = Getter(host='127.0.0.1', port=10050)
resp = agent.get('system.uname')

print(resp.value)
```

Con el registro de depuración habilitado, se imprime información detallada sobre las solicitudes y respuestas, por ejemplo:

```
[2023-10-01 12:00:01,587] DEBUG Content of the packet: b'ZBXD\x01\x0c\x00\x00\x00\x00\x00\x00\x00system.un
[2023-10-01 12:00:01,722] DEBUG Zabbix response header: b'ZBXD\x01C\x00\x00\x00C\x00\x00\x00'
[2023-10-01 12:00:01,723] DEBUG Zabbix response body: Linux test_server 5.15.0-3.60.5.1.el9uek.x86_64
```

```
[2023-10-01 12:00:01,724] DEBUG Response from [127.0.0.1:10050]: Linux test_server 5.15.0-3.60.5.1.el9uek.
Linux test_server 5.15.0-3.60.5.1.el9uek.x86_64
```

Módulos

¿Qué es un módulo de frontend PHP?

- Un módulo es una entidad con un ID único, nombre, descripción, autor y otros campos definidos en su archivo de manifiesto, junto con archivos PHP, Javascript y otros ubicados en un único directorio dentro del directorio *modules* de su instalación de frontend de Zabbix (por ejemplo, *zabbix/ui/modules*).
- Un módulo debe cumplir reglas simples para garantizar su correcto funcionamiento.
- Un módulo debe ser instalado (descomprimido) y habilitado en el frontend por el administrador.

Para qué se puede usar un módulo

- Añadir nueva funcionalidad mediante secciones personalizadas en la interfaz web;
- Crear tipos de widgets personalizados para el panel (ver módulos de widgets);
- Sobrescribir o ampliar la funcionalidad existente.

Para qué no se puede usar un módulo

- Registrar un nuevo método de API o modificar uno existente.

Cómo funcionan los módulos

- Un módulo habilitado se inicia en cada solicitud HTTP, antes de ejecutar el código de la acción.
- El módulo registrará nuevas acciones o redefinirá las existentes.
- El módulo añadirá nuevas secciones en la interfaz y eliminará o redefinirá las existentes.
- El módulo se enganchará a eventos de la interfaz como onBeforeAction y onTerminate, si es necesario.
- Finalmente, la acción solicitada se ejecuta ejecutando el código de la acción, ya sea el predeterminado o el definido por el módulo.

¿Dónde ir a continuación? Ya sea que prefiera aprender haciendo o leer las directrices primero, estas páginas contienen la información y los pasos necesarios para crear sus propios módulos:

- Tutoriales paso a paso para escribir su primer módulo
- Estructura de archivos del módulo
- Especificaciones del módulo de widget
- Ejemplos de módulos para reutilizar

Estructura de archivos del módulo

Todo el código relacionado con un módulo se almacena en un único directorio dentro del directorio **modules** de su instalación del frontend de Zabbix (por ejemplo, *zabbix/ui/modules*).

```
example_module_directory/          (obligatorio)
   manifest.json                   (obligatorio)  Metadatos y definición de acciones.
   Module.php                                     Inicialización del módulo y manejo de eventos.
   actions/                                       Archivos de controladores de acciones.
       SomethingView.php
       SomethingCreate.php
       SomethingDelete.php
       data_export/
           ExportAsXml.php
           ExportAsExcel.php
   views/                                         Archivos de vistas.
       example.something.view.php
       example.something.delete.php
```

```
   assets/                                  Cualquier archivo adicional para ser usado en las vistas. D
       js/                                  Archivos JavaScript usados en las vistas.
           example.something.view.js.php
       css/                                 Archivos CSS usados en las vistas.
           example.something.css
        image.png                           Imágenes usadas en las vistas.
        example.something.file              Cualquier archivo para usar en las vistas.
```

Árbol de archivos del módulo

Escribir un módulo Un proceso de escritura de un módulo de ejemplo consta de los siguientes pasos (cuando esté disponible, haga clic en el nombre del archivo o de la carpeta para ver detalles adicionales sobre el paso):

1. Cree un nuevo directorio para el módulo dentro de **zabbix/ui/modules/**.
2. Añada el archivo manifest.json con los metadatos del módulo.
3. Cree la carpeta *views* y defina una o varias vistas del módulo.
4. Cree la carpeta *actions* y defina una o varias acciones del módulo.
5. Cree el archivo Module.php (o Widget.php para widgets del dashboard) y defina las reglas de inicialización y de gestión de eventos.
6. Cree la carpeta *assets* para archivos JavaScript (colóquelos en *assets/js*), estilos CSS (colóquelos en *assets/css*) o cualquier otro archivo adicional.
7. Asegúrese de especificar las vistas, acciones y archivos de assets requeridos en manifest.json.
8. Registre el módulo en el frontend de Zabbix y empiece a usarlo.

Note:
Puede registrar y habilitar un módulo en cuanto cree el archivo manifest.json. Una vez habilitado el módulo, puede previsualizar de inmediato todos los cambios realizados en los archivos del módulo actualizando el frontend de Zabbix.

manifest.json

Cualquier módulo necesita el archivo manifest.json. El archivo debe estar ubicado en el directorio principal del módulo (por ejemplo, *zabbix/ui/modules/module_name/manifest.json*).

Como mínimo, manifest.json debe especificar estos campos:

```
{
    "manifest_version": 2.0,
    "id": "my_ip_address",
    "name": "My IP Address",
    "namespace": "MyIPAddress",
    "version": "1.0"
}
```

Parámetros admitidos en **manifest.json** (haga clic en el nombre del parámetro para ver una descripción detallada):

Parameter	Description	Required
manifest_version	Versión del manifiesto del módulo.	Yes
id	ID único del módulo.	
name	Nombre del módulo que se mostrará en la sección de Administración.	
namespace	Espacio de nombres de PHP para las clases del módulo.	
version	Versión del módulo.	
type	Tipo del módulo. Para widget debe establecerse en *widget*	Yes for widgets, otherwise no
widget	Configuración del widget. Se usa solo para widgets.	
actions	Acciones que se registrarán con el módulo.	
assets	Archivos CSS y JavaScript que se incluirán.	No
author	Autor del módulo.	
config	Valores predeterminados para opciones personalizadas del módulo.	
description	Descripción del módulo.	
url	Un enlace a la descripción del módulo.	

manifest_version

Versión del manifiesto del módulo. Actualmente, la versión soportada es **2.0**.

Tipo: Doble

Ejemplo:

```
"manifest_version": 2.0
```

id

ID del módulo. Debe ser único. Para evitar conflictos de nombres en el futuro, se recomienda utilizar un prefijo para los módulos (autor o nombre de la empresa, o cualquier otro). Por ejemplo, si un módulo es un ejemplo para lecciones y el nombre del módulo es "Mi módulo", entonces el ID será "example_my_module".

Tipo: Cadena

Ejemplo:

```
"id": "example_my_module"
```

name

Nombre del módulo que se mostrará en la sección de Administración.

Tipo: Cadena

Ejemplo:

```
"name": "Mi módulo"
```

namespace

Espacio de nombres PHP para las clases del módulo.

Tipo: Cadena

Ejemplo:

```
"namespace": "ClockWidget"
```

versión

Versión del módulo. La versión se mostrará en la sección de Administración.

Tipo: Cadena

Ejemplo:

```
"version": "1.0"
```

type

Tipo de módulo. Obligatorio para widgets y debe ser igual a "widget".

Tipo: Cadena

Por defecto: "module"

Ejemplo:

```
"type": "widget"
```

acciones

Acciones para registrar con el módulo. Definir la clave de objeto *class* para cada acción es obligatorio, otras claves de acción son opcionales.

Tipo: Objeto

Claves de objeto soportadas si *type* es *module*:

- **write.your.action.name** (objeto) - nombre de la acción, debe escribirse en minúsculas [a-z], separando las palabras con puntos. Soporta las claves:
  - **class** (cadena; obligatorio) - nombre de la clase de la acción.
  - **layout** (cadena) - diseño de la acción. Valores soportados: *layout.json*, *layout.htmlpage* (por defecto), *null*.
  - **view** (cadena) - vista de la acción.

Ejemplo:

```
"actions": {
    "module.example.list": {
        "class": "ExampleList",
        "view": "example.list",
        "layout": "layout.htmlpage"
        }
    }
```

Claves de objeto soportadas si *type* es *widget*:

- **widget.{id}.view** (objeto) - nombre de archivo y clase para la vista del widget. Reemplace **{id}** con el valor id del widget (por ejemplo, *widget.example_clock.view*). Soporta las claves:
  - **class** (cadena; obligatorio) - nombre de la clase de la acción para el modo de vista del widget para extender la clase por defecto CControllerDashboardWidgetView. El archivo fuente de la clase debe estar ubicado en el directorio *actions*.
  - **view** (cadena) - vista del widget. Debe estar ubicada en el directorio *views*. Si el archivo de vista es *widget.view.php*, que es el esperado por defecto, este parámetro puede omitirse. Si se utiliza un nombre diferente, especifíquelo aquí.
- **widget.{id}.edit** (objeto) - nombre de archivo para la vista de configuración del widget. Reemplace **{id}** con el valor id del widget (por ejemplo, *widget.example_clock.edit*). Soporta las claves:
  - **class** (cadena; obligatorio) - nombre de la clase de la acción para el modo de vista de configuración del widget. El archivo fuente de la clase debe estar ubicado en el directorio *actions*.
  - **view** (cadena) - vista de configuración del widget. Debe estar ubicada en el directorio *views*. Si el archivo de vista es *widget.edit.php*, que es el esperado por defecto, este parámetro puede omitirse. Si se utiliza un nombre diferente, especifíquelo aquí.

Ejemplo:

```
"actions": {
    "widget.tophosts.view": {
        "class": "WidgetView"
    },
    "widget.tophosts.column.edit": {
        "class": "ColumnEdit",
        "view": "column.edit",
        "layout": "layout.json"
    }
}
```

assets

Archivos de estilos CSS y JavaScript para incluir.

Tipo: Objeto

Claves de objeto soportadas:

- **css** (array) - Archivos CSS para incluir. Los archivos deben estar ubicados en *assets/css*.
- **js** (array) - Archivos JavaScript para incluir. Los archivos deben estar ubicados en *assets/js*.

Ejemplo:

```
"assets": {
    "css": ["widget.css"],
    "js": ["class.widget.js"]
}
```

autor

Autor del módulo. El autor se mostrará en la sección de Administración.

Tipo: Cadena

Ejemplo:

```
"author": "John Smith"
```

config

Valores predeterminados para las opciones del módulo. El objeto puede contener cualquier clave personalizada. Si se especifican, estos valores se escribirán en la base de datos durante el registro del módulo. Las nuevas variables añadidas posteriormente se escribirán en la primera llamada. Posteriormente, los valores de las variables solo se pueden cambiar directamente en la base de datos.

Tipo: Objeto

Ejemplo:

```
"config": {
    "username": "Admin",
    "password": "",
    "auth_url": "https://example.com/auth"
}
```

description

Descripción del módulo.

Tipo: Cadena

Ejemplo:

```
"description": "Este es un widget de reloj."
```

widget

Configuración del widget. Se utiliza si *type* está establecido en *widget*.

Tipo: Objeto

Claves de objeto soportadas:

- **name** (string) - se utiliza en la lista de widgets y como encabezado por defecto. Si está vacío, se usará el parámetro "name" del módulo.
- **size** (object) - dimensiones por defecto del widget. Soporta las claves:
  - *width* (integer) - ancho por defecto del widget.
  - *height* (integer) - alto por defecto del widget.
- **form_class** (string) - clase con el formulario de campos del widget. Debe estar ubicada en el directorio *includes*. Si la clase es *WidgetForm.php*, que es la esperada por defecto, este parámetro puede omitirse. Si se utiliza un nombre diferente, especifíquelo aquí.
- **js_class** (string) - nombre de una clase JavaScript para el modo de vista del widget que extiende la clase CWidget por defecto. La clase se cargará con el dashboard. El archivo fuente de la clase debe estar ubicado en el directorio assets/js. Véase también: assets.
- **use_time_selector** (boolean) - determina si el widget requiere el selector de tiempo del dashboard. Valores soportados: *true*, *false* (por defecto).
- **refresh_rate** (integer) - intervalo de actualización del widget en segundos (por defecto: 60).

Ejemplo:

```
"widget": {
    "name": "",
    "size": {
        "width": 12,
        "height": 5
    },
    "form_class": "WidgetForm",
    "js_class": "CWidget",
    "use_time_selector": false,
    "refresh_rate": 60
}
```

url

Un enlace a la descripción del módulo. Para los widgets, este enlace se abrirá al hacer clic en el icono de ayuda ? en la ventana *Añadir widget* o *Editar widget*. Si no se especifica **url**, al hacer clic en el icono de ayuda se abrirá la página general Widgets del dashboard.

Tipo: String

Ejemplo:

```
"url": "http://example.com"
```

Acciones

Las acciones son responsables de la 'lógica de negocio' del módulo. Una acción generalmente consiste en un controlador y una vista de acción.

Un módulo puede:

- Llamar a acciones que ya están definidas en el frontend de Zabbix.
- Sobrescribir acciones por defecto con acciones personalizadas.
- Definir acciones completamente nuevas.

Para sobrescribir el comportamiento de una acción por defecto con algún comportamiento personalizado, defina una acción con el mismo nombre en la configuración del módulo. Cuando se llame a la acción, se ejecutará la acción del módulo en lugar de la acción por defecto de Zabbix.

Los archivos de acciones deben almacenarse en la carpeta *actions*. Las acciones deben especificarse en el manifest.json.

Controller

Flujo de trabajo del controller:

1. Comprobar que todos los parámetros pasados en una solicitud HTTP sean válidos:
   - Llamar al método *checkInput()* del controller;
   - Usar las reglas de validación definidas en CNewValidator.php;
   - Llamar al método *validateInput()*.
2. Comprobar los permisos del usuario.
3. Preparar los datos según los parámetros pasados: si *checkInput()* devuelve true, Zabbix llama al método *doAction()* del controller.
4. Preparar el array **$data** para la vista. Use *CControllerResponseData* y el método *setResponse()* para almacenar la respuesta en el array **$data**.

Ejemplo:

```
/**
 * Validate input parameters.
 *
 * @return bool
 */
protected function checkInput(): bool {
    $ret = $this->validateInput([
        'status' => 'in '.implode(',', [HOST_STATUS_MONITORED, HOST_STATUS_NOT_MONITORED])
    ]);

    if (!$ret) {
        $this->setResponse(new CControllerResponseFatal());
    }

    return $ret;
}

/**
 * Check user permissions.
 *
 * @return bool
 */
protected function checkPermissions() {
    return $this->getUserType() >= USER_TYPE_ZABBIX_ADMIN;
}

/**
 * Execute action and generate response object.
 */
```

```
protected function do Action(): void {
    $data = [
        'hosts_count' => API::Host()->get([
            'countOutput' => true,
            'filter' => [
                'status' => $this->getInput('status')
            ]
        ])
    ];

    $this->setResponse(new CControllerResponseData($data));
}
```

Note:
Puede ver la lista completa de clases de controller disponibles en el código fuente de Zabbix.

Vistas

El archivo de vista recibe los datos de un controlador y luego prepara su apariencia HTML.

Note:
Definir vista(s) para un módulo de frontend es opcional, a menos que el módulo sea un widget.
 Los widgets del dashboard necesitan al menos dos vistas: una para el modo de edición y otra para el modo de visualización (deben almacenarse en el directorio *views*).

Es posible utilizar clases HTML predefinidas de Zabbix (de */zabbix/ui/include/classes/html*) en la vista, así como añadir nuevas clases HTML y CSS. Las nuevas clases deben almacenarse en la carpeta *assets* del módulo.

Ejemplo:

```
...
(new CColHeader(_('Name')))
```

Esto añadirá un nuevo nombre de columna *Name* y aplicará estilo a la fila superior de la tabla como en otras páginas de Zabbix.

Vista de acción

Este es un archivo de referencia para definir una vista de acción.

```
<?php declare(strict_types = 1);

/**
* @var CView $this
*/

$this->includeJsFile('example.something.view.js.php');

(new CWidget())
    ->setTitle(_('Vista de algo'))
    ->addItem(new CDiv($data['name']))
    ->addItem(new CPartial('module.example.something.reusable', [
        'contacts' => $data['contacts']
    ]))
    ->show();
```

Recursos

La carpeta *assets* puede contener cualquier archivo y subcarpeta que no pertenezca a otros directorios. Puede utilizarla para:

- Estilos JavaScript (deben estar dentro de *assets/js*);
- Estilos CSS (deben estar dentro de *assets/css*);
- Imágenes;
- Fuentes;

- Cualquier otra cosa que necesite incluir.

assets/js

El directorio *assets/js* está reservado y solo debe contener archivos JavaScript. Para que sean utilizados por el widget, especifique estos archivos en el *manifest.json*.

Por ejemplo:

```
"assets": {
    "js": ["class.widget.js"]
}
```

assets/css

assets/css está reservado y solo debe contener archivos de estilos CSS. Para que el widget los utilice, especifique estos archivos en el *manifest.json*.

Por ejemplo:

```
"assets": {
    "css": ["mywidget.css"]
}
```

Estilos CSS

Los archivos CSS pueden contener un atributo personalizado `theme` para definir un estilo diferente para un tema específico del frontend.

Temas disponibles y sus valores de atributo:

- **Azul** - [theme='blue-theme']
- **Azul (clásico)** - [theme='blue-classic-theme']
- **Oscuro** - [theme='dark-theme']
- **Azul oscuro** - [theme='dark-blue-theme']
- **Oscuro (clásico)** - [theme='dark-classic-theme']
- **Claro de alto contraste** - [theme='hc-light']
- **Oscuro de alto contraste** - [theme='hc-dark']

Ejemplo:

```
.widget {
    background-color: red;
}

[theme='dark-theme'] .widget {
    background-color: green;
}
```

Registrar un nuevo módulo

Esta sección explica cómo añadir un nuevo módulo al frontend de Zabbix.

Requisitos previos Antes de continuar, asegúrese de que:

- El módulo se encuentra dentro del directorio *modules* de su instalación de la interfaz web de Zabbix (por ejemplo, *zabbix/ui/modules*).
- El módulo tiene al menos una versión básica del archivo manifest.json.
- Tiene acceso a la sección de menú Administración en Zabbix (requiere el tipo de rol de usuario Super admin).

> **Note:**
> La interfaz web no instalará ni siquiera reconocerá módulos incompatibles.

Agregar un módulo Abra la página *Administration* > *General* > *Modules* y haga clic en *Scan directory*.



Localice su módulo en la lista y actívelo.

Para activar un módulo, haga clic en el hipervínculo *Disabled*; el estado del módulo cambiará a *Enabled*.

Haga clic en el nombre del módulo para ver información adicional sobre el módulo, como el autor, la versión o una breve descripción (si está definida en el manifiesto).

Vista previa del widget Los módulos de widget, una vez añadidos, se vuelven inmediatamente visibles en la lista de widgets del dashboard.

Puede abrir un dashboard, cambiar al modo de edición y añadir el widget al dashboard como de costumbre.

Cuando realice algún cambio en el widget, actualice el dashboard para ver cómo se ve el widget con las actualizaciones más recientes.

Widgets

Los widgets son módulos del frontend de Zabbix que se utilizan en los paneles. A menos que se indique lo contrario, todas las directrices de los módulos también son aplicables a los widgets.

Sin embargo, un widget es notablemente diferente de un módulo. Para construir un widget:

- especifique el tipo "widget" en el archivo manifest.json ("type": "widget");
- incluya al menos dos vistas: una para el modo de presentación del widget y otra para el modo de configuración del widget (example.widget.view.php y example.widget.edit.php);
- y un controlador para la presentación del widget (WidgetView.php);
- utilice y extienda las clases de widget por defecto.

Configuración

Esta página describe las clases que se pueden utilizar para crear una vista de configuración de widget con campos de configuración personalizados. La vista de configuración del widget es la parte del widget que permite al usuario configurar los parámetros del widget para la presentación.

Widget

Clase principal del widget, extiende la clase base de todos los widgets del panel - *CWidget*. Requerido para sobrescribir el comportamiento predeterminado del widget.

La clase *Widget* debe ubicarse en el directorio raíz del widget (por ejemplo, *zabbix/ui/modules/my_custom_widget*).

Ejemplo de Widget.php

```
<?php

namespace Modules\MyCustomWidget;

use Zabbix\Core\CWidget;

class Widget extends CWidget {

    public const MY_CONSTANT = 0;

    public function getTranslationStrings(): array {
        return [
            'class.widget.js' => [
                'No data' => _('No data')
            ]
        ];
    }
}
```

WidgetForm

La clase *WidgetForm* extiende la clase por defecto *CWidgetForm* y contiene un conjunto de campos *CWidgetField* que son necesarios para definir la estructura de almacenamiento de la configuración del widget en la base de datos y gestionar la validación de entrada.

La clase *WidgetForm* debe ubicarse en el directorio *includes*. Si la clase tiene un nombre diferente, el nombre debe especificarse en el parámetro *widget/form_class* en el archivo *manifest.json*.

Ejemplo de includes/WidgetForm.php

```
<?php

namespace Modules\MyCustomWidget\Includes;

use Modules\MyCustomWidget\Widget;

use Zabbix\Widgets\{
    CWidgetField,
    CWidgetForm
};

use Zabbix\Widgets\Fields\{
    CWidgetFieldMultiSelectItem,
    CWidgetFieldTextBox,
    CWidgetFieldColor
};


class WidgetForm extends CWidgetForm {

    public const DEFAULT_COLOR_PALETTE = [
        'FF465C', 'B0AF07', '0EC9AC', '524BBC', 'ED1248', 'D1E754', '2AB5FF', '385CC7', 'EC1594', 'BAE37D',
        '6AC8FF', 'EE2B29', '3CA20D', '6F4BBC', '00A1FF', 'F3601B', '1CAE59', '45CFDB', '894BBC', '6D6D6D'
    ];

    public function addFields(): self {
        return $this
            ->addField(
                (new CWidgetFieldMultiSelectItem('itemid', _('Métrica')))
                    ->setFlags(CWidgetField::FLAG_NOT_EMPTY | CWidgetField::FLAG_LABEL_ASTERISK)
                    ->setMultiple(false)
            )
            ->addField(
                new CWidgetFieldTextBox('description', _('Descripción'))
            )
            ->addField(
                (new CWidgetFieldColor('chart_color', _('Color')))->setDefault('FF0000')
            );
    }
}
```

CWidgetFormView

La clase *CWidgetFormView* es necesaria para especificar la lógica de presentación de los campos definidos en la clase *WidgetForm* y determinar su apariencia y comportamiento al representarse en la vista de configuración.

La clase *CWidgetFormView* admite los siguientes métodos:

- *addField()*: recibe como parámetro una instancia de la clase *CWidgetField*; cada clase *CWidgetField* tiene una clase *CWidgetFieldView* correspondiente para utilizarla en la vista de configuración del widget.
- *addFieldset()*: recibe una instancia de la clase *CWidgetFieldsGroupView*, que combina los campos en un contenedor contraíble.
- *addFieldsGroup()*: recibe una instancia de *CWidgetFormFieldsetCollapsibleView*, que combina visualmente los campos en un grupo mediante un borde.
- *includeJsFile()*: permite añadir un archivo JavaScript a la vista de configuración del widget.
- *addJavaScript()*: permite añadir JavaScript en línea que se ejecutará en cuanto se cargue la vista de configuración del widget.

La clase *CWidgetFormView* debe ubicarse en el directorio *views*.

Ejemplo de views/widget.edit.php

```php
<?php

/**
 * Vista del formulario del widget personalizado.
 *
 * @var CView $this
 * @var array $data
 */

use Modules\MyCustomWidget\Includes\WidgetForm;

(new CWidgetFormView($data))
    ->addField(
        (new CWidgetFieldMultiSelectItemView($data['fields']['itemid']))->setPopupParameter('numeric', tru
    )
    ->addFieldset(
        (new CWidgetFormFieldsetCollapsibleView(_('Configuración avanzada')))
            ->addField(
                new CWidgetFieldTextBoxView($data['fields']['description'])
            )
            ->addField(
                new CWidgetFieldColorView($data['fields']['chart_color'])
            )
    )
    ->includeJsFile('widget.edit.js.php')
    ->addJavaScript('my_custom_widget_form.init('.json_encode([
        'color_palette' => WidgetForm::DEFAULT_COLOR_PALETTE
    ]).');')
    ->show();
```

JavaScript

Se puede utilizar una clase JavaScript para añadir comportamiento dinámico e interactividad a la vista de configuración del widget. Por ejemplo, puede inicializar un selector de color, definido en la clase *CWidgetFormView*.

La clase JavaScript debe cargarse con el formulario, por lo tanto, debe ser referenciada en la clase *CWidgetFormView* utilizando los métodos *includeJsFile()* y *addJavaScript()*.

En el siguiente ejemplo, se crea inmediatamente una instancia de clase singleton y se almacena bajo el nombre *window.my_custom_widget_form*. Así, al abrir el formulario por segunda vez, se volverá a crear la instancia.

La clase JavaScript debe ubicarse en el directorio *views*.

Ejemplo de views/widget.edit.js.php

```php
<?php

use Modules\MyCustomWidget\Widget;

?>

window.my_custom_widget_form = new class {

    init({color_palette}) {
        colorPalette.setThemeColors(color_palette);

        for (const colorpicker of jQuery('.<?= ZBX_STYLE_COLOR_PICKER ?> input')) {
            jQuery(colorpicker).colorpicker();
        }

        const overlay = overlays_stack.getById('widget_properties');

        for (const event of ['overlay.reload', 'overlay.close']) {
            overlay.$dialogue[0].addEventListener(event, () => { jQuery.colorpicker('hide'); });
        }
    }
```

```
};
```

CWidgetField

La clase *CWidgetField* es una clase base de la que heredan todas las clases de campos de formulario (*CWidgetFieldCheckBox*, *CWidgetFieldTextArea*, *CWidgetFieldRadioButtonList*, etc.). Las clases que extienden *CWidgetField* son responsables de recibir, guardar y validar los valores de configuración del widget.

Están disponibles las siguientes clases *CWidgetField*.

CWidgetField class	Database field type	Description
CWidgetFieldCheckBox	int32	Casilla de verificación única.
CWidgetFieldCheckBoxList	array of int32	Varias casillas de verificación bajo un único campo de configuración.
CWidgetFieldColor	string	Campo de selección de color.
CWidgetFieldDatePicker	string	Campo de selección de fecha.
CWidgetFieldIntegerBox	int32	Campo para introducir un número entero. Se puede utilizar para configurar valores mínimos y máximos.
CWidgetFieldLatLng	string	Cuadro de texto para introducir la latitud, la longitud y el nivel de zoom del mapa, separados por comas.
CWidgetFieldMultiSelect	-	Clase base para los campos de selección múltiple, extendida por todas las clases `CWidgetFieldMultiSelect*`.
CWidgetFieldMultiSelectAction	ID	Campo de selección múltiple para seleccionar acciones de la lista de acciones definida en *Alerts > Actions*.
CWidgetFieldMultiSelectGraph	ID	Campo de selección múltiple para seleccionar gráficos personalizados.
CWidgetFieldMultiSelectGraphPrototype	ID	Campo de selección múltiple para seleccionar prototipos de gráficos personalizados.
CWidgetFieldMultiSelectGroup	ID	Campo de selección múltiple para seleccionar grupos de hosts.
CWidgetFieldMultiSelectHost	ID	Campo de selección múltiple para seleccionar hosts.
CWidgetFieldMultiSelectHostInventory	array of int32	Campo de selección múltiple para seleccionar campos del inventario de hosts.
CWidgetFieldMultiSelectItem	ID	Campo de selección múltiple para seleccionar items.
CWidgetFieldMultiSelectItemPrototype	ID	Campo de selección múltiple para seleccionar prototipos de items.
CWidgetFieldMultiSelectMap	ID	Campo de selección múltiple para seleccionar mapas.
CWidgetFieldMultiSelectMediaType	ID	Campo de selección múltiple para seleccionar tipos de medios.
CWidgetFieldMultiSelectOverrideHost	ID	Campo de selección múltiple para seleccionar una fuente de datos para los hosts desde un dashboard u otro widget.
CWidgetFieldMultiSelectService	ID	Campo de selección múltiple para seleccionar servicios.
CWidgetFieldMultiSelectSla	ID	Campo de selección múltiple para seleccionar SLA.
CWidgetFieldMultiSelectUser	ID	Campo de selección múltiple para seleccionar usuarios.
CWidgetFieldNavTree	array of (multiple mixed)	Almacena una estructura de árbol de navegación de mapas para utilizarla en el widget *Map navigation tree*.
CWidgetFieldNumericBox	string	Campo para introducir un número decimal.
CWidgetFieldPatternSelect	-	Clase base para los campos de selección de patrones, extendida por todas las clases `CWidgetPatternSelect*`.
CWidgetFieldPatternSelectHost	string	Campo de selección múltiple para seleccionar hosts. Admite patrones de nombres de host.
CWidgetFieldPatternSelectItem	string	Campo de selección múltiple para seleccionar items. Admite patrones de nombres de item.
CWidgetFieldRadioButtonList	int32	Grupo de botones de opción compuesto por uno o más botones de opción.
CWidgetFieldRangeControl	int32	Control deslizante para seleccionar un valor de tipo entero.
CWidgetFieldReference	string	Almacena un identificador único para este widget en el dashboard; se utiliza para la comunicación entre widgets.
CWidgetFieldSelect	int32	Cuadro de selección desplegable.
CWidgetFieldSeverities	array of int32	*CWidgetFieldCheckBoxList* preconfigurado con las severidades de los triggers.
CWidgetFieldSparkline	array of (multiple mixed)	Permite configurar un gráfico sparkline.
CWidgetFieldTags	array of (string, int32, string)	Permite configurar una o más filas de filtros de etiquetas.
CWidgetFieldTextArea	string	Área de texto para introducir texto de varias líneas.
CWidgetFieldTextBox	string	Cuadro de texto para introducir texto de una sola línea.

CWidgetField class	Database field type	Description
CWidgetFieldThresholds	array of (string, string)	Permite configurar pares de colores y números.
CWidgetFieldTimePeriod	array of string	Campo de selección del período de tiempo.
CWidgetFieldTimeZone	string	Lista desplegable con zonas horarias.
CWidgetFieldUrl	string	Cuadro de texto para introducir URL.

Note:
La longitud máxima de los campos `int32` es de 10 caracteres, la de los campos `ID` es de 20 caracteres y la de los campos `string` es de 65535 caracteres. Para los tipos compuestos (por ejemplo, `array of (string, int32, string)`), el límite se aplica a cada elemento individualmente. Las restricciones de longitud específicas de cada campo se pueden establecer mediante métodos como `setMaxLength()`.

Presentación

Esta página describe los componentes que se pueden utilizar para crear una vista de presentación de widget. La vista de presentación del widget es la parte del widget que recibe los datos según su configuración y los muestra en el panel en un contenedor.

La vista de presentación consta de tres partes:

- Acción del widget
- Vista del widget
- JavaScript

Acción de widget

La clase de acción de widget (*WidgetView*) contiene métodos para operaciones con widgets en el modo de vista de presentación. La mayoría de las acciones de widget utilizan y/o extienden la clase de controlador por defecto *CControllerDashboardWidgetView*.

La clase de acción de widget debe ubicarse en el directorio *actions* y especificarse en el parámetro *actions* (*actions/widget.{id}.view/class*) en el archivo *manifest.json*.

Ejemplo de actions/WidgetView.php (implementado en el widget nativo de Zabbix *Información del sistema*)

```
class WidgetView extends CControllerDashboardWidgetView {

    protected function doAction(): void {
        $this->setResponse(new CControllerResponseData([
            'name' => $this->getInput('name', $this->widget->getDefaultName()),
            'system_info' => CSystemInfoHelper::getData(),
            'info_type' => $this->fields_values['info_type'],
            'user_type' => CWebUser::getType(),
            'user' => [
                'debug_mode' => $this->getDebugMode()
            ]
        ]));
    }
}
```

Vista del widget

La clase de vista del widget (*CWidgetView*) es responsable de construir la vista de presentación del widget.

La clase de vista del widget debe ubicarse en el directorio *views*. Si el archivo que contiene la clase de vista del widget tiene un nombre diferente al predeterminado (*widget.view.php*), entonces debe especificarse en el parámetro *actions* del archivo *manifest.json* (*actions/widget.{id}.view/view*).

Ejemplo de views/widget.view.php

```
<?php

/**
 * Mi vista personalizada de widget.
 *
 * @var CView $this
 * @var array $data
```

```
 */

(new CWidgetView($data))
    ->addItem(
        new CTag('h1', true, $data['name'])
    )
    ->show();
```

JavaScript

La clase JavaScript es responsable de determinar el comportamiento del widget, como actualizar los datos del widget, cambiar el tamaño del widget, mostrar los elementos del widget, etc.

Todas las operaciones de JavaScript utilizan y/o extienden la clase base de JavaScript de todos los widgets del panel - *CWidget*. La clase *CWidget* contiene un conjunto de métodos con la implementación predeterminada para el comportamiento del widget. Dependiendo de la complejidad del widget, estos métodos pueden utilizarse tal cual o extenderse.

La clase *CWidget* contiene los siguientes métodos:

- Métodos que definen el ciclo de vida del widget: *onInitialize()*, *onStart()*, *onActivate()*, *onDeactivate()*, *onDestroy()*, *onEdit()*.
- Métodos que gestionan la actualización y visualización de los datos del widget: *promiseUpdate()*, *getUpdateRequestData()*, *processUpdateResponse(response)*, *processUpdateErrorResponse(error)*, *setContents(response)*.
- Métodos que modifican la apariencia del widget: *onResize()*, *hasPadding()*.

La clase JavaScript debe ubicarse en el directorio *assets/js* y especificarse en el parámetro *assets* (*assets/js*) en el archivo *manifest.json*.

Métodos del ciclo de vida

Los métodos del ciclo de vida del widget son invocados por el panel, y en diferentes etapas del ciclo de vida del widget durante su existencia dentro del panel.

El método ***onInitialize()*** define el estado inicial y/o los valores del widget, sin realizar ninguna manipulación de HTML o datos. Este método se invoca cuando se crea un widget (se instancia un objeto widget), normalmente al añadir el widget a una página de panel o al cargar la página del panel.

Ejemplo:

```
onInitialize() {
    this._time_offset = 0;
    this._interval_id = null;
    this._clock_type = CWidgetClock.TYPE_ANALOG;
    this._time_zone = null;
    this._show_seconds = true;
    this._time_format = 0;
    this._tzone_format = 0;
    this._show = [];
    this._has_contents = false;
    this._is_enabled = true;
}
```

El método ***onStart()*** define la estructura HTML del widget, sin realizar ninguna manipulación de datos. Este método se invoca antes de la primera activación de la página del panel, es decir, antes de que el panel y sus widgets se muestren completamente al usuario.

Ejemplo:

```
onStart() {
    this._events.resize = () => {
        const padding = 25;
        const header_height = this._view_mode === ZBX_WIDGET_VIEW_MODE_HIDDEN_HEADER
            ? 0
            : this._header.offsetHeight;

        this._target.style.setProperty(
            '--content-height',
            `${this._cell_height * this._pos.height - padding * 2 - header_height}px`
        );
```

```
    }
}
```

El método ***onActivate()*** hace que el widget esté activo e interactivo al habilitar los listeners de eventos personalizados (para responder a acciones del usuario) e iniciar el ciclo de actualización del widget (para mantener su contenido actualizado). Este método se invoca cuando la página del panel se activa, es decir, cuando se muestra completamente en la interfaz de usuario.

Tenga en cuenta que antes de que se invoque el método *onActivate()*, el widget está en estado inactivo (`WIDGET_STATE_INACTIVE`). Después de una invocación exitosa, el widget pasa al estado activo (`WIDGET_STATE_ACTIVE`). En el estado activo, el widget es receptivo, escucha eventos, actualiza su contenido periódicamente y puede interactuar con otros widgets.

Ejemplo:

```
onActivate() {
    this._startClock();

    this._resize_observer = new ResizeObserver(this._events.resize);
    this._resize_observer.observe(this._target);
}
```

El método ***onDeactivate()*** detiene cualquier actividad e interactividad del widget desactivando los listeners de eventos personalizados y deteniendo el ciclo de actualización del widget. Este método se invoca cuando la página del panel se desactiva, es decir, se cambia o elimina, o cuando el widget se elimina de la página del panel.

Tenga en cuenta que antes de que se invoque el método *onDeactivate()*, el widget está en el estado activo (`WIDGET_STATE_ACTIVE`). Después de una invocación exitosa, el widget pasa al estado inactivo (`WIDGET_STATE_INACTIVE`).

Ejemplo:

```
onDeactivate() {
    this._stopClock();
    this._resize_observer.disconnect();
}
```

El método ***onDestroy()*** realiza tareas de limpieza antes de que el widget sea eliminado del panel, lo que puede incluir cerrar una conexión a la base de datos que se estableció durante la inicialización del widget, limpiar datos temporales para liberar memoria del sistema y evitar fugas de recursos, cancelar el registro de escuchas de eventos relacionados con eventos de cambio de tamaño o clics de botones para evitar el manejo innecesario de eventos y fugas de memoria, etc. Este método se invoca cuando se elimina el widget o la página del panel que lo contiene.

Tenga en cuenta que antes de que se invoque el método *onDestroy()*, un widget en estado activo (`WIDGET_STATE_ACTIVE`) siempre se desactiva con la invocación del método *onDeactivate()*.

Ejemplo:

```
onDestroy() {
    if (this._filter_widget) {
        this._filter_widget.off(CWidgetMap.WIDGET_NAVTREE_EVENT_MARK, this._events.mark);
        this._filter_widget.off(CWidgetMap.WIDGET_NAVTREE_EVENT_SELECT, this._events.select);
    }
}
```

El método ***onEdit()*** define la apariencia y el comportamiento del widget cuando el dashboard pasa al modo de edición. Este método se invoca cuando el dashboard pasa al modo de edición, normalmente cuando un usuario interactúa con el botón *Editar* del widget o el botón *Editar dashboard* del dashboard.

Ejemplo:

```
onEdit() {
    this._deactivateGraph();
}
```

Métodos del proceso de actualización

Los métodos del proceso de actualización del widget son responsables de recuperar datos actualizados desde el servidor Zabbix o cualquier otra fuente de datos y mostrarlos en el widget.

El método ***promiseUpdate()*** inicia el proceso de actualización de datos recuperando datos, normalmente utilizando solicitudes web o llamadas a API. Este método se invoca cuando se muestra una página de panel y periódicamente después, hasta que la página de panel se cambia a otra página de panel.

A continuación se muestra un ejemplo de la implementación por defecto del método *promiseUpdate()* utilizado por la mayoría de los widgets nativos de Zabbix. En la implementación por defecto, el método *promiseUpdate()* sigue un patrón general para recuperar datos del servidor. Crea un nuevo objeto `Curl` con la URL y los parámetros de solicitud apropiados, envía una solicitud POST utilizando el método *fetch()* con el objeto de datos construido por el método *getUpdateRequestData()*, y procesa la respuesta (o una respuesta de error) con *processUpdateResponse(response)* o *processUpdateErrorResponse(error)* según corresponda. Esta implementación es adecuada para la mayoría de los widgets, ya que normalmente recuperan datos en formato JSON y los gestionan de manera coherente.

```
promiseUpdate() {
    const curl = new Curl('zabbix.php');

    curl.setArgument('action', `widget.${this._type}.view`);

    return fetch(curl.getUrl(), {
        method: 'POST',
        headers: {'Content-Type': 'application/json'},
        body: JSON.stringify(this.getUpdateRequestData()),
        signal: this._update_abort_controller.signal
    })
        .then((response) => response.json())
        .then((response) => {
            if ('error' in response) {
                this.processUpdateErrorResponse(response.error);

                return;
            }

            this.processUpdateResponse(response);
        });
    }
```

El método ***getUpdateRequestData()*** prepara los datos de la solicitud al servidor para actualizar el widget recopilando varias propiedades y sus valores correspondientes (identificadores de widget, configuraciones de filtro, rangos de tiempo, etc.) del estado y la configuración del widget, y construyendo un objeto de datos que representa la información necesaria que se enviará al servidor en la solicitud de actualización. Este método se invoca solo como parte del método predeterminado *promiseUpdate()*, es decir, durante el proceso de actualización del widget.

Implementación por defecto:

```
getUpdateRequestData() {
    return {
        templateid: this._dashboard.templateid ?? undefined,
        dashboardid: this._dashboard.dashboardid ?? undefined,
        widgetid: this._widgetid ?? undefined,
        name: this._name !== '' ? this._name : undefined,
        fields: Object.keys(this._fields).length > 0 ? this._fields : undefined,
        view_mode: this._view_mode,
        edit_mode: this._is_edit_mode ? 1 : 0,
        dynamic_hostid: this._dashboard.templateid !== null || this.supportsDynamicHosts()
            ? (this._dynamic_hostid ?? undefined)
            : undefined,
        ...this._contents_size
    };
}
```

El método ***processUpdateResponse(response)*** gestiona la respuesta recibida del servidor tras la solicitud de actualización y, si el proceso de actualización ha sido exitoso y sin errores, limpia los datos del widget y muestra los nuevos contenidos con el método *setContents()*. Este método se invoca únicamente como parte del método por defecto *promiseUpdate()*, es decir, durante el proceso de actualización del widget.

Implementación por defecto:

```
processUpdateResponse(response) {
    this._setHeaderName(response.name);

    this._updateMessages(response.messages);
```

```
    this._updateInfo(response.info);
    this._updateDebug(response.debug);

    this.setContents(response);
}
```

El método ***processUpdateErrorResponse(error)*** gestiona la respuesta recibida del servidor después de la solicitud de actualización si la respuesta es un error, y muestra el/los mensaje(s) de error. Este método se invoca solo como parte del método por defecto *promiseUpdate()*, es decir, durante el proceso de actualización del widget.

Implementación por defecto:

```
processUpdateErrorResponse(error) {
    this._updateMessages(error.messages, error.title);
}
```

El método ***setContents(response)*** muestra el contenido del widget si el proceso de actualización del widget se ha realizado correctamente y sin errores, lo que puede incluir la manipulación de elementos DOM, la actualización de componentes de la interfaz de usuario, la aplicación de estilos o formatos, etc. Este método se invoca solo como parte del método predeterminado *processUpdateResponse(response)*, es decir, durante el proceso de manejo de la respuesta recibida del servidor después de la solicitud de actualización.

Implementación por defecto:

```
setContents(response) {
    this._body.innerHTML = response.body ?? '';
}
```

Métodos de modificación de presentación

Los métodos de modificación de presentación del widget son responsables de modificar la apariencia del widget.

El método ***onResize()*** es responsable de ajustar los elementos visuales del widget para adaptarse al nuevo tamaño del widget, lo que puede incluir reorganizar elementos, ajustar dimensiones de los elementos, truncar texto, implementar carga diferida para mejorar la capacidad de respuesta durante el cambio de tamaño, etc. Este método se invoca cuando se cambia el tamaño del widget, por ejemplo, cuando el usuario cambia manualmente el tamaño del widget o cuando se cambia el tamaño de la ventana del navegador.

Ejemplo:

```
onResize() {
    if (this.getState() === WIDGET_STATE_ACTIVE) {
        this._startUpdating();
    }
}
```

El método ***hasPadding()*** es responsable de aplicar un padding vertical de 8px en la parte inferior del widget cuando está configurado para mostrar su encabezado. Este método se invoca cuando la página del dashboard se activa, es decir, cuando se convierte en la página mostrada en la interfaz de usuario.

Implementación por defecto:

```
hasPadding() {
    return this.getViewMode() !== ZBX_WIDGET_VIEW_MODE_HIDDEN_HEADER;
}
```

Para algunos widgets es necesario utilizar todo el espacio disponible del widget para configurar, por ejemplo, un color de fondo personalizado. A continuación se muestra un ejemplo de la implementación del método *hasPadding()* utilizado en el widget nativo de Zabbix *Valor de la métrica*.

```
hasPadding() {
    return false;
}
```

Tutoriales

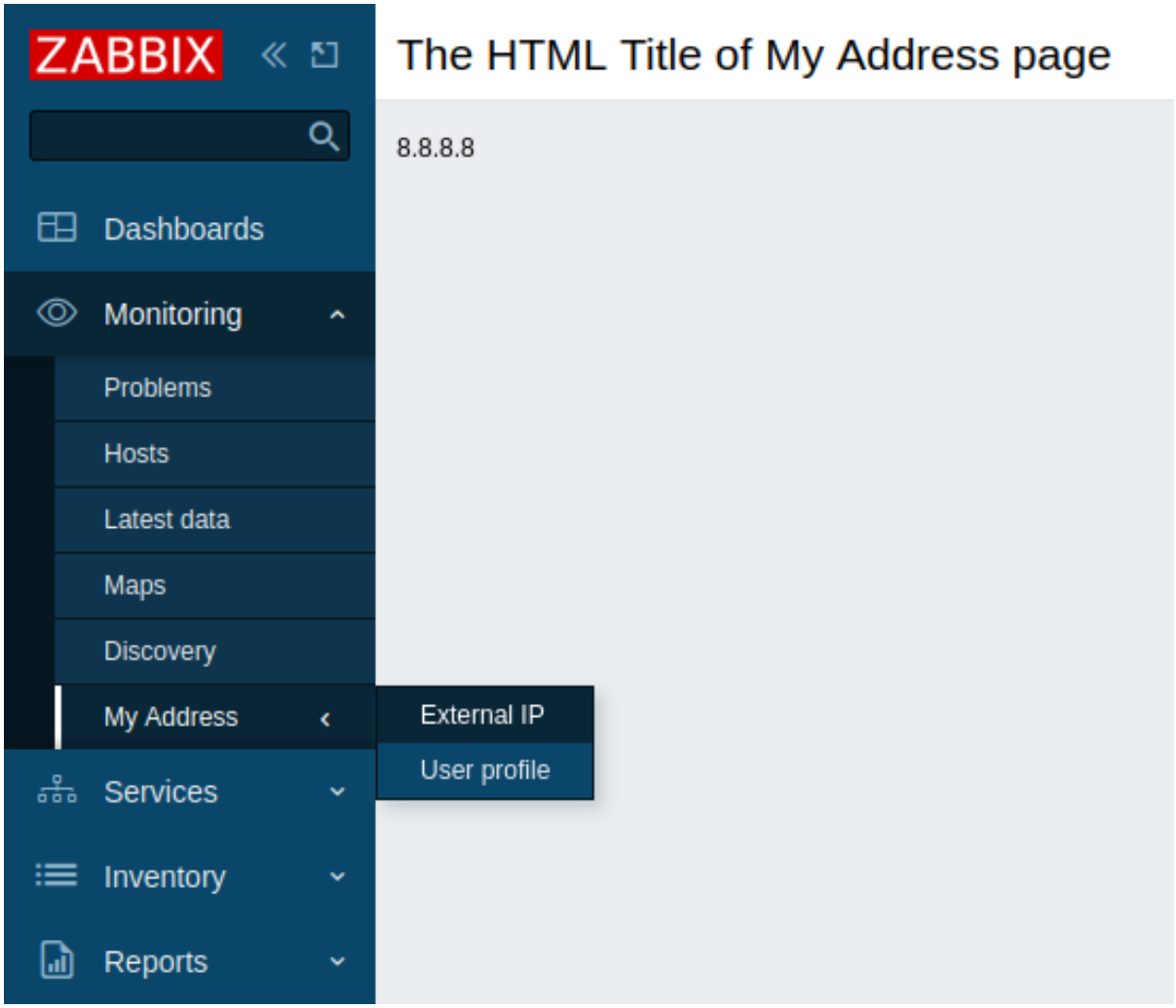
Esta sección contiene tutoriales prácticos paso a paso para ilustrar cómo construir un módulo y un widget personalizados en Zabbix.

Crear un módulo (tutorial)

Este es un tutorial paso a paso que muestra cómo crear un módulo sencillo para la interfaz web de Zabbix. Puede descargar todos los archivos de este módulo como un archivo ZIP: MyAddress.zip.

Lo que crearás

Durante este tutorial, primero crearás un módulo de frontend que añade una nueva sección de menú *My Address* y luego lo convertirás en un módulo de frontend más avanzado que realiza una solicitud HTTP a *https://api.seeip.org* y muestra la respuesta —la dirección IP de tu equipo— en una nueva página dentro de la sección de menú *My Address* recién creada. Así es como se verá el módulo terminado:



Parte I - Nueva sección de menú

Agregar un módulo vacío al frontend de Zabbix

1. Cree un directorio *MyAddress* en el directorio *modules* de su instalación del frontend de Zabbix (por ejemplo, *zabbix/ui/modules*).
2. Cree un archivo *manifest.json* con los metadatos básicos del módulo (consulte la descripción de los parámetros admitidos).

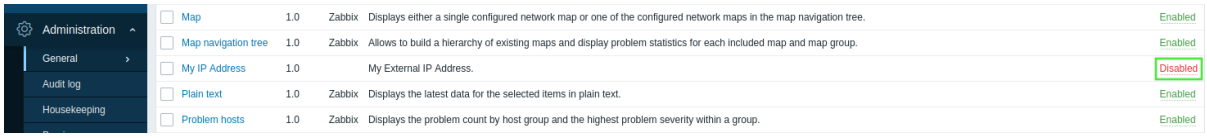
ui/modules/MyAddress/manifest.json

```
{
    "manifest_version": 2.0,
    "id": "my-address",
    "name": "My IP Address",
    "version": "1.0",
    "namespace": "MyAddress",
    "description": "My External IP Address."
}
```

3. En el frontend de Zabbix, vaya a la sección *Administration* > *General* > *Modules* y haga clic en el botón *Scan directory*.



4. Localice el nuevo módulo *My IP Address* en la lista y haga clic en el hipervínculo "Disabled" para cambiar el estado del módulo de "Disabled" a "Enabled" (si el módulo no aparece en la lista, consulte la sección de troubleshooting).



El módulo ya está registrado en el frontend. Sin embargo, todavía no es visible, porque aún debe definir la funcionalidad del módulo. Una vez que agregue contenido al directorio del módulo, verá inmediatamente los cambios en el frontend de Zabbix al actualizar la página.

Crear una sección de menú

1. Cree un archivo *Module.php* en el directorio *MyAddress*.

Este archivo implementa una nueva clase *Module* que extiende la clase predeterminada *CModule*. La clase *Module* insertará una nueva sección de menú *My Address* en el menú principal.

El método *setAction()* especifica una acción que se ejecutará al hacer clic en la sección de menú. Para empezar, puede usar la acción predefinida *userprofile.edit*, que abrirá la página *User profile*. En la parte III de este tutorial, aprenderá a crear una acción personalizada.

ui/modules/MyAddress/Module.php

```
<?php

namespace Modules\MyAddress;

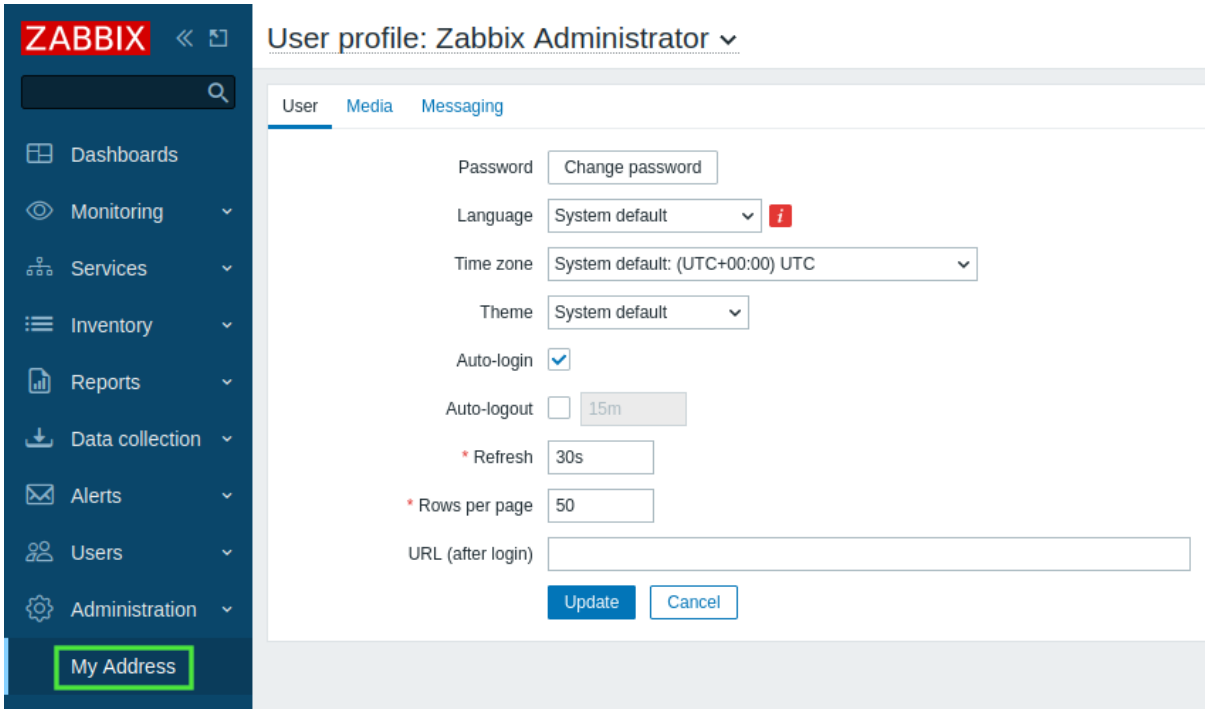
use Zabbix\Core\CModule,
    APP,
    CMenuItem;

class Module extends CModule {

    public function init(): void {
        APP::Component()->get('menu.main')
            ->add((new CMenuItem(_('My Address')))
            ->setAction('userprofile.edit'));
    }
}
```

Note:
Puede reemplazar `'userprofile.edit'` por otras acciones, por ejemplo, `'charts.view'` (abre gráficos personalizados), `'problems.view'` (abre *Monitoring* > *Problems*), o `'report.status'` (abre el informe *System information*).

3. Actualice el frontend de Zabbix. Ahora hay una nueva sección *My Address* en la parte inferior del menú principal de Zabbix. Haga clic en *My Address* para abrir la página *User profile*.



Parte II - Cambio de ubicación de la sección del menú

En esta parte, moverá la sección del menú *My Address* a la sección *Monitoring* y luego le agregará un menú anidado. Como resultado, los usuarios podrán acceder a dos páginas de submenú desde la sección de menú *Monitoring* > *My Address*.

1. Abra y edite el archivo *Module.php*.

ui/modules/MyAddress/Module.php

```
<?php

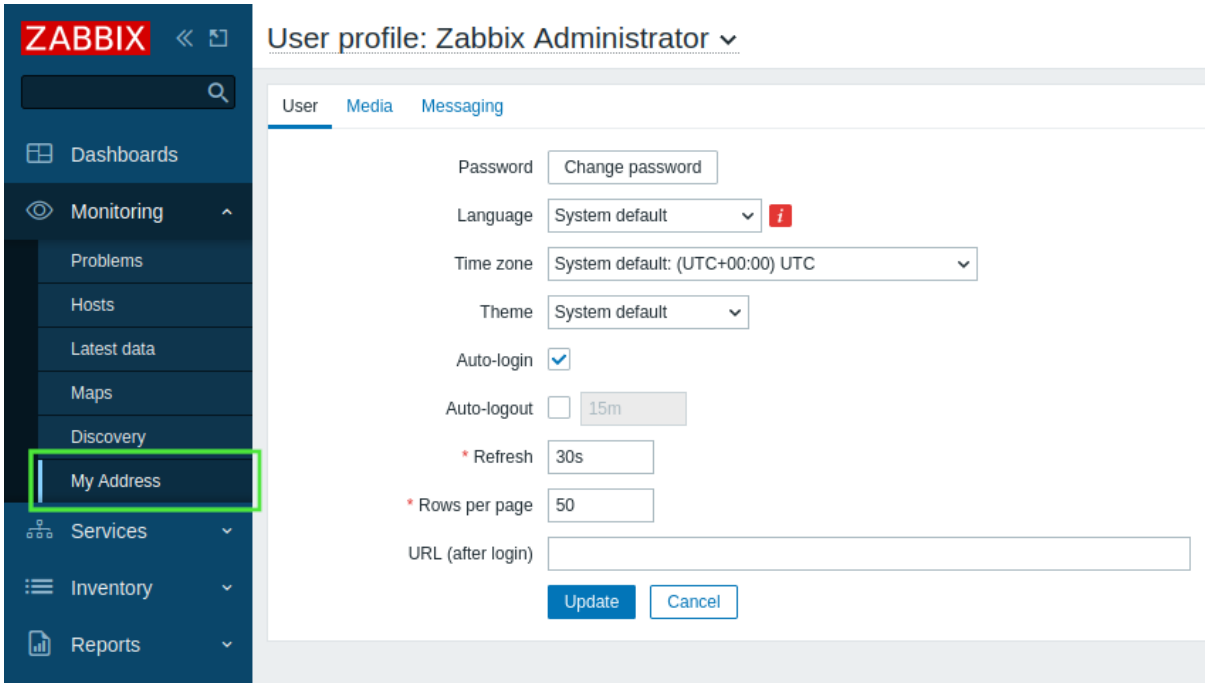
namespace Modules\MyAddress;

use Zabbix\Core\CModule,
    APP,
    CMenuItem;

class Module extends CModule {

    public function init(): void {
        APP::Component()->get('menu.main')
            ->findOrAdd(_('Monitoring'))
            ->getSubmenu()
            ->insertAfter(_('Discovery'),
                (new CMenuItem(_('My Address')))->setAction('userprofile.edit')
            );
    }
}
```

2. Actualice el frontend de Zabbix. Expanda la sección del menú *Monitoring* y observe que la sección *My address* ahora se encuentra debajo de la sección *Discovery*.



3. Para agregar páginas anidadas a la sección del menú *My Address*, abra y edite nuevamente el archivo *Module.php*.

Este paso crea dos subsecciones:

- *External IP*, que ejecuta una nueva acción 'my.address' que se definirá en los siguientes pasos;
- *User profile*, que ejecuta la acción predefinida 'userprofile.edit' para abrir la página *User profile*.

Tenga en cuenta que, para el menú anidado, debe usar la clase *CMenu* además de las clases utilizadas en los pasos anteriores.

ui/modules/MyAddress/Module.php

```
<?php

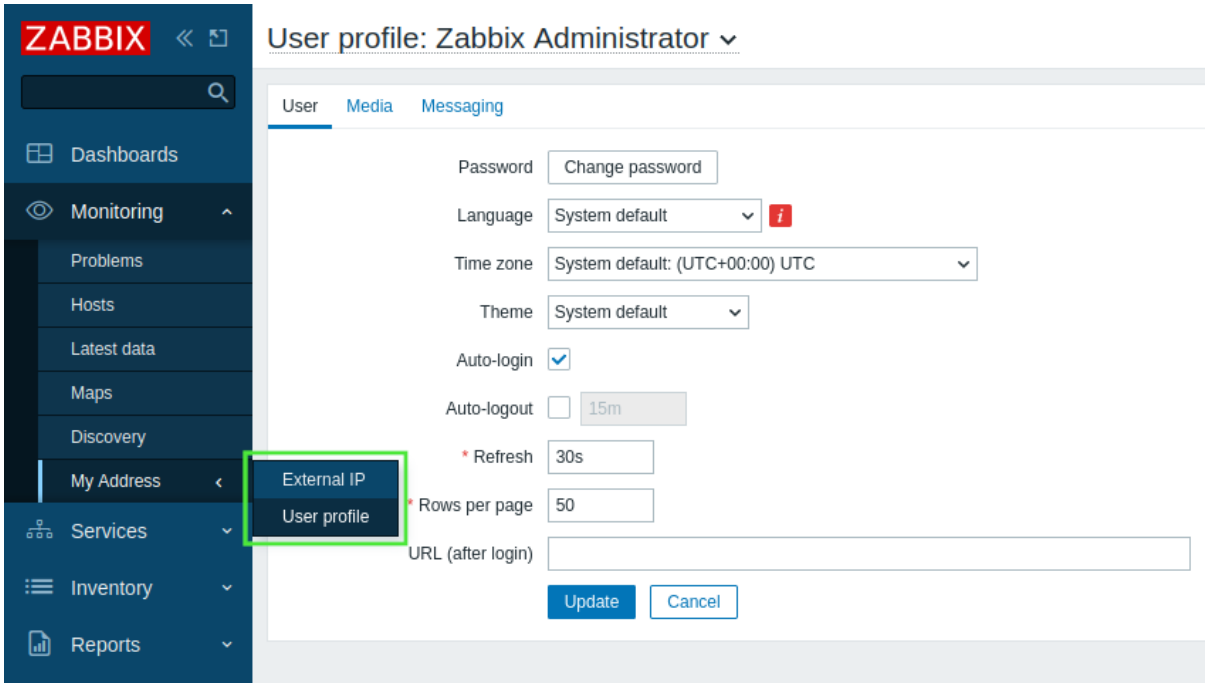
namespace Modules\MyAddress;

use Zabbix\Core\CModule,
    APP,
    CMenu,
    CMenuItem;

class Module extends CModule {

    public function init(): void {
        APP::Component()->get('menu.main')
            ->findOrAdd(_('Monitoring'))
            ->getSubmenu()
            ->insertAfter(_('Discovery'),
                (new CMenuItem(_('My Address')))->setSubMenu(
                    new CMenu([
                        (new CMenuItem(_('External IP')))->setAction('my.address'),
                        (new CMenuItem(_('User profile')))->setAction('userprofile.edit')
                    ])
                )
            );
    }
}
```

4. Actualice el frontend de Zabbix. Observe que la sección del menú *My address* ahora contiene un menú de tercer nivel con dos páginas: *External IP* y *User profile*.



Parte III - Acción del módulo

Una acción se implementa en dos archivos: *actions/MyAddress.php* y *views/my.address.php*. El archivo ***actions/MyAddress.php*** se encarga de la implementación de la lógica de negocio, mientras que el archivo ***views/my.address.php*** es responsable de la vista.

1. Cree un directorio *actions* en el directorio *MyAddress*.
2. Cree un archivo *MyAddress.php* en el directorio *actions*.

La lógica de la acción se definirá en la clase *MyAddress*. Esta clase de acción implementará cuatro funciones: *init()*, *checkInput()*, *checkPermissions()* y *doAction()*. Zabbix frontend llama a la función *doAction()* cuando se solicita la acción. Esta función es responsable de la lógica de negocio del módulo.

Attention:
Los datos deben organizarse como un array asociativo. El array puede ser multidimensional y puede contener cualquier dato esperado por la vista.

ui/modules/MyAddress/actions/MyAddress.php

```
<?php

namespace Modules\MyAddress\Actions;

use CController,
    CControllerResponseData;

class MyAddress extends CController {

    public function init(): void {
        $this->disableCsrfValidation();
    }

    protected function checkInput(): bool {
        return true;
    }

    protected function checkPermissions(): bool {
        return true;
    }

    protected function doAction(): void {
        $data = ['my-ip' => file_get_contents("https://api.seeip.org")];
```

```
        $response = new CControllerResponseData($data);
        $this->setResponse($response);
    }
}
```

3. Cree un nuevo directorio *views* en el directorio *MyAddress*.
4. Cree un archivo *my.address.php* en el directorio *views* y defina la vista del módulo.

Tenga en cuenta que la variable `$data` está disponible en la vista sin necesidad de definirla específicamente. El framework pasa automáticamente el array asociativo a la vista.

ui/modules/MyAddress/views/my.address.php

```
<?php

(new CHtmlPage())
    ->setTitle(_('The HTML Title of My Address Page'))
    ->addItem(new CDiv($data['my-ip']))
    ->show();
```

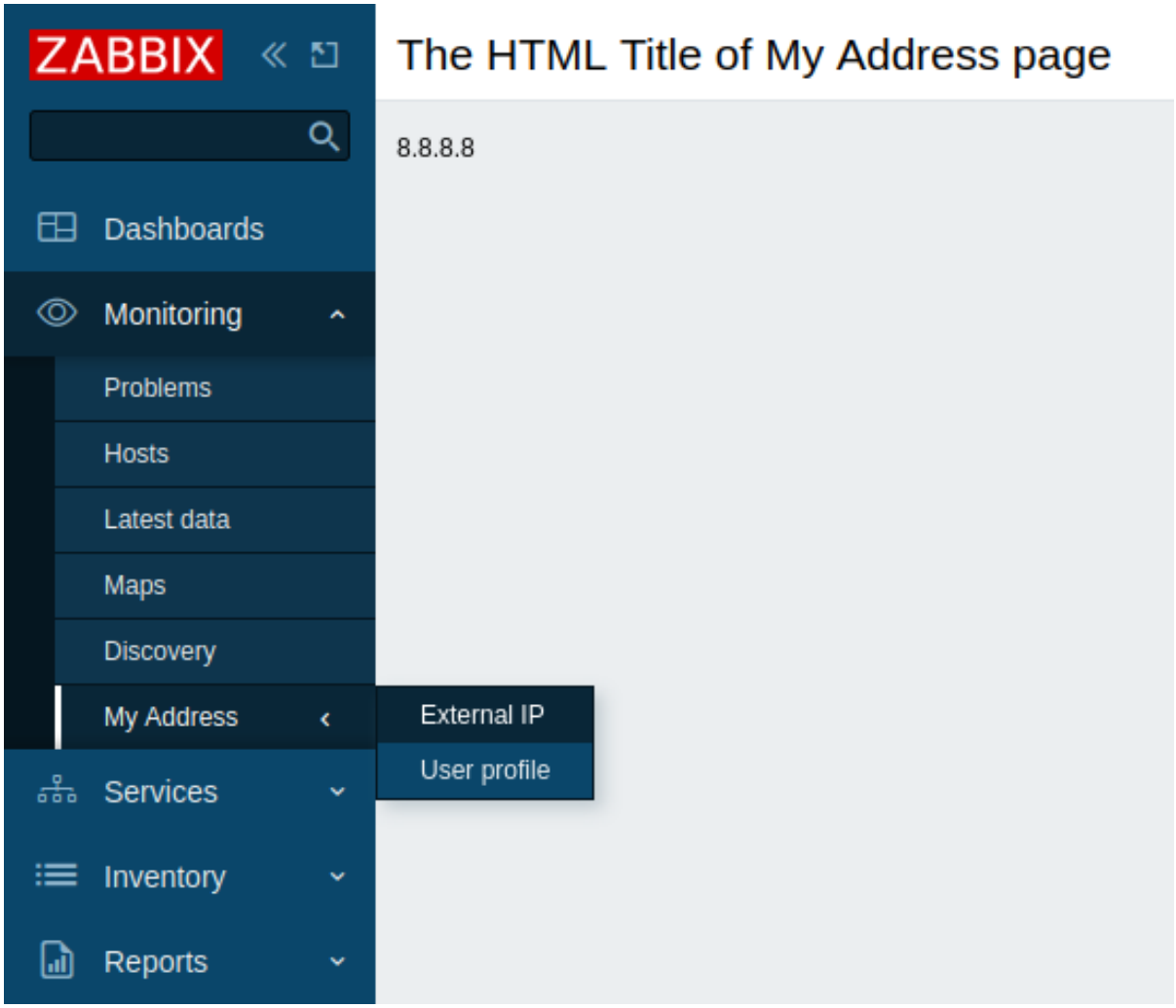
5. La acción del módulo debe registrarse en el archivo *manifest.json*. Abra *manifest.json* y agregue un nuevo objeto `actions` que contenga:

- la clave de la acción con el nombre de la acción escrito en minúsculas (a-z) y con las palabras separadas por puntos (por ejemplo, `my.address`);
- el nombre de la clase de la acción (`MyAddress`) como valor de la clave `class` del objeto `my.address`;
- el nombre de la vista de la acción (`my.address`) como valor de la clave `view` del objeto `my.address`.

ui/modules/MyAddress/manifest.json

```
{
    "manifest_version": 2.0,
    "id": "my-address",
    "name": "My IP Address",
    "version": "1.0",
    "namespace": "MyAddress",
    "description": "My External IP Address.",
    "actions": {
        "my.address": {
            "class": "MyAddress",
            "view": "my.address"
        }
    }
}
```

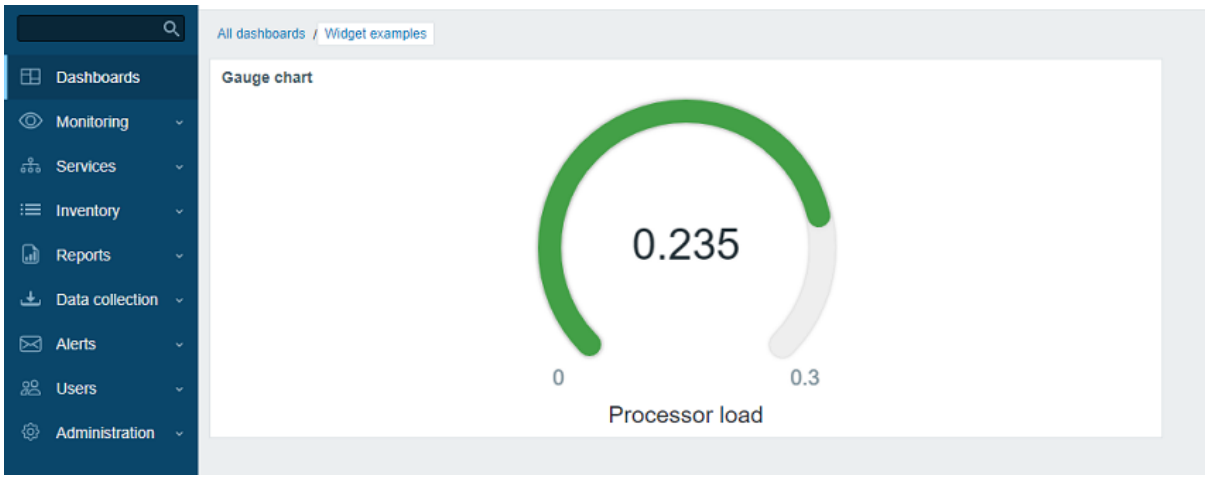
6. Actualice Zabbix frontend. Haga clic en *My address* > *External IP* para ver la dirección IP de su computadora.



Crear un widget (tutorial)

Este es un tutorial paso a paso que muestra cómo crear un widget de panel simple. Puede descargar todos los archivos de este widget como un archivo ZIP: lesson_gauge_chart.zip.

Lo que crearás Durante este tutorial, primero crearás un widget básico de "¡Hola, mundo!" y luego lo convertirás en un widget más avanzado que muestra el valor de un item como un gráfico de indicador. Así es como se verá el widget terminado:



Parte I - "¡Hola, mundo!" En esta sección aprenderá cómo crear los elementos mínimos requeridos de un widget y añadir un nuevo widget al frontend de Zabbix.

Añadir un widget en blanco al frontend de Zabbix

1. Cree un directorio `lesson_gauge_chart` en el directorio `modules` de la instalación de su frontend de Zabbix (por ejemplo, *zabbix/ui/modules*).

Note:
Todos los widgets personalizados se tratan como módulos externos y deben añadirse al directorio `modules` de la instalación de su frontend de Zabbix (por ejemplo, *zabbix/ui/modules*). El directorio `zabbix/ui/widgets` está reservado para los widgets integrados de Zabbix y se actualiza junto con la interfaz de usuario de Zabbix.

2. Cree un archivo `manifest.json` con los metadatos básicos del widget (consulte la descripción de los parámetros admitidos).

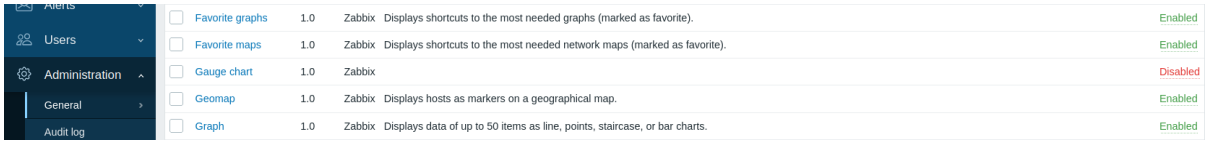
ui/modules/lesson_gauge_chart/manifest.json

```
{
    "manifest_version": 2.0,
    "id": "lesson_gauge_chart",
    "type": "widget",
    "name": "Gauge chart",
    "namespace": "LessonGaugeChart",
    "version": "1.1",
    "author": "Zabbix"
}
```

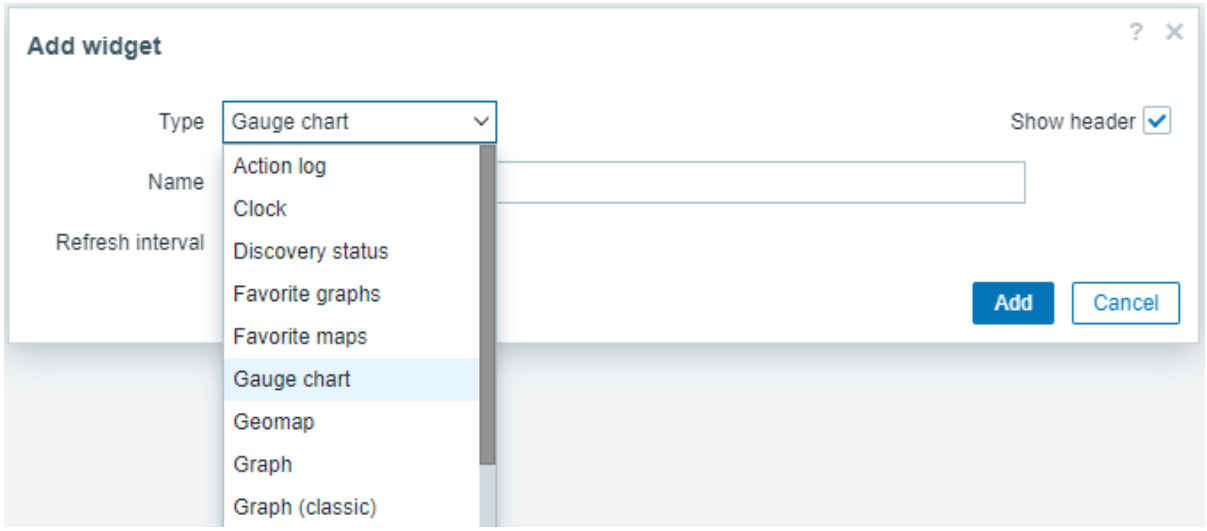
3. En el frontend de Zabbix, vaya a la sección *Administration* > *General* > *Modules* y haga clic en el botón *Scan directory*.



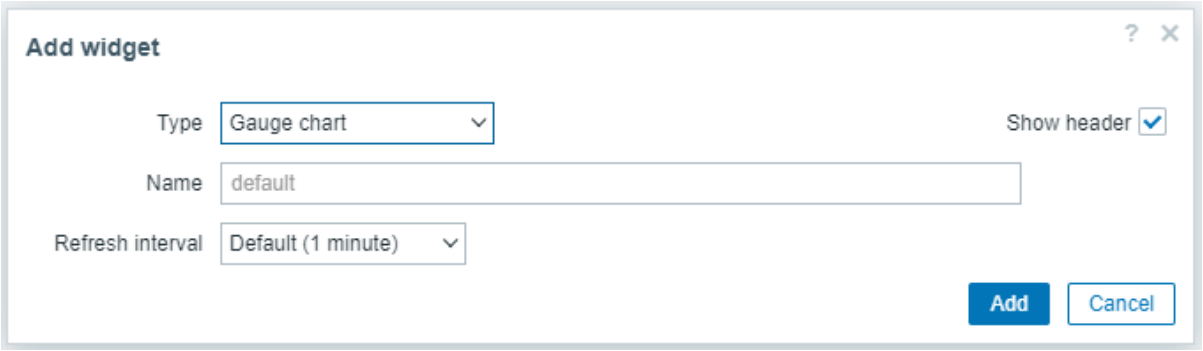
4. Localice el nuevo módulo *Gauge chart* en la lista y haga clic en el hipervínculo "Disabled" para cambiar el estado del módulo de "Disabled" a "Enabled" (si el módulo no aparece en la lista, consulte la sección de solución de problemas).



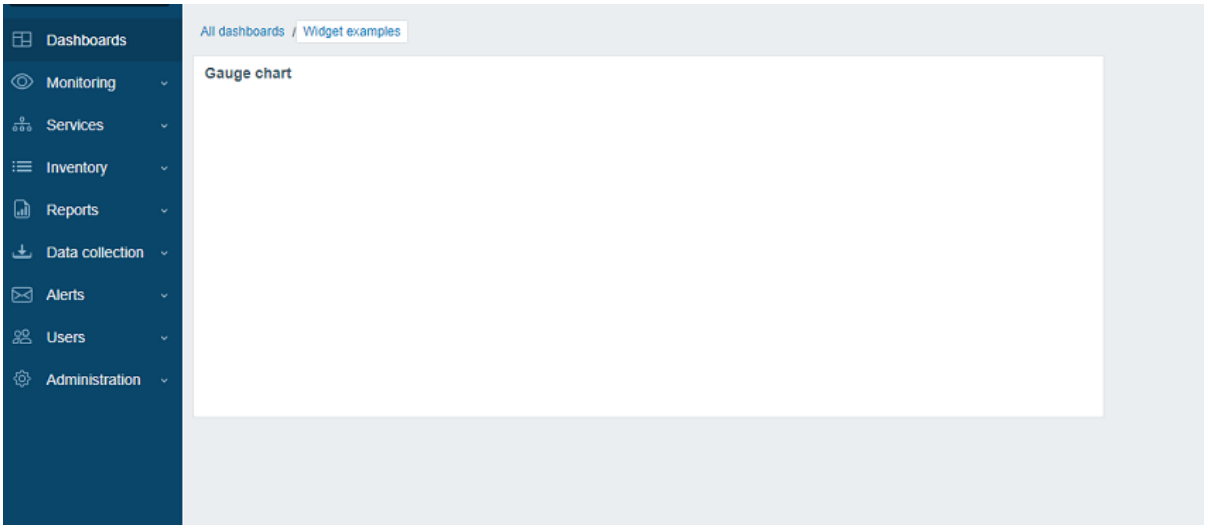
5. Abra un dashboard, cámbielo al modo de edición y añada un nuevo widget. En el campo *Type*, seleccione "Gauge chart".



6. En este punto, la configuración del widget *Gauge chart* contiene solo los campos comunes del widget *Name* y *Refresh interval*. Haga clic en *Add* para añadir el widget al dashboard.



7. Debería aparecer un widget en blanco en el dashboard. Haga clic en *Save changes* en la esquina superior derecha para guardar el dashboard.



Añadir una vista de widget

Note:
El archivo **view** del widget debe estar ubicado en el directorio `views` (para este tutorial, `ui/modules/lesson_gauge_chart/views/`). Si el archivo tiene el nombre predeterminado `widget.view.php`, no es necesario registrarlo en el archivo `manifest.json`. Si el archivo tiene un nombre diferente, especifíquelo en la sección `actions/widget.lesson_gauge_chart.view` del archivo `manifest.json`.

1. Cree un directorio `views` en el directorio `lesson_gauge_chart`.
2. Cree un archivo `widget.view.php` en el directorio `views`.

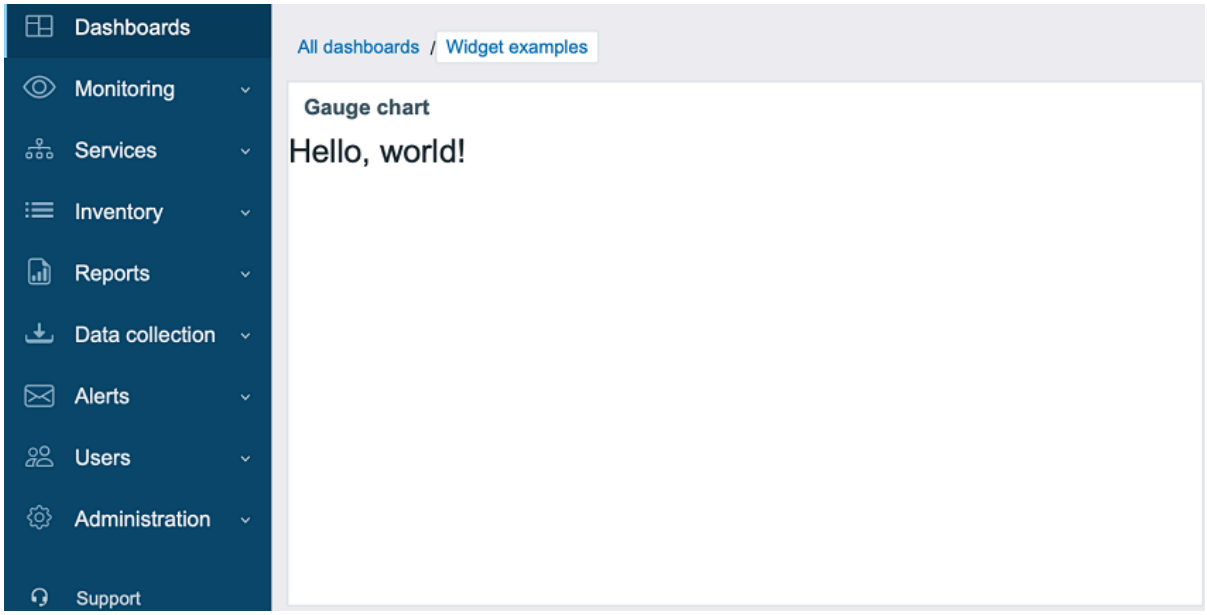
`ui/modules/lesson_gauge_chart/views/widget.view.php`

```
<?php

/**
 * Vista del widget de gráfico de medidor.
 *
 * @var CView $this
 * @var array $data
 */

(new CWidgetView($data))
    ->addItem(
        new CTag('h1', true, 'Hello, world!')
    )
    ->show();
```

3. Actualice el dashboard. El widget *Gauge chart* ahora muestra "Hello, world!".



Parte II - Gráfico de indicador Añadir ajustes a una vista de configuración y usarlos en una vista de widget

En esta sección, aprenderá cómo añadir un campo de configuración de widget y mostrar el valor introducido en la vista del widget como texto.

La configuración del widget consta de un formulario (`Zabbix\Widgets\CWidgetForm`) y una vista del formulario del widget (`widget.edit.php`). Para añadir campos (`Zabbix\Widgets\CWidgetField`), debe crear una clase `WidgetForm`, que extenderá `Zabbix\Widgets\CWidgetForm`.

El formulario contiene el conjunto de campos (`Zabbix\Widgets\CWidgetField`) de varios tipos, que se utilizan para validar los valores introducidos por el usuario. El campo del formulario (`Zabbix\Widgets\CWidgetField`) para cada tipo de elemento de entrada convierte el valor a un formato único para almacenarlo en la base de datos.

> **Note:**
> El archivo de **formulario** del widget debe estar ubicado en el directorio `includes` (para este tutorial, `ui/modules/lesson_gauge_chart/includes/`). Si el archivo tiene el nombre predeterminado `WidgetForm.php`, no necesita registrarlo en el archivo `manifest.json`. Si el archivo tiene un nombre diferente, especifíquelo en la sección `widget/form_class` del archivo `manifest.json`.

1. Cree un nuevo directorio `includes` en el directorio `lesson_gauge_chart`.
2. Cree un archivo `WidgetForm.php` en el directorio `includes`.

`ui/modules/lesson_gauge_chart/includes/WidgetForm.php`

```
<?php

namespace Modules\LessonGaugeChart\Includes;

use Zabbix\Widgets\CWidgetForm;

class WidgetForm extends CWidgetForm {
}
```

3. Añada un campo *Description* al formulario de configuración del widget. Se trata de un campo de texto normal, donde el usuario puede introducir cualquier conjunto de caracteres. Puede usar la clase `CWidgetFieldTextBox` para ello.

`ui/modules/lesson_gauge_chart/includes/WidgetForm.php`

```
<?php

namespace Modules\LessonGaugeChart\Includes;

use Zabbix\Widgets\CWidgetForm;

use Zabbix\Widgets\Fields\CWidgetFieldTextBox;
```

```
class WidgetForm extends CWidgetForm {

    public function addFields(): self {
        return $this
            ->addField(
               new CWidgetFieldTextBox('description', _('Description'))
            );
    }
}
```

4. En el directorio `views`, cree un archivo de vista de configuración del widget `widget.edit.php` y añada una vista para el nuevo campo *Description*. Para la clase de campo `CWidgetFieldTextBox`, la vista es `CWidgetFieldTextBoxView`.

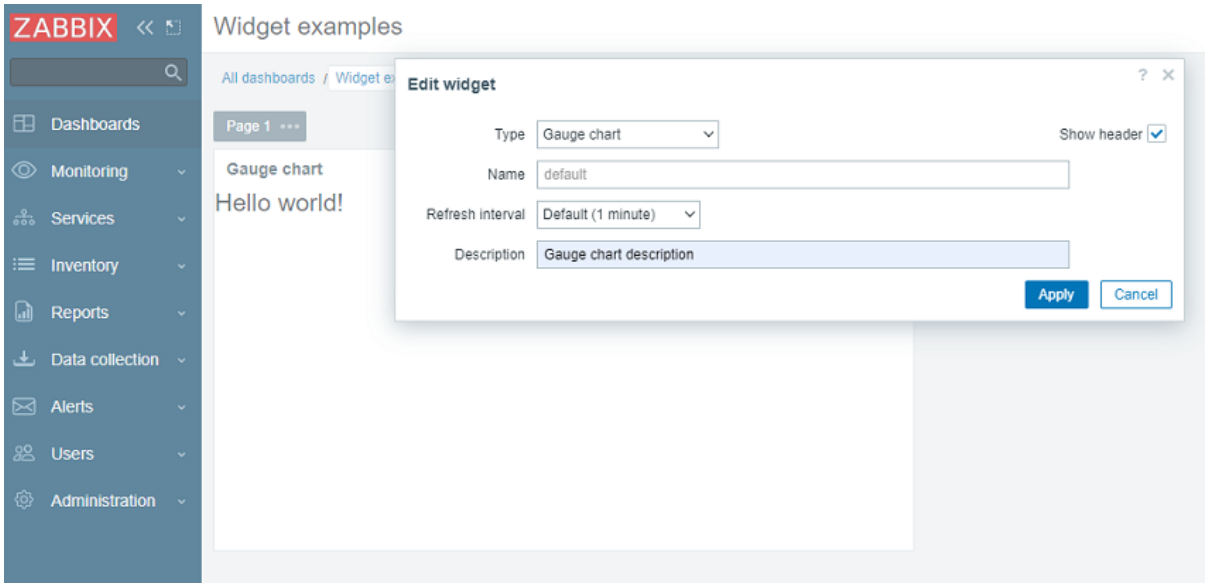
`ui/modules/lesson_gauge_chart/views/widget.edit.php`

```
<?php

/**
 * Gauge chart widget form view.
 *
 * @var CView $this
 * @var array $data
 */

(new CWidgetFormView($data))
    ->addField(
        new CWidgetFieldTextBoxView($data['fields']['description'])
    )
    ->show();
```

5. Vaya al dashboard y haga clic en el icono de engranaje del widget para abrir el formulario de configuración del widget.
6. El formulario de configuración del widget ahora contiene un nuevo campo de texto *Description*. Introduzca cualquier valor, por ejemplo, *Gauge chart description*.



7. Haga clic en *Apply* en el formulario de configuración del widget. Luego haga clic en *Save changes* en la esquina superior derecha para guardar el dashboard. Tenga en cuenta que la nueva descripción no es visible en ninguna parte, y el widget sigue mostrando "Hello, world!".

Para que la nueva descripción aparezca en el widget, es necesario recuperar el valor del campo *Description* de la base de datos y pasarlo a la vista del widget. Para ello, debe crear una clase de acción.

8. Cree un nuevo directorio `actions` en el directorio `lesson_gauge_chart`.
9. Cree un archivo `WidgetView.php` en el directorio `actions`. La clase de acción `WidgetView` extenderá la clase `CControllerDashboardWidgetView`.

Los valores de los campos de configuración del widget se almacenan en la propiedad `$fields_values` de la clase de acción.

ui/modules/lesson_gauge_chart/actions/WidgetView.php

```
<?php

namespace Modules\LessonGaugeChart\Actions;

use CControllerDashboardWidgetView,
    CControllerResponseData;

class WidgetView extends CControllerDashboardWidgetView {

    protected function doAction(): void {
        $this->setResponse(new CControllerResponseData([
            'name' => $this->getInput('name', $this->widget->getName()),
            'description' => $this->fields_values['description'],
            'user' => [
                'debug_mode' => $this->getDebugMode()
            ]
        ]));
    }
}
```

10. Abra `manifest.json` y registre `WidgetView` como clase de acción en la sección `actions/widget.lesson_gauge_chart.view`.

ui/modules/lesson_gauge_chart/manifest.json

```
{
    "manifest_version": 2.0,
    "id": "lesson_gauge_chart",
    "type": "widget",
    "name": "Gauge chart",
    "namespace": "LessonGaugeChart",
    "version": "1.0",
    "author": "Zabbix",
    "actions": {
        "widget.lesson_gauge_chart.view": {
            "class": "WidgetView"
        }
    }
}
```

11. Ahora puede usar el valor del campo de descripción, contenido en `$data['description']`, en la vista del widget. Abra `views/widget.view.php` y reemplace el texto estático "Hello, world!" por `$data['description']`.

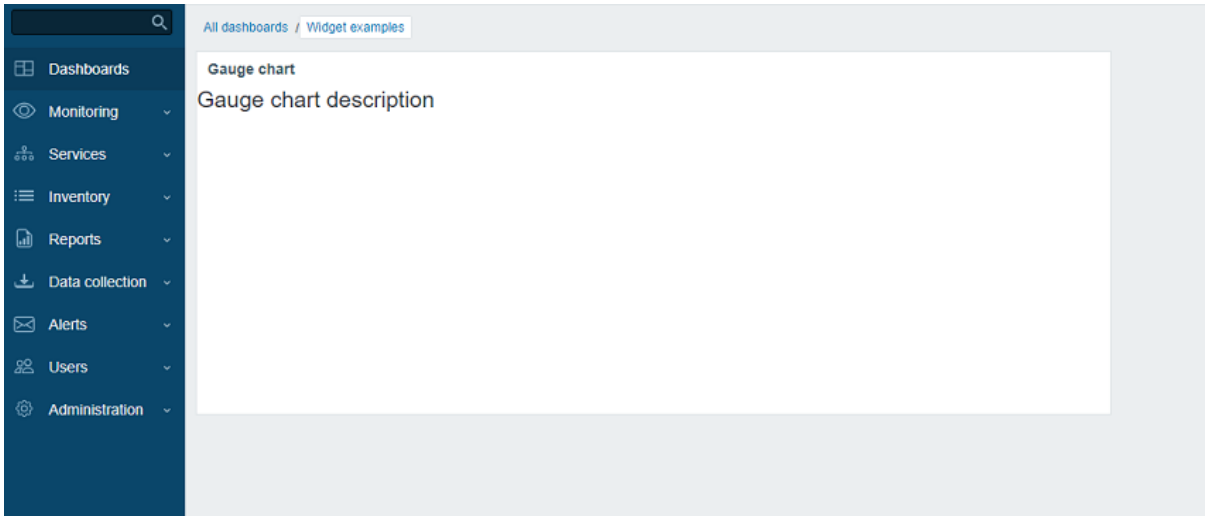
ui/modules/lesson_gauge_chart/views/widget.view.php

```
<?php

/**
 * Gauge chart widget view.
 *
 * @var CView $this
 * @var array $data
 */

(new CWidgetView($data))
    ->addItem(
        new CTag('h1', true, $data['description'])
    )
    ->show();
```

12. Actualice la página del dashboard. Ahora debería ver el texto de la descripción del widget en lugar de "Hello, world!".



Recuperar el valor de un item mediante la API

El widget debe mostrar el último valor de un item elegido por el usuario. Para ello, debe agregar la capacidad de seleccionar items en la configuración del widget.

En esta sección, aprenderá a agregar un campo de selección de item al formulario del widget y cómo agregar la parte visual de este campo a la vista de configuración. Después, el controlador del widget podrá recuperar los datos del item y su valor mediante una solicitud a la API. Una vez recibido, el valor se puede mostrar en la vista del widget.

1. Abra `includes/WidgetForm.php` y agregue el campo `CWidgetFieldMultiSelectItem`. Esto permitirá seleccionar un item en el formulario de configuración.

`ui/modules/lesson_gauge_chart/includes/WidgetForm.php`

```
<?php

namespace Modules\LessonGaugeChart\Includes;

use Zabbix\Widgets\{
    CWidgetField,
    CWidgetForm
};

use Zabbix\Widgets\Fields\{
    CWidgetFieldMultiSelectItem,
    CWidgetFieldTextBox
};

/**
 * Gauge chart widget form.
 */
class WidgetForm extends CWidgetForm {

    public function addFields(): self {
        return $this
            ->addField(
                (new CWidgetFieldMultiSelectItem('itemid', _('Item')))
                    ->setFlags(CWidgetField::FLAG_NOT_EMPTY | CWidgetField::FLAG_LABEL_ASTERISK)
                    ->setMultiple(false)
            )
            ->addField(
                new CWidgetFieldTextBox('description', _('Description'))
            );
    }
}
```

2. Abra `views/widget.edit.php` y agregue el componente visual del campo a la vista de configuración.

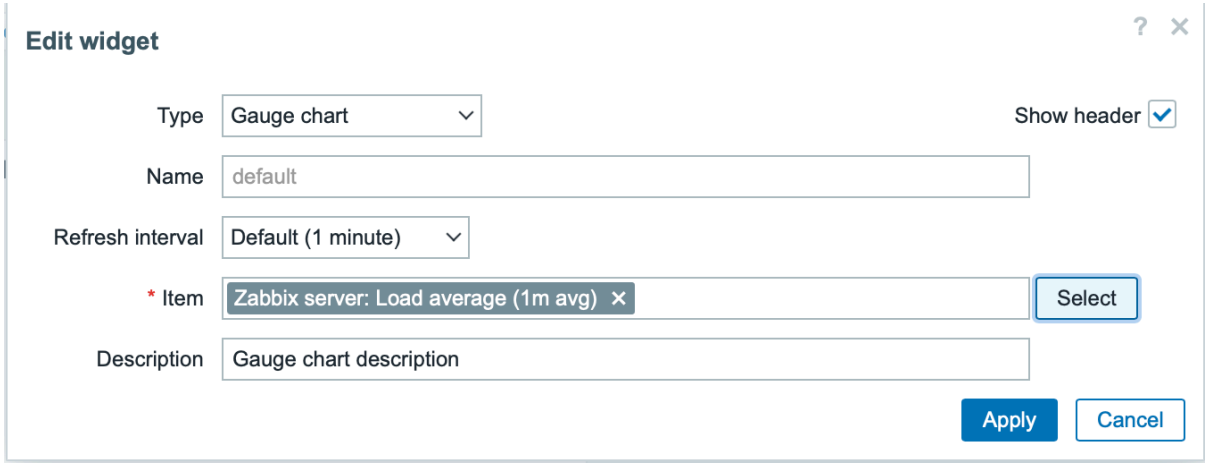
`ui/modules/lesson_gauge_chart/views/widget.edit.php`

```php
<?php

/**
 * Gauge chart widget form view.
 *
 * @var CView $this
 * @var array $data
 */

(new CWidgetFormView($data))
    ->addField(
        new CWidgetFieldMultiSelectItemView($data['fields']['itemid'])
    )
    ->addField(
        new CWidgetFieldTextBoxView($data['fields']['description'])
    )
    ->show();
```

3. Vuelva al dashboard y haga clic en el icono de engranaje del widget para abrir el formulario de configuración del widget.
4. El formulario de configuración del widget ahora contiene un nuevo campo de entrada *Item*. Seleccione el host "Zabbix server" y el item "Load average (1m avg)".



5. Haga clic en *Apply* en el formulario de configuración del widget. Luego haga clic en *Save changes* en la esquina superior derecha para guardar el dashboard.
6. Abra y modifique `actions/WidgetView.php`.

A partir de ahora, el ID del item estará disponible en el controlador del widget en `$this->fields_values['itemid']`. El método `doAction()` del controlador recopila los datos del item (nombre, tipo de valor, unidades) mediante el método de API `item.get` y el último valor del item mediante el método de API `history.get`.

ui/modules/lesson_gauge_chart/actions/WidgetView.php

```php
<?php

namespace Modules\LessonGaugeChart\Actions;

use API,
    CControllerDashboardWidgetView,
    CControllerResponseData;

class WidgetView extends CControllerDashboardWidgetView {

    protected function doAction(): void {
        $db_items = API::Item()->get([
            'output' => ['itemid', 'value_type', 'name', 'units'],
            'itemids' => $this->fields_values['itemid'],
            'webitems' => true,
            'filter' => [
                'value_type' => [ITEM_VALUE_TYPE_UINT64, ITEM_VALUE_TYPE_FLOAT]
```

```
            ]
        ]);

        $value = null;

        if ($db_items) {
            $item = $db_items[0];

            $history = API::History()->get([
                'output' => API_OUTPUT_EXTEND,
                'itemids' => $item['itemid'],
                'history' => $item['value_type'],
                'sortfield' => 'clock',
                'sortorder' => ZBX_SORT_DOWN,
                'limit' => 1
            ]);

            if ($history) {
                $value = convertUnitsRaw([
                    'value' => $history[0]['value'],
                    'units' => $item['units']
                ]);
            }
        }

        $this->setResponse(new CControllerResponseData([
            'name' => $this->getInput('name', $this->widget->getName()),
            'value' => $value,
            'description' => $this->fields_values['description'],
            'user' => [
                'debug_mode' => $this->getDebugMode()
            ]
        ]));
    }
}
```

7. Abra `views/widget.view.php` y agregue el valor del item a la vista del widget.

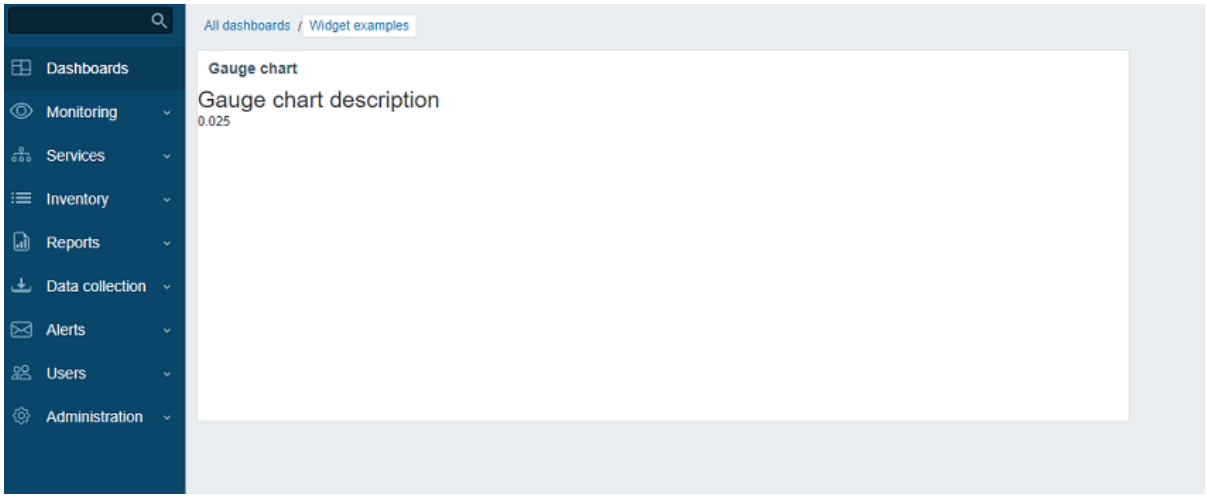
`ui/modules/lesson_gauge_chart/views/widget.view.php`

```
<?php

/**
 * Gauge chart widget view.
 *
 * @var CView $this
 * @var array $data
 */

(new CWidgetView($data))
    ->addItem([
        new CTag('h1', true, $data['description']),
        new CDiv($data['value'] !== null ? $data['value']['value'] : _('No data'))
    ])
    ->show();
```

8. Actualice la página del dashboard. El widget mostrará el último valor del item.



Añadir ajustes de configuración avanzados a una vista de configuración

En esta sección, aprenderá cómo añadir una sección *Configuración avanzada* expandible/plegable con parámetros opcionales, como color, valores mínimo y máximo, unidades y el campo *Description* creado anteriormente.

1. Cree un archivo `Widget.php` en el directorio principal del widget `lesson_gauge_chart` para crear una nueva clase `Widget`.

La clase `Widget` extenderá la clase base `CWidget` para añadir o sobrescribir la configuración predeterminada del widget (en este caso, las traducciones). El JavaScript, proporcionado a continuación, muestra la cadena "No data" en caso de que falten datos. La cadena "No data" está presente en los archivos de traducción de la interfaz de usuario de Zabbix.

Si existen constantes del widget, también se recomienda especificarlas en la clase `Widget`.

`ui/modules/lesson_gauge_chart/Widget.php`

```
<?php

namespace Modules\LessonGaugeChart;

use Zabbix\Core\CWidget;

class Widget extends CWidget {

    public const UNIT_AUTO = 0;
    public const UNIT_STATIC = 1;

    public function getTranslationStrings(): array {
        return [
            'class.widget.js' => [
                'No data' => _('No data')
            ]
        ];
    }
}
```

2. Abra `includes/WidgetForm.php` y añada los nuevos campos *Color* (selector de color), *Min* (campo numérico), *Max* (campo numérico) y *Units* (select), y defina la paleta de colores predeterminada para el selector de color, de modo que pueda utilizarse en los siguientes pasos.

`ui/modules/lesson_gauge_chart/includes/WidgetForm.php`

```
<?php

namespace Modules\LessonGaugeChart\Includes;

use Modules\LessonGaugeChart\Widget;

use Zabbix\Widgets\{
    CWidgetField,
    CWidgetForm
```

```
};

use Zabbix\Widgets\Fields\{
    CWidgetFieldColor,
    CWidgetFieldMultiSelectItem,
    CWidgetFieldNumericBox,
    CWidgetFieldSelect,
    CWidgetFieldTextBox
};

/**
 * Gauge chart widget form.
 */
class WidgetForm extends CWidgetForm {

    public const DEFAULT_COLOR_PALETTE = [
        'FF465C', 'B0AF07', '0EC9AC', '524BBC', 'ED1248', 'D1E754', '2AB5FF', '385CC7', 'EC1594', 'BAE37D',
        '6AC8FF', 'EE2B29', '3CA20D', '6F4BBC', '00A1FF', 'F3601B', '1CAE59', '45CFDB', '894BBC', '6D6D6D'
    ];

    public function addFields(): self {
        return $this
            ->addField(
                (new CWidgetFieldMultiSelectItem('itemid', _('Item')))
                    ->setFlags(CWidgetField::FLAG_NOT_EMPTY | CWidgetField::FLAG_LABEL_ASTERISK)
                    ->setMultiple(false)
            )
            ->addField(
                (new CWidgetFieldColor('chart_color', _('Color')))->setDefault('FF0000')
            )
            ->addField(
                (new CWidgetFieldNumericBox('value_min', _('Min')))
                    ->setDefault(0)
                    ->setFlags(CWidgetField::FLAG_NOT_EMPTY | CWidgetField::FLAG_LABEL_ASTERISK)
            )
            ->addField(
                (new CWidgetFieldNumericBox('value_max', _('Max')))
                    ->setDefault(100)
                    ->setFlags(CWidgetField::FLAG_NOT_EMPTY | CWidgetField::FLAG_LABEL_ASTERISK)
            )
            ->addField(
                (new CWidgetFieldSelect('value_units', _('Units'), [
                    Widget::UNIT_AUTO => _x('Auto', 'history source selection method'),
                    Widget::UNIT_STATIC => _x('Static', 'history source selection method')
                ]))->setDefault(Widget::UNIT_AUTO)
            )
            ->addField(
                (new CWidgetFieldTextBox('value_static_units'))
            )
            ->addField(
                new CWidgetFieldTextBox('description', _('Description'))
            );
    }
}
```

3. Abra `views/widget.edit.php` y añada los componentes visuales de los campos a la vista de configuración.

`ui/modules/lesson_gauge_chart/views/widget.edit.php`

```
<?php

/**
 * Gauge chart widget form view.
 *
```

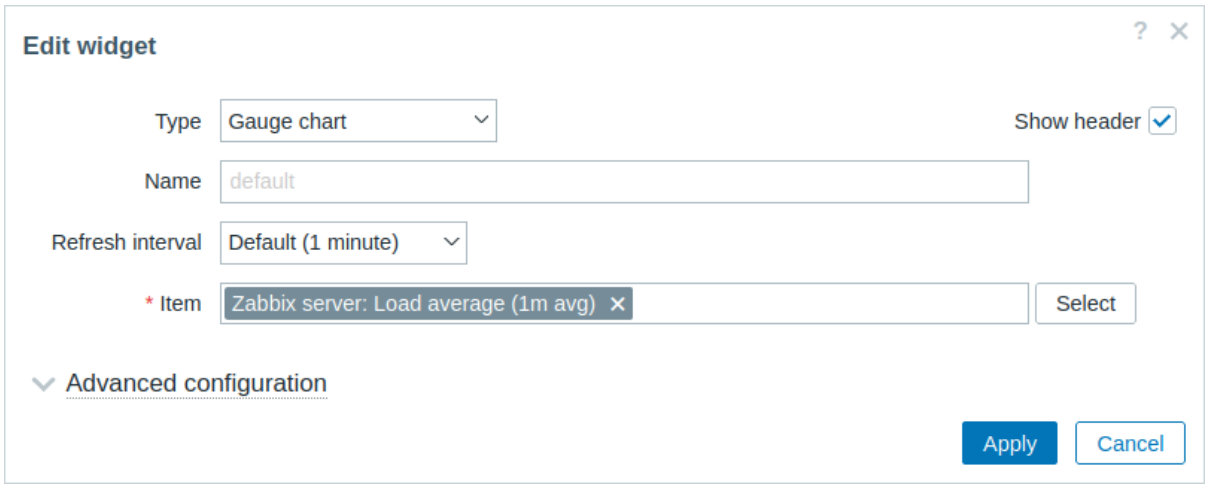
```
 * @var CView $this
 * @var array $data
 */

$lefty_units = new CWidgetFieldSelectView($data['fields']['value_units']);
$lefty_static_units = (new CWidgetFieldTextBoxView($data['fields']['value_static_units']))
    ->setPlaceholder(_('value'))
    ->setWidth(ZBX_TEXTAREA_TINY_WIDTH);

(new CWidgetFormView($data))
    ->addField(
        (new CWidgetFieldMultiSelectItemView($data['fields']['itemid']))
            ->setPopupParameter('numeric', true)
    )
    ->addFieldset(
        (new CWidgetFormFieldsetCollapsibleView(_('Advanced configuration')))
            ->addField(
                new CWidgetFieldColorView($data['fields']['chart_color'])
            )
            ->addField(
                new CWidgetFieldNumericBoxView($data['fields']['value_min'])
            )
            ->addField(
                new CWidgetFieldNumericBoxView($data['fields']['value_max'])
            )
            ->addItem([
                $lefty_units->getLabel(),
                (new CFormField([
                    $lefty_units->getView()->addClass(ZBX_STYLE_FORM_INPUT_MARGIN),
                    $lefty_static_units->getView()
                ]))
            ])
            ->addField(
                new CWidgetFieldTextBoxView($data['fields']['description'])
            )
    )
    ->show();
```

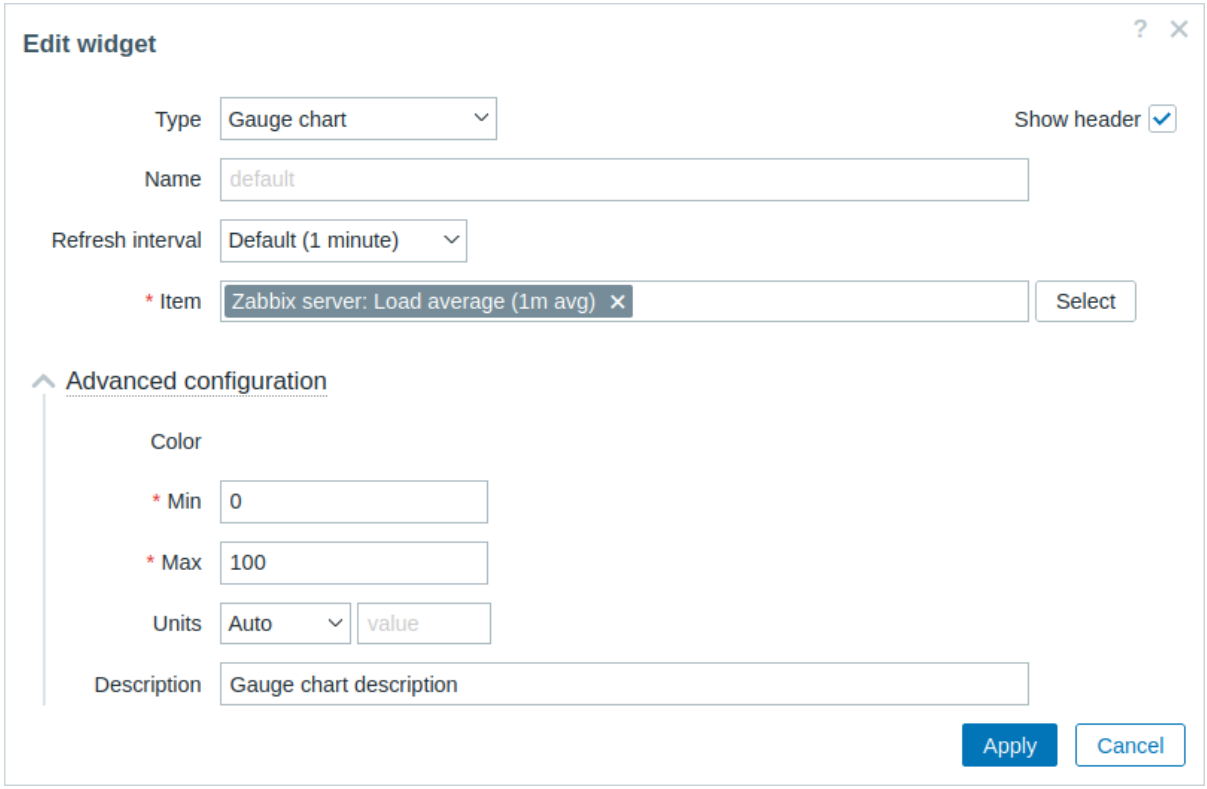
Note:
El método `addField()` de la clase `CWidgetFormView` toma una cadena de clase CSS como segundo parámetro.

4. Vuelva al dashboard, cambie al modo de edición y haga clic en el icono de engranaje del widget para abrir el formulario de configuración del widget. El formulario de configuración del widget ahora contiene una nueva sección expandible/plegable *Advanced configuration*.



5. Expanda la sección *Advanced configuration* para ver campos adicionales de configuración del widget. Tenga en cuenta que el campo *Color* todavía no tiene selector de color. Esto se debe a que el selector de color debe inicializarse con JavaScript,

que se añadirá en la siguiente sección: *Añadir JavaScript al widget*.



Añadir JavaScript al widget

En esta sección, aprenderá cómo añadir un gráfico de indicador, creado con JavaScript, que muestra si el último valor es normal o demasiado alto/demasiado bajo.

1. Cree un archivo `widget.edit.js.php` en el directorio `views`.

JavaScript se encargará de inicializar el selector de color en la vista de configuración.

`ui/modules/lesson_gauge_chart/views/widget.edit.js.php`

```
<?php

use Modules\LessonGaugeChart\Widget;

?>

window.widget_lesson_gauge_chart_form = new class {

    init({color_palette}) {
        this._unit_select = document.getElementById('value_units');
        this._unit_value = document.getElementById('value_static_units');

        this._unit_select.addEventListener('change', () => this.updateForm());

        colorPalette.setThemeColors(color_palette);

        for (const colorpicker of jQuery('.<?= ZBX_STYLE_COLOR_PICKER ?> input')) {
            jQuery(colorpicker).colorpicker();
        }

        const overlay = overlays_stack.getById('widget_properties');

        for (const event of ['overlay.reload', 'overlay.close']) {
            overlay.$dialogue[0].addEventListener(event, () => { jQuery.colorpicker('hide'); });
        }

        this.updateForm();
    }
```

```
    updateForm() {
        this._unit_value.disabled = this._unit_select.value == <?= Widget::UNIT_AUTO ?>;
    }
};
```

2. Abra `views/widget.edit.php` y añada el archivo `widget.edit.js.php` con el JavaScript a la vista de configuración. Para ello, use el método `includeJsFile()`. Para añadir JavaScript en línea, use el método `addJavaScript()`.

`ui/modules/lesson_gauge_chart/views/widget.edit.php`

```
<?php

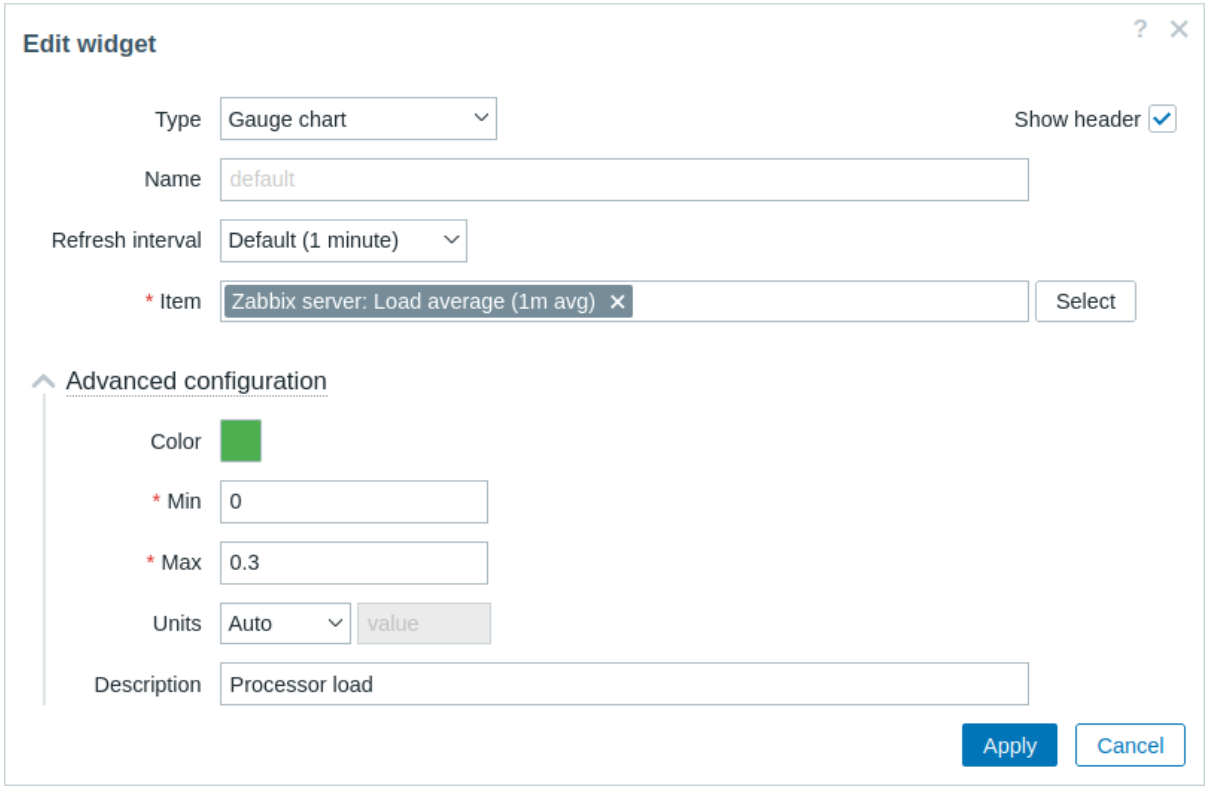
/**
 * Gauge chart widget form view.
 *
 * @var CView $this
 * @var array $data
 */

use Modules\LessonGaugeChart\Includes\WidgetForm;

$lefty_units = new CWidgetFieldSelectView($data['fields']['value_units']);
$lefty_static_units = (new CWidgetFieldTextBoxView($data['fields']['value_static_units']))
    ->setPlaceholder(_('value'))
    ->setWidth(ZBX_TEXTAREA_TINY_WIDTH);

(new CWidgetFormView($data))
    ->addField(
        (new CWidgetFieldMultiSelectItemView($data['fields']['itemid']))
            ->setPopupParameter('numeric', true)
    )
    ->addFieldset(
        (new CWidgetFormFieldsetCollapsibleView(_('Advanced configuration')))
            ->addField(
                new CWidgetFieldColorView($data['fields']['chart_color'])
            )
            ->addField(
                new CWidgetFieldNumericBoxView($data['fields']['value_min'])
            )
            ->addField(
                new CWidgetFieldNumericBoxView($data['fields']['value_max'])
            )
            ->addItem([
                $lefty_units->getLabel(),
                (new CFormField([
                    $lefty_units->getView()->addClass(ZBX_STYLE_FORM_INPUT_MARGIN),
                    $lefty_static_units->getView()
                ]))
            ])
            ->addField(
                new CWidgetFieldTextBoxView($data['fields']['description'])
            )
    )
    ->includeJsFile('widget.edit.js.php')
    ->addJavaScript('widget_lesson_gauge_chart_form.init('.json_encode([
        'color_palette' => WidgetForm::DEFAULT_COLOR_PALETTE
    ], JSON_THROW_ON_ERROR).');')
    ->show();
```

3. Vuelva al dashboard, haga clic en el icono de engranaje del widget para abrir el formulario de configuración del widget. Ahora, expanda la sección *Advanced configuration* para ver el selector de color inicializado. Rellene los campos con valores y seleccione un color para el gráfico de indicador.



4. Haga clic en *Apply* en el formulario de configuración del widget. Luego haga clic en *Save changes* en la esquina superior derecha para guardar el dashboard.

5. Abra `actions/WidgetView.php` y actualice el controlador.

La propiedad `$this->fields_values` ahora contiene los valores de todos los campos de *Advanced configuration*. Finalice el controlador para permitir pasar la configuración y el valor del item seleccionado a la vista del widget.

`ui/modules/lesson_gauge_chart/actions/WidgetView.php`

```
<?php

namespace Modules\LessonGaugeChart\Actions;

use API,
    CControllerDashboardWidgetView,
    CControllerResponseData;

class WidgetView extends CControllerDashboardWidgetView {

    protected function doAction(): void {
        $db_items = API::Item()->get([
            'output' => ['itemid', 'value_type', 'name', 'units'],
            'itemids' => $this->fields_values['itemid'],
            'webitems' => true,
            'filter' => [
                'value_type' => [ITEM_VALUE_TYPE_UINT64, ITEM_VALUE_TYPE_FLOAT]
            ]
        ]);

        $history_value = null;

        if ($db_items) {
            $item = $db_items[0];

            $history = API::History()->get([
                'output' => API_OUTPUT_EXTEND,
                'itemids' => $item['itemid'],
                'history' => $item['value_type'],
                'sortfield' => 'clock',
```

```
                'sortorder' => ZBX_SORT_DOWN,
                'limit' => 1
            ]);

            if ($history) {
                $history_value = convertUnitsRaw([
                    'value' => $history[0]['value'],
                    'units' => $item['units']
                ]);
            }
        }

        $this->setResponse(new CControllerResponseData([
            'name' => $this->getInput('name', $this->widget->getName()),
            'history' => $history_value,
            'fields_values' => $this->fields_values,
            'user' => [
                'debug_mode' => $this->getDebugMode()
            ]
        ]));
    }
}
```

6. Abra y modifique `views/widget.view.php`.

Debe crear un contenedor para el gráfico de indicador, que dibujará en los siguientes pasos, y un contenedor para la descripción.

Para pasar valores a JavaScript como un objeto JSON, use el método `setVar()`.

`ui/modules/lesson_gauge_chart/views/widget.view.php`

```
<?php

/**
 * Gauge chart widget view.
 *
 * @var CView $this
 * @var array $data
 */

(new CWidgetView($data))
    ->addItem([
        (new CDiv())->addClass('chart'),
        $data['fields_values']['description']
            ? (new CDiv($data['fields_values']['description']))->addClass('description')
            : null
    ])
    ->setVar('history', $data['history'])
    ->setVar('fields_values', $data['fields_values'])
    ->show();
```

7. Cree un nuevo directorio `assets` en el directorio `lesson_gauge_chart`. Este directorio se usará para almacenar JavaScript, CSS y, potencialmente, otros recursos, como fuentes o imágenes.
8. Para el JavaScript de la vista del widget, cree un directorio `js` en el directorio `assets`.
9. Cree un archivo `class.widget.js` en el directorio `assets/js`.

Esta clase JavaScript del widget extenderá la clase JavaScript base de todos los widgets del dashboard: `CWidget`.

El dashboard depende de una implementación correcta de un widget y se comunica con él mediante las llamadas a los métodos JavaScript correspondientes. El dashboard también espera que el widget genere eventos cuando ocurre alguna interacción. Por lo tanto, la clase `CWidget` contiene un conjunto de métodos con la implementación predeterminada del comportamiento del widget, que puede personalizarse extendiendo la clase.

En este caso, es necesaria cierta personalización, por lo que se implementará lógica personalizada para el siguiente comportamiento del widget:

- inicialización del widget, responsable de definir el estado inicial del widget (consulte el método `onInitialize()`);
- visualización del contenido del widget (es decir, el dibujo del gráfico de indicador) si el proceso de actualización del widget se ha completado correctamente y sin errores (consulte el método `processUpdateResponse(response)` y los métodos relacionados `_resizeChart()` y `_updatedChart()`)
- redimensionamiento del widget (consulte el método `onResize()` y el método relacionado `_resizeChart()`)

Para otros aspectos del widget de gráfico de indicador, se utilizará la implementación predeterminada del comportamiento del widget. Para obtener más información sobre los métodos JavaScript de la clase `CWidget`, consulte: JavaScript.

Dado que este JavaScript es necesario para la vista del widget, debe cargarse junto con la página del dashboard. Para habilitar la carga de JavaScript, deberá actualizar los parámetros `assets/js` y `js_class` en el archivo `manifest.json` como se muestra en el paso 10.

`ui/modules/lesson_gauge_chart/assets/js/class.widget.js`

```
class WidgetLessonGaugeChart extends CWidget {

    static UNIT_AUTO = 0;
    static UNIT_STATIC = 1;

    onInitialize() {
        super.onInitialize();

        this._refresh_frame = null;
        this._chart_container = null;
        this._canvas = null;
        this._chart_color = null;
        this._min = null;
        this._max = null;
        this._value = null;
        this._last_value = null;
        this._units = '';
    }

    processUpdateResponse(response) {
        if (response.history === null) {
            this._value = null;
            this._units = '';
        }
        else {
            this._value = Number(response.history.value);
            this._units = response.fields_values.value_units == WidgetLessonGaugeChart.UNIT_AUTO
                ? response.history.units
                : response.fields_values.value_static_units;
        }

        this._chart_color = response.fields_values.chart_color;
        this._min = Number(response.fields_values.value_min);
        this._max = Number(response.fields_values.value_max);

        super.processUpdateResponse(response);
    }

    setContents(response) {
        if (this._canvas === null) {
            super.setContents(response);

            this._chart_container = this._body.querySelector('.chart');
            this._chart_container.style.height =
                `${this._getContentsSize().height - this._body.querySelector('.description').clientHeight}
            this._canvas = document.createElement('canvas');

            this._chart_container.appendChild(this._canvas);

            this._resizeChart();
```

```
        }

        this._updatedChart();
    }

    onResize() {
        super.onResize();

        if (this._state === WIDGET_STATE_ACTIVE) {
            this._resizeChart();
        }
    }

    _resizeChart() {
        const ctx = this._canvas.getContext('2d');
        const dpr = window.devicePixelRatio;

        this._canvas.style.display = 'none';
        const size = Math.min(this._chart_container.offsetWidth, this._chart_container.offsetHeight);
        this._canvas.style.display = '';

        this._canvas.width = size * dpr;
        this._canvas.height = size * dpr;

        ctx.scale(dpr, dpr);

        this._canvas.style.width = `${size}px`;
        this._canvas.style.height = `${size}px`;

        this._refresh_frame = null;

        this._updatedChart();
    }

    _updatedChart() {
        if (this._last_value === null) {
            this._last_value = this._min;
        }

        const start_time = Date.now();
        const end_time = start_time + 400;

        const animate = () => {
            const time = Date.now();

            if (time <= end_time) {
                const progress = (time - start_time) / (end_time - start_time);
                const smooth_progress = 0.5 + Math.sin(Math.PI * (progress - 0.5)) / 2;
                let value = this._value !== null ? this._value : this._min;
                value = (this._last_value + (value - this._last_value) * smooth_progress - this._min) / (t

                const ctx = this._canvas.getContext('2d');
                const size = this._canvas.width;
                const char_weight = size / 12;
                const char_shadow = 3;
                const char_x = size / 2;
                const char_y = size / 2;
                const char_radius = (size - char_weight) / 2 - char_shadow;

                const font_ratio = 32 / 100;

                ctx.clearRect(0, 0, size, size);
```

```
                ctx.beginPath();
                ctx.shadowBlur = char_shadow;
                ctx.shadowColor = '#bbb';
                ctx.strokeStyle = '#eee';
                ctx.lineWidth = char_weight;
                ctx.lineCap = 'round';
                ctx.arc(char_x, char_y, char_radius, Math.PI * 0.749, Math.PI * 2.251, false);
                ctx.stroke();

                ctx.beginPath();
                ctx.strokeStyle = `#${this._chart_color}`;
                ctx.lineWidth = char_weight - 2;
                ctx.lineCap = 'round';
                ctx.arc(char_x, char_y, char_radius, Math.PI * 0.75,
                    Math.PI * (0.75 + (1.5 * Math.min(1, Math.max(0, value)))), false
                    );
                ctx.stroke();

                ctx.shadowBlur = 2;
                ctx.fillStyle = '#1f2c33';
                ctx.font = `${(char_radius * font_ratio)|0}px Arial`;
                ctx.textAlign = 'center';
                ctx.textBaseline = 'middle';
                ctx.fillText(`${this._value !== null ? this._value : t('No data')}${this._units}`,
                    char_x, char_y, size - char_shadow * 4 - char_weight * 2
                );

                ctx.fillStyle = '#768d99';
                ctx.font = `${(char_radius * font_ratio * .5)|0}px Arial`;
                ctx.textBaseline = 'top';

                ctx.textAlign = 'left';
                ctx.fillText(`${this._min}${this._min != '' ? this._units : ''}`,
                    char_weight * .75, size - char_weight * 1.25, size / 2 - char_weight
                );

                ctx.textAlign = 'right';
                ctx.fillText(`${this._max}${this._max != '' ? this._units : ''}`,
                    size - char_weight * .75, size - char_weight * 1.25, size / 2 - char_weight
                );

                requestAnimationFrame(animate);
            }
            else {
                this._last_value = this._value;
            }
        };

        requestAnimationFrame(animate);
    }
}
```

10. Abra `manifest.json` y añada:

- el nombre del archivo (`class.widget.js`) al array en la sección `assets/js`;
- el nombre de la clase (`WidgetLessonGaugeChart`) al parámetro `js_class` en la sección `widget`.

La clase `WidgetLessonGaugeChart` ahora se cargará automáticamente con el dashboard.

`ui/modules/lesson_gauge_chart/manifest.json`

```
{
    "manifest_version": 2.0,
    "id": "lesson_gauge_chart",
```

```
    "type": "widget",
    "name": "Gauge chart",
    "namespace": "LessonGaugeChart",
    "version": "1.0",
    "author": "Zabbix",
    "actions": {
        "widget.lesson_gauge_chart.view": {
            "class": "WidgetView"
        }
    },
    "widget": {
        "js_class": "WidgetLessonGaugeChart"
    },
    "assets": {
        "js": ["class.widget.js"]
    }
}
```

Añadir estilos CSS al widget

En esta sección aprenderás cómo añadir estilos CSS personalizados para que el widget tenga un aspecto más atractivo.

1. Para los estilos del widget, crea un nuevo directorio `css` en el directorio `assets`.
2. Crea un archivo `widget.css` en el directorio `assets/css`. Para dar estilo a los elementos del widget, usa el selector `div.dashboard-widget-{widget id}`. Para configurar CSS para todo el widget, usa el selector `form.dashboard-widget-{widg id}`

ui/modules/lesson_gauge_chart/assets/css/widget.css

```
div.dashboard-widget-lesson_gauge_chart {
    display: grid;
    grid-template-rows: 1fr;
    padding: 0;
}

div.dashboard-widget-lesson_gauge_chart .chart {
    display: grid;
    align-items: center;
    justify-items: center;
}

div.dashboard-widget-lesson_gauge_chart .chart canvas {
    background: white;
}

div.dashboard-widget-lesson_gauge_chart .description {
    padding-bottom: 8px;
    font-size: 1.750em;
    line-height: 1.2;
    text-align: center;
}

.dashboard-grid-widget-hidden-header div.dashboard-widget-lesson_gauge_chart .chart {
    margin-top: 8px;
}
```

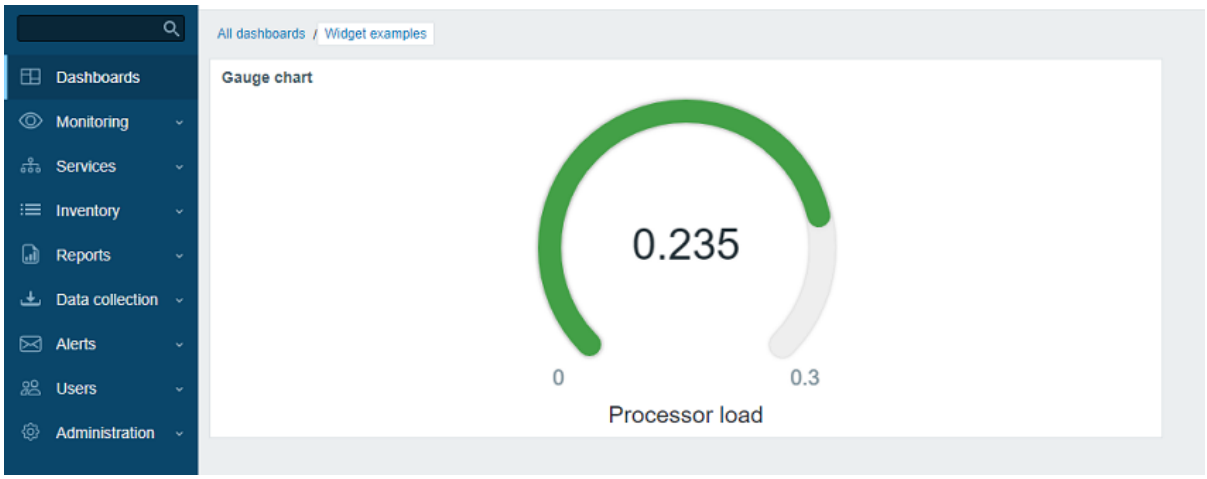
3. Abre `manifest.json` y añade el nombre del archivo CSS (`widget.css`) al array en la sección `assets/css`. Esto permitirá que los estilos CSS definidos en `widget.css` se carguen junto con la página del dashboard.

ui/modules/lesson_gauge_chart/manifest.json

```
{
    "manifest_version": 2.0,
    "id": "lesson_gauge_chart",
    "type": "widget",
    "name": "Gauge chart",
```

```
    "namespace": "LessonGaugeChart",
    "version": "1.0",
    "author": "Zabbix",
    "actions": {
        "widget.lesson_gauge_chart.view": {
            "class": "WidgetView"
        }
    },
    "widget": {
        "js_class": "WidgetLessonGaugeChart"
    },
    "assets": {
        "css": ["widget.css"],
        "js": ["class.widget.js"]
    }
}
```

4. Actualiza la página del dashboard para ver la versión final del widget.



Ejemplos

Esta sección proporciona archivos de módulos y widgets de ejemplo, que puede utilizar como base para sus propios módulos personalizados.

Para utilizar un módulo:

1. Descargue el archivo ZIP.
2. Descomprima el contenido en un directorio separado dentro del directorio *modules* de su instalación del frontend de Zabbix (por ejemplo, *zabbix/ui/modules*).
3. Registre el módulo en el frontend de Zabbix.

Ejemplo de módulo

- Al crear un grupo de equipos, otorgue permisos de lectura a los grupos de usuarios configurados - hg_auto_perm.zip

Ejemplos de widgets

- Widget mínimo - widget_min.zip
- Widget ”Hola, mundo” usando solo CSS - hello_world_css.zip
- Widget ”Hola, mundo” usando solo JavaScript - hello_world_js.zip
- Widget ”Hola, mundo” usando PHP - hello_world_php.zip

Note:
También puede utilizar widgets nativos de Zabbix como ejemplos.

Cambios en el desarrollo de extensiones

Cambios de 7.4 a 8.0

Páginas de manual de Zabbix

Documentación de referencia para las herramientas y servicios de línea de comandos de Zabbix, que enumera los comandos disponibles, las opciones y ejemplos de uso.

zabbix_agent2

```
ZABBIX_AGENT2(8)         Manual del administrador del sistema        ZABBIX_AGENT2(8)

NOMBRE
       zabbix_agent2 - Zabbix agent 2

SINOPSIS
       zabbix_agent2 [-c config-file]
       zabbix_agent2 [-c config-file] [-v] -p
       zabbix_agent2 [-c config-file] [-v] -t item-key
       zabbix_agent2 [-c config-file] -R runtime-option
       zabbix_agent2 [-c config-file] -T
       zabbix_agent2 -h
       zabbix_agent2 -V

DESCRIPCIÓN
       zabbix_agent2 es una aplicación para monitorizar parámetros de varios
       servicios.

OPCIONES
       -c, --config config-file
              Utiliza el archivo de configuración alternativo config-file en
              lugar del predeterminado.

       -R, --runtime-control runtime-option
              Realiza funciones administrativas de acuerdo con runtime-option.

           Opciones de control en tiempo de ejecución
               userparameter_reload
                   Recarga los parámetros de usuario desde el archivo de configuración

               log_level_increase
                   Aumenta el nivel de registro

               log_level_decrease
                   Disminuye el nivel de registro

               help
                   Muestra las opciones de control en tiempo de ejecución disponibles

               metrics
                   Muestra las métricas disponibles
               version
                   Muestra la versión

       -p, --print
              Muestra los items conocidos y sale. Para cada item se usan
```

```
              valores predeterminados genéricos o se proporcionan valores
              predeterminados específicos para las pruebas. Estos valores
              predeterminados se muestran entre corchetes como parámetros de
              clave del item. Los valores devueltos se muestran entre
              corchetes y van precedidos por el tipo del valor devuelto,
              separado por el carácter de barra vertical. Para los parámetros
              de usuario, el tipo siempre es t, ya que el agent no puede
              determinar todos los posibles valores devueltos. Los items que
              se muestran como funcionales no tienen garantizado su
              funcionamiento desde el Zabbix server o zabbix_get al consultar
              un demonio agent en ejecución, ya que los permisos o el entorno
              pueden ser diferentes. Los tipos de valores devueltos son:

          d   Número con parte decimal.

          m   No soportado. Esto puede deberse a la consulta de un item que
              solo funciona en modo activo, como un item de monitorización de
              logs, o un item que requiere múltiples valores recopilados. Los
              problemas de permisos o parámetros de usuario incorrectos
              también pueden dar como resultado el estado de no soportado.

          s   Texto. Longitud máxima no limitada.

          t   Texto. Igual que s.

          u   Entero sin signo.

       -t, --test item-key
              Prueba un único item y sale. Consulte --print para la
              descripción de la salida.

       -v, --verbose
              Habilita la salida detallada al probar un item o mostrar los
              items conocidos.

       -T, --test-config
              Valida el archivo de configuración y sale.

       -h, --help
              Muestra esta ayuda y sale.

       -V, --version
              Muestra información de la versión y sale.

ARCHIVOS
       /usr/local/etc/zabbix_agent2.conf
              Ubicación predeterminada del archivo de configuración de Zabbix
              agent 2 (si no se modificó durante el tiempo de compilación).

VÉASE TAMBIÉN
       Documentación

       zabbix_agentd(8), zabbix_get(1), zabbix_proxy(8), zabbix_sender(1),
       zabbix_server(8), zabbix_js(1), zabbix_web_service(8)

Zabbix                          2022-01-06                ZABBIX_AGENT2(8)
```

zabbix_agentd

```
ZABBIX_AGENTD(8)        Manual del administrador del sistema        ZABBIX_AGENTD(8)
```

NOMBRE
       zabbix_agentd - Daemon de agent de Zabbix

SINOPSIS
       zabbix_agentd [-c config-file]
       zabbix_agentd [-c config-file] -p
       zabbix_agentd [-c config-file] -t item-key
       zabbix_agentd [-c config-file] -R runtime-option
       zabbix_agentd [-c config-file] -T
       zabbix_agentd -h
       zabbix_agentd -V

DESCRIPCIÓN
       zabbix_agentd es un daemon para monitorizar varios parámetros del server.

OPCIONES
       -c, --config config-file
              Utiliza el config-file alternativo en lugar del predeterminado.

       -f, --foreground
              Ejecuta Zabbix agent en primer plano.

       -R, --runtime-control runtime-option
              Realiza funciones administrativas de acuerdo con runtime-option.

           Opciones de control en tiempo de ejecución
               userparameter_reload
                   Recarga los parámetros de usuario desde el archivo de configuración

               log_level_increase[=target]
                   Incrementa el nivel de registro; afecta a todos los procesos
                   si no se especifica target

               log_level_decrease[=target]
                   Disminuye el nivel de registro; afecta a todos los procesos
                   si no se especifica target

           Destinos de control del nivel de registro
               process-type
                   Todos los procesos del tipo especificado (comprobaciones activas,
                   collector, listener)

               process-type,N
                   Tipo de proceso y número (p. ej., listener,3)
               pid Identificador del proceso, hasta 65535. Para valores mayores,
                   especifique target como "process-type,N"

       -p, --print
              Imprime los items conocidos y sale. Para cada item se utilizan
              valores predeterminados genéricos, o se proporcionan valores
              predeterminados específicos para las pruebas. Estos valores
              predeterminados se muestran entre corchetes como parámetros de la
              clave del item. Los valores devueltos se encierran entre corchetes
              y van precedidos por el tipo del valor devuelto, separado por un
              carácter de barra vertical. Para los parámetros de usuario, el tipo
              siempre es t, ya que el agent no puede determinar todos los valores
              de retorno posibles. No se garantiza que los items mostrados como
              funcionales vayan a funcionar desde Zabbix server o zabbix_get al
              consultar un daemon de agent en ejecución, ya que los permisos o el
              entorno pueden ser diferentes. Los tipos de valores devueltos son:

           d   Número con parte decimal.

```
           m   No soportado. Esto puede deberse a la consulta de un item que
               solo funciona en modo activo, como un item de monitorización de
               logs, o un item que requiere múltiples valores recopilados. Los
               problemas de permisos o parámetros de usuario incorrectos también
               pueden dar como resultado el estado no soportado.

           s   Texto. Longitud máxima no limitada.

           t   Texto. Igual que s.

           u   Entero sin signo.

       -t, --test item-key
              Prueba un único item y sale. Consulte --print para la descripción
              de la salida.

       -T, --test-config
              Valida el archivo de configuración y sale.

       -h, --help
              Muestra esta ayuda y sale.

       -V, --version
              Muestra la información de versión y sale.

ARCHIVOS
       /usr/local/etc/zabbix_agentd.conf
              Ubicación predeterminada del archivo de configuración de Zabbix
              agent (si no se modificó durante el tiempo de compilación).

VÉASE TAMBIÉN
       Documentation

       zabbix_get(1),  zabbix_proxy(8),  zabbix_sender(1),   zabbix_server(8),
       zabbix_js(1), zabbix_agent2(8), zabbix_web_service(8)

Zabbix                            2022-01-06                  ZABBIX_AGENTD(8)
```

zabbix_get

```
ZABBIX_GET(1)            Manual general de comandos             ZABBIX_GET(1)

NOMBRE
       zabbix_get - Utilidad de obtención de Zabbix

SINOPSIS
       zabbix_get  -s  host-name-or-IP  [-p  port-number]  [-I IP-address] [-t
       timeout] -k item-key
       zabbix_get -s host-name-or-IP  [-p  port-number]  [-I  IP-address]  [-t
       timeout]   --tls-connect  cert  --tls-ca-file  CA-file  [--tls-crl-file
       CRL-file] [--tls-agent-cert-issuer cert-issuer]  [--tls-agent-cert-sub-
       ject  cert-subject]  --tls-cert-file  cert-file --tls-key-file key-file
       [--tls-cipher13 cipher-string] [--tls-cipher cipher-string] -k item-key
       zabbix_get -s host-name-or-IP  [-p  port-number]  [-I  IP-address]  [-t
       timeout]     --tls-connect    psk    --tls-psk-identity    PSK-identity
       --tls-psk-file PSK-file  [--tls-cipher13  cipher-string]  [--tls-cipher
       cipher-string] -k item-key
       zabbix_get -h
       zabbix_get -V

DESCRIPCIÓN
```

zabbix_get es una utilidad de línea de comandos para obtener datos del Zabbix agent.

OPCIONES

-s, --host host-name-or-IP
Especifica el nombre del host o la dirección IP de un host.

-p, --port port-number
Especifica el número de puerto del agent que se ejecuta en el host. El valor predeterminado es 10050.

-I, --source-address IP-address
Especifica la dirección IP de origen.

-t, --timeout seconds
Especifica el tiempo de espera. Rango válido: 1-600 segundos (predeterminado: 30)

-k, --key item-key
Especifica la clave del item cuyo valor se va a recuperar.

-P, --protocol value
Protocolo utilizado para comunicarse con el agent. Valores:

auto	conectar usando el protocolo JSON, con retroceso y reintento con el protocolo de texto sin formato (predeterminado)
json	conectar usando el protocolo JSON
plaintext	conectar usando el protocolo de texto sin formato donde solo se envía la clave del item (versiones 6.4.x y anteriores)

--tls-connect value
Cómo conectarse al agent. Valores:

unencrypted	conectar sin cifrado (predeterminado)
psk	conectar usando TLS y una clave precompartida
cert	conectar usando TLS y un certificado

--tls-ca-file CA-file
Ruta completa de un archivo que contiene los certificados de la CA de nivel superior para la verificación del certificado del par.

--tls-crl-file CRL-file
Ruta completa de un archivo que contiene certificados revocados.

--tls-agent-cert-issuer cert-issuer
Emisor permitido del certificado del agent.

--tls-agent-cert-subject cert-subject
Sujeto permitido del certificado del agent.

--tls-cert-file cert-file
Ruta completa de un archivo que contiene el certificado o la cadena de certificados.

--tls-key-file key-file
Ruta completa de un archivo que contiene la clave privada.

```
       --tls-psk-identity PSK-identity
              Cadena de identidad PSK.

       --tls-psk-file PSK-file
              Ruta completa de un archivo que contiene la clave precompartida.

       --tls-cipher13 cipher-string
              Cadena de cifrado para OpenSSL 1.1.1 o posterior para TLS 1.3.
              Sobrescribe los criterios predeterminados de selección del conjunto
              de cifrados. Esta opción no está disponible si la versión de
              OpenSSL es inferior a 1.1.1.

       --tls-cipher cipher-string
              Cadena de prioridad de GnuTLS (para TLS 1.2 y superior) o cadena
              de cifrado de OpenSSL (solo para TLS 1.2). Sobrescribe los
              criterios predeterminados de selección del conjunto de cifrados.

       -h, --help
              Muestra esta ayuda y sale.

       -V, --version
              Muestra la información de versión y sale.
ESTADO DE SALIDA
       El estado de salida es 0 si zabbix_get recupera correctamente un valor del
       Zabbix agent. Si la recuperación del valor falla, el estado de salida es 1.

EJEMPLOS
       zabbix_get -s 127.0.0.1 -p 10050 -k "system.cpu.load[all,avg1]"
       zabbix_get   -s   127.0.0.1  -p  10050  -k  "system.cpu.load[all,avg1]"
       --tls-connect    cert     --tls-ca-file     /home/zabbix/zabbix_ca_file
       --tls-agent-cert-issuer   "CN=Signing   CA,OU=IT   operations,O=Example
       Corp,DC=example,DC=com" --tls-agent-cert-subject "CN=server1,OU=IT  op-
       erations,O=Example  Corp,DC=example,DC=com"  --tls-cert-file /home/zab-
       bix/zabbix_get.crt --tls-key-file /home/zabbix/zabbix_get.key
       zabbix_get  -s  127.0.0.1  -p  10050   -k   "system.cpu.load[all,avg1]"
       --tls-connect   psk   --tls-psk-identity   "PSK   ID   Zabbix   agentd"
       --tls-psk-file /home/zabbix/zabbix_agentd.psk

VÉASE TAMBIÉN
       Documentación

       zabbix_agentd(8), zabbix_proxy(8), zabbix_sender(1),  zabbix_server(8),
       zabbix_js(1), zabbix_agent2(8), zabbix_web_service(8)

Zabbix                            2022-01-06                      ZABBIX_GET(1)
```

zabbix_js

```
ZABBIX_JS(1)                Manual general de comandos                ZABBIX_JS(1)

NOMBRE
       zabbix_js - Utilidad JS de Zabbix

SINOPSIS
       zabbix_js -s script-file -p input-param [-l log-level] [-t timeout]
       zabbix_js -s script-file -i input-file [-l log-level] [-t timeout]
       zabbix_js -h
       zabbix_js -V

DESCRIPCIÓN
```

```
       zabbix_js es una utilidad de línea de comandos que puede utilizarse para
       probar scripts embebidos.

OPCIONES
       -s, --script script-file
              Especifica el nombre del archivo del script que se va a ejecutar.
              Si se especifica '-' como nombre de archivo, el script se leerá
              desde stdin.

       -p, --param input-param
              Especifica el parámetro de entrada que se pasa al script. El
              parámetro se pasa literalmente como una cadena y está disponible
              en el script mediante la variable 'value'. Es mutuamente
              excluyente con -i.

       -i, --input input-file
              Especifica un archivo que contiene el parámetro de entrada pasado
              al script. El contenido del archivo se pasa literalmente como una
              cadena y está disponible en el script mediante la variable
              'value'. Si se especifica '-' como nombre de archivo, la entrada
              se leerá desde stdin. Es mutuamente excluyente con -p.

       -w, --webdriver url
              Especifica la URL del webdriver.

       -l, --loglevel log-level
              Especifica el nivel de registro (rango: 0-5).

       -t, --timeout timeout
              Especifica el tiempo de espera en segundos. Rango válido: 1-600
              segundos (predeterminado: 10)

       -h, --help
              Muestra esta ayuda y sale.

       -V, --version
              Muestra la información de versión y sale.

EJEMPLOS
       zabbix_js -s script-file.js -p example

VÉASE TAMBIÉN
       Documentación

       zabbix_agentd(8), zabbix_proxy(8), zabbix_sender(1),  zabbix_server(8),
       zabbix_get(1), zabbix_agent2(8), zabbix_web_service(8)

Zabbix                            2022-01-06                       ZABBIX_JS(1)
```

zabbix_proxy

```
ZABBIX_PROXY(8)             Manual del administrador del sistema            ZABBIX_PROXY(8)

NOMBRE
       zabbix_proxy - daemon proxy de Zabbix

SINOPSIS
       zabbix_proxy [-c archivo-de-configuración]
       zabbix_proxy [-c archivo-de-configuración] -R opción-en-tiempo-de-ejecución
       zabbix_proxy [-c archivo-de-configuración] -T
       zabbix_proxy -h
```

```
       zabbix_proxy -V

DESCRIPCIÓN
       zabbix_proxy es un daemon que recopila datos de monitorización de los dispositivos y
       los envía al server de Zabbix.

OPCIONES
       -c, --config archivo-de-configuración
              Usa el archivo-de-configuración alternativo en lugar del predeterminado.

       -f, --foreground
              Ejecuta Zabbix proxy en primer plano.

       -R, --runtime-control opción-en-tiempo-de-ejecución
              Realiza funciones administrativas según la opción-en-tiempo-de-ejecución.

           Opciones de control en tiempo de ejecución
               config_cache_reload
                   Recarga la caché de configuración. Se ignora si la caché se está cargando
                   actualmente. El Zabbix proxy activo se conectará al server de Zabbix y solicitará
                   los datos de configuración. Se usará el archivo de configuración predeterminado
                   (a menos que se especifique la opción -c) para localizar el archivo PID y se enviará
                   la señal al proceso listado en el archivo PID.

               snmp_cache_reload
                   Recarga la caché SNMP.

               housekeeper_execute
                   Ejecuta el housekeeper. Se ignora si el housekeeper se está ejecutando actualmente.

               diaginfo[=section]
                   Registra información interna de diagnóstico de la sección especificada. La sección pued
                   historycache, preprocessing, locks.
                   De forma predeterminada, se registra la información de diagnóstico de todas las seccion

               prof_enable[=target]
                   Habilita el perfilado; afecta a todos los procesos si no se especifica target

               prof_disable[=target]
                   Deshabilita el perfilado; afecta a todos los procesos si no se especifica target

               log_level_increase[=target]
                   Aumenta el nivel de registro; afecta a todos los procesos si no se especifica target.

               log_level_decrease[=target]
                   Disminuye el nivel de registro; afecta a todos los procesos si no se especifica target.

           Destinos de control del nivel de registro
               process-type
                   Todos los procesos del tipo especificado (availability manager,
                   browser poller, configuration syncer, data sender, discovery manager,
                   history syncer, housekeeper, http poller, icmp pinger, internal poller,
                   ipmi manager, ipmi poller, java poller, odbc poller, poller, agent poller,
                   http agent poller, snmp poller, preprocessing manager, preprocessing worker,
                   self-monitoring, snmp trapper, supervisor, task manager, trapper,
                   unreachable poller, vmware collector)

               process-type,N
                   Tipo de proceso y número (por ejemplo, poller,3)

               pid Identificador de proceso

           Destinos de control del perfilado
```

```
              process-type
                  Todos los procesos del tipo especificado (availability manager,
                  browser poller, configuration syncer, data sender, discovery manager,
                  history syncer, housekeeper, http poller, icmp pinger, internal poller,
                  ipmi manager, ipmi poller, java poller, odbc poller, poller, agent poller,
                  http agent poller, snmp poller, preprocessing manager, self-monitoring,
                  snmp trapper, task manager, trapper, unreachable poller, vmware collector)

              process-type,N
                  Tipo de proceso y número (por ejemplo, history syncer,1)

              pid Identificador de proceso

              scope
                  (rwlock, mutex, processing) puede usarse con process-type
                  (por ejemplo, history syncer,1,processing)

       -T, --test-config
              Valida el archivo de configuración y sale.

       -h, --help
              Muestra esta ayuda y sale.

       -V, --version
              Muestra la información de la versión y sale.

ARCHIVOS
       /usr/local/etc/zabbix_proxy.conf
              Ubicación predeterminada del archivo de configuración de Zabbix proxy (si no se
              modificó durante la compilación).

VÉASE TAMBIÉN
       Documentación

       zabbix_agentd(8), zabbix_get(1), zabbix_sender(1), zabbix_server(8),
       zabbix_js(1), zabbix_agent2(8), zabbix_web_service(8)

Zabbix                            2022-01-06                    ZABBIX_PROXY(8)
```

zabbix_sender

```
ZABBIX_SENDER(1)            Manual de comandos generales           ZABBIX_SENDER(1)

NOMBRE
       zabbix_sender - utilidad de envío de Zabbix

SINOPSIS
       zabbix_sender [-v] -z server [-p port] [-I IP-address] [-t timeout] -s
       host -k key -o value
       zabbix_sender [-v] -z server [-p port] [-I IP-address] [-t timeout] [-s
       host] [-T] [-N] [-r] -i input-file
       zabbix_sender [-v] -c config-file [-z server] [-p port] [-I IP-address]
       [-t timeout] [-s host] -k key -o value
       zabbix_sender [-v] -c config-file [-z server] [-p port] [-I IP-address]
       [-t timeout] [-s host] [-T] [-N] [-r] -i input-file
       zabbix_sender [-v] -z server [-p port] [-I IP-address] [-t timeout] -s
       host --tls-connect cert --tls-ca-file CA-file [--tls-crl-file CRL-file]
       [--tls-server-cert-issuer cert-issuer] [--tls-server-cert-subject
       cert-subject] --tls-cert-file cert-file --tls-key-file key-file
       [--tls-cipher13 cipher-string] [--tls-cipher cipher-string] -k key -o
       value
```

```
zabbix_sender [-v] -z server [-p port] [-I IP-address] [-t timeout] [-s
host] --tls-connect cert --tls-ca-file CA-file [--tls-crl-file
CRL-file] [--tls-server-cert-issuer cert-issuer]
[--tls-server-cert-subject cert-subject] --tls-cert-file cert-file
--tls-key-file key-file [--tls-cipher13 cipher-string] [--tls-cipher
cipher-string] [-T] [-N] [-r] -i input-file
zabbix_sender [-v] -c config-file [-z server] [-p port] [-I IP-address]
[-t timeout] [-s host] --tls-connect cert --tls-ca-file CA-file
[--tls-crl-file CRL-file] [--tls-server-cert-issuer cert-issuer]
[--tls-server-cert-subject cert-subject] --tls-cert-file cert-file
--tls-key-file key-file [--tls-cipher13 cipher-string] [--tls-cipher
cipher-string] -k key -o value
zabbix_sender [-v] -c config-file [-z server] [-p port] [-I IP-address]
[-t timeout] [-s host] --tls-connect cert --tls-ca-file CA-file
[--tls-crl-file CRL-file] [--tls-server-cert-issuer cert-issuer]
[--tls-server-cert-subject cert-subject] --tls-cert-file cert-file
--tls-key-file key-file [--tls-cipher13 cipher-string] [--tls-cipher
cipher-string] [-T] [-N] [-r] -i input-file
zabbix_sender [-v] -z server [-p port] [-I IP-address] [-t timeout] -s
host --tls-connect psk --tls-psk-identity PSK-identity --tls-psk-file
PSK-file [--tls-cipher13 cipher-string] [--tls-cipher cipher-string] -k
key -o value
zabbix_sender [-v] -z server [-p port] [-I IP-address] [-t timeout] [-s
host] --tls-connect psk --tls-psk-identity PSK-identity --tls-psk-file
PSK-file [--tls-cipher13 cipher-string] [--tls-cipher cipher-string]
[-T] [-N] [-r] -i input-file
zabbix_sender [-v] -c config-file [-z server] [-p port] [-I IP-address]
[-t timeout] [-s host] --tls-connect psk --tls-psk-identity PSK-iden-
tity --tls-psk-file PSK-file [--tls-cipher13 cipher-string] [--tls-ci-
pher cipher-string] -k key -o value
zabbix_sender [-v] -c config-file [-z server] [-p port] [-I IP-address]
[-t timeout] [-s host] --tls-connect psk --tls-psk-identity PSK-iden-
tity --tls-psk-file PSK-file [--tls-cipher13 cipher-string] [--tls-ci-
pher cipher-string] [-T] [-N] [-r] -i input-file
zabbix_sender -h
zabbix_sender -V
```

DESCRIPCIÓN

zabbix_sender es una utilidad de línea de comandos para enviar datos de monitorización al server o proxy de Zabbix. En el server de Zabbix debe crearse un item de tipo Zabbix trapper con la clave correspondiente. Tenga en cuenta que los valores entrantes solo se aceptarán de los hosts especificados en el campo Allowed hosts de este item.

OPCIONES

-c, --config config-file

Utiliza config-file. zabbix_sender lee los detalles del server del archivo de configuración de agentd. De forma predeterminada, zabbix_sender no lee ningún archivo de configuración. Solo se admiten los parámetros Hostname, ServerActive, SourceIP, TLSConnect, TLSCAFile, TLSCRLFile, TLSServerCertIssuer, TLSServerCertSubject, TLSCertFile, TLSKeyFile, TLSPSKIdentity y TLSPSKFile. El Hostname definido mediante el parámetro HostnameItem no se utilizará; en este caso, el nombre del host debe especificarse mediante la línea de comandos (consulte la opción -s). Todas las direcciones definidas en el parámetro de configuración ServerActive del agent se utilizan para enviar datos. Si el envío de datos por lotes falla en una dirección, los lotes siguientes no se enviarán a esa dirección.

-z, --zabbix-server server

Nombre de host o dirección IP del server de Zabbix. Si un host está monitorizado por un proxy, debe utilizarse el nombre de host

o la dirección IP del proxy. Cuando se utiliza junto con --config, reemplaza las entradas del parámetro ServerActive especificadas en el archivo de configuración de agentd.

-p, --port port
Especifica el número de puerto en el que se ejecuta el trapper del server de Zabbix. El valor predeterminado es 10051. Cuando se utiliza junto con --config, reemplaza las entradas de puerto del parámetro ServerActive especificadas en el archivo de configuración de agentd.

-I, --source-address IP-address
Especifica la dirección IP de origen. Cuando se utiliza junto con --config, reemplaza el parámetro SourceIP especificado en el archivo de configuración de agentd.

-t, --timeout seconds
Especifica el tiempo de espera. Rango válido: 1-300 segundos (predeterminado: 60)

-s, --host host
Especifica el nombre del host al que pertenece el item (tal como está registrado en el frontend de Zabbix). La dirección IP y el nombre DNS del host no funcionarán. Cuando se utiliza junto con --config, reemplaza el parámetro Hostname especificado en el archivo de configuración de agentd.

-k, --key key
Especifica la clave del item al que se enviará el valor.

-o, --value value
Especifica el valor del item.

-i, --input-file input-file
Carga los valores desde el archivo de entrada. Especifique - como <input-file> para leer los valores desde la entrada estándar. Cada línea del archivo contiene, separados por espacios en blanco: <hostname> <key> <value>. Cada valor debe especificarse en su propia línea. Cada línea debe contener 3 entradas separadas por espacios en blanco: <hostname> <key> <value>, donde "hostname" es el nombre del host monitorizado tal como está registrado en el frontend de Zabbix, "key" es la clave del item de destino y "value" es el valor que se enviará. Especifique - como <hostname> para utilizar el nombre de host del archivo de configuración del agent o del argumento --host.

Un ejemplo de línea de un archivo de entrada:

"Linux DB3" db.connections 43

El tipo de valor debe estar configurado correctamente en la configuración del item del frontend de Zabbix. zabbix_sender enviará hasta 250 valores en una conexión. El contenido del archivo de entrada debe estar codificado en UTF-8. Todos los valores del archivo de entrada se envían en orden secuencial, de arriba abajo. Las entradas deben tener el siguiente formato:

- Se admiten entradas entre comillas y sin comillas.
- Las comillas dobles son el carácter de delimitación.
- Las entradas que contienen espacios en blanco deben estar entre comillas.

- Las comillas dobles y las barras invertidas dentro de una entrada entre comillas deben escaparse con una barra invertida.

- No se admite el escape en entradas sin comillas.

- Las secuencias de escape de salto de línea (\n) se admiten en cadenas entre comillas.

- Las secuencias de escape de salto de línea se eliminan del final de una entrada.

-T, --with-timestamps
Esta opción solo puede utilizarse con la opción --input-file.

Cada línea del archivo de entrada debe contener 4 entradas separadas por espacios en blanco: <hostname> <key> <timestamp> <value>. La marca de tiempo debe especificarse en formato de marca de tiempo Unix. Si el item de destino tiene triggers que hacen referencia a él, todas las marcas de tiempo deben estar en orden creciente; de lo contrario, el cálculo de eventos no será correcto.

Un ejemplo de línea de un archivo de entrada:

"Linux DB3" db.connections 1429533600 43

Para obtener más información, consulte la opción --input-file.

Si se envía un valor con marca de tiempo para un host que se encuentra en un mantenimiento de tipo “no data”, este valor se descartará; sin embargo, es posible enviar un valor con marca de tiempo correspondiente a un período de mantenimiento expirado y este será aceptado.

-N, --with-ns
Esta opción solo puede utilizarse con la opción --with-timestamps.

Cada línea del archivo de entrada debe contener 5 entradas separadas por espacios en blanco: <hostname> <key> <timestamp> <ns> <value>.

Un ejemplo de línea de un archivo de entrada:

"Linux DB3" db.connections 1429533600 7402561 43

Para obtener más información, consulte la opción --input-file.

-r, --real-time
Envía los valores uno por uno tan pronto como se reciben. Esto puede utilizarse al leer desde la entrada estándar.

--tls-connect value
Cómo conectarse al server o proxy. Valores:

unencrypted
conectarse sin cifrado (predeterminado)

psk conectarse mediante TLS y una clave precompartida

cert conectarse mediante TLS y un certificado

```
       --tls-ca-file CA-file
              Ruta completa de un archivo que contiene los certificados de la
              CA de nivel superior para la verificación del certificado del
              par.

       --tls-crl-file CRL-file
              Ruta completa de un archivo que contiene certificados revocados.

       --tls-server-cert-issuer cert-issuer
              Emisor de certificado del server permitido.

       --tls-server-cert-subject cert-subject
              Sujeto del certificado del server permitido.

       --tls-cert-file cert-file
              Ruta completa de un archivo que contiene el certificado o la
              cadena de certificados.

       --tls-key-file key-file
              Ruta completa de un archivo que contiene la clave privada.

       --tls-psk-identity PSK-identity
              Cadena de identidad PSK.

       --tls-psk-file PSK-file
              Ruta completa de un archivo que contiene la clave precompartida.

       --tls-cipher13 cipher-string
              Cadena de cifrado para OpenSSL 1.1.1 o posterior para TLS 1.3.
              Reemplaza los criterios de selección predeterminados del conjunto
              de cifrado. Esta opción no está disponible si la versión de
              OpenSSL es anterior a 1.1.1.

       --tls-cipher cipher-string
              Cadena de prioridad de GnuTLS (para TLS 1.2 y posteriores) o
              cadena de cifrado de OpenSSL (solo para TLS 1.2). Reemplaza los
              criterios de selección predeterminados del conjunto de cifrado.

       -v, --verbose
              Modo detallado; utilice -vv para obtener más detalles.

       -h, --help
              Muestra esta ayuda y sale.

       -V, --version
              Muestra la información de versión y sale.

ESTADO DE SALIDA
       El estado de salida es 0 si los valores se enviaron y todos fueron
       procesados correctamente por el server. Si se enviaron datos, pero el
       procesamiento de al menos uno de los valores falló, el estado de salida
       es 2. Si falló el envío de datos, el estado de salida es 1.

EJEMPLOS
       zabbix_sender -c  /etc/zabbix/zabbix_agentd.conf  -k  mysql.queries  -o
       342.45
              Envía 342.45 como valor del item mysql.queries del host
              monitorizado. Utiliza el host monitorizado y el server de Zabbix
              definidos en el archivo de configuración del agent.

       zabbix_sender -c /etc/zabbix/zabbix_agentd.conf -s "Monitored Host"  -k
       mysql.queries -o 342.45
```

```
              Envía 342.45 como valor del item mysql.queries del host Monitored
              Host utilizando el server de Zabbix definido en el archivo de
              configuración del agent.

       zabbix_sender -z 192.168.1.113 -i data_values.txt
              Envía los valores del archivo data_values.txt al server de Zabbix
              con la IP 192.168.1.113. Los nombres de host y las claves están
              definidos en el archivo.

       echo "- hw.serial.number  1287872261  SQ4321ASDF"  |  zabbix_sender  -c
       /usr/local/etc/zabbix_agentd.conf -T -i -
              Envía un valor con marca de tiempo desde la línea de comandos al
              server de Zabbix especificado en el archivo de configuración del
              agent. El guion en los datos de entrada indica que también debe
              utilizarse el nombre de host del mismo archivo de configuración.

       echo '"Zabbix server" trapper.item ""' | zabbix_sender -z 192.168.1.113
       -p 10000 -i -
              Envía un valor vacío de un item al server de Zabbix con dirección
              IP 192.168.1.113 en el puerto 10000 desde la línea de comandos.
              Los valores vacíos deben indicarse mediante comillas dobles
              vacías.

       zabbix_sender -z 192.168.1.113  -s "Monitored Host" -k mysql.queries -o
       342.45  --tls-connect  cert  --tls-ca-file  /home/zabbix/zabbix_ca_file
       --tls-cert-file      /home/zabbix/zabbix_agentd.crt      --tls-key-file
       /home/zabbix/zabbix_agentd.key
              Envía 342.45 como valor del item mysql.queries del host Monitored
              Host al server con IP 192.168.1.113 mediante TLS con certificado.

       zabbix_sender -z 192.168.1.113  -s "Monitored Host" -k mysql.queries -o
       342.45  --tls-connect  psk  --tls-psk-identity  "PSK  ID Zabbix agentd"
       --tls-psk-file /home/zabbix/zabbix_agentd.psk
              Envía 342.45 como valor del item mysql.queries del host Monitored
              Host al server con IP 192.168.1.113 mediante TLS con una clave
              precompartida (PSK).

VÉASE TAMBIÉN
       Documentación

       zabbix_agentd(8),  zabbix_get(1),  zabbix_proxy(8),   zabbix_server(8),
       zabbix_js(1), zabbix_agent2(8), zabbix_web_service(8)

Zabbix                            2022-01-06                  ZABBIX_SENDER(1)
```

zabbix_server

```
ZABBIX_SERVER(8)            Manual del administrador del sistema           ZABBIX_SERVER(8)

NOMBRE
       zabbix_server - demonio del server de Zabbix

SINOPSIS
       zabbix_server [-c archivo-de-configuración]
       zabbix_server [-c archivo-de-configuración] -R opción-en-tiempo-de-ejecución
       zabbix_server [-c archivo-de-configuración] -T
       zabbix_server -h
       zabbix_server -V

DESCRIPCIÓN
       zabbix_server es el demonio principal del software Zabbix.
```

OPCIONES

-c, --config archivo-de-configuración
Usa el archivo-de-configuración alternativo en lugar del predeterminado.

-f, --foreground
Ejecuta Zabbix server en primer plano.

-R, --runtime-control opción-en-tiempo-de-ejecución
Realiza funciones administrativas según la opción-en-tiempo-de-ejecución.

Opciones de control en tiempo de ejecución

config_cache_reload
Recarga la caché de configuración. Se ignora si la caché se está cargando actualmente. Se usará el archivo de configuración predeterminado (a menos que se especifique la opción -c) para localizar el archivo PID y se enviará la señal al proceso listado en el archivo PID.

snmp_cache_reload
Recarga la caché SNMP.

housekeeper_execute
Ejecuta el housekeeper. Se ignora si el housekeeper se está ejecutando actualmente.

trigger_housekeeper_execute
Ejecuta el trigger housekeeper (elimina problemas de triggers eliminados). Se ignora si el trigger housekeeper se está ejecutando actualmente.

secrets_reload
Recarga los secretos desde Vault. Se ignora si la caché de configuración o los secretos se están cargando actualmente.

diaginfo[=sección]
Registra información interna de diagnóstico de la sección especificada. La sección puede ser historycache, preprocessing, alerting, lld, valuecache, locks y connector. De forma predeterminada, se registra la información de diagnóstico de todas las secciones.

prof_enable[=objetivo]
Habilita el perfilado; afecta a todos los procesos si no se especifica el objetivo

prof_disable[=objetivo]
Deshabilita el perfilado; afecta a todos los procesos si no se especifica el objetivo

ha_status
Muestra el estado del clúster de alta disponibilidad. Solo se puede ejecutar en el nodo activo.

ha_remove_node
ha_remove_node[=nombre|ID] Elimina un nodo de alta disponibilidad por su nombre o ID listado en ha_status. Solo se puede ejecutar en el nodo activo.

ha_set_failover_delay[=retardo]
Establece el retardo de conmutación por error de alta disponibilidad. El retardo puede especificarse usando sufijos de tiempo. Solo se puede ejecutar en el nodo activo.

```
            log_level_increase[=objetivo]
                Aumenta el nivel de registro; afecta a todos los procesos si
                no se especifica el objetivo

            log_level_decrease[=objetivo]
                Disminuye el nivel de registro; afecta a todos los procesos si
                no se especifica el objetivo

        Objetivos de control del nivel de registro
            process-type
                Todos los procesos del tipo especificado (alerter, alert manager,
                alert syncer, availability manager, browser poller, configuration
                syncer, configuration syncer worker, connector manager, connector
                worker, discovery manager, escalator, ha manager, history poller,
                history syncer, housekeeper, http poller, http agent poller,
                icmp pinger, internal poller, ipmi manager, ipmi poller, java poller,
                lld manager, lld worker, odbc poller, poller, agent poller,
                preprocessing manager, preprocessing worker, proxy poller,
                proxy group manager, report manager, report writer,
                self-monitoring, service manager, snmp poller, snmp trapper,
                supervisor, task manager, timer, trapper, trigger housekeeper,
                unreachable poller, vmware collector)

            process-type,N
                Tipo de proceso y número (por ejemplo, poller,3)

            pid Identificador de proceso

        Objetivos de control del perfilado
            process-type
                Todos los procesos del tipo especificado (alerter, alert manager,
                alert syncer, availability manager, browser poller, configuration
                syncer, configuration syncer worker, connector manager, connector
                worker, discovery manager, escalator, ha manager, history poller,
                history syncer, housekeeper, http poller, http agent poller,
                icmp pinger, internal poller, ipmi manager, ipmi poller, java poller,
                lld manager, lld worker, odbc poller, poller, agent poller,
                preprocessing manager, proxy poller, proxy group manager,
                report manager, report writer, self-monitoring, service manager,
                snmp poller, snmp trapper, task manager, timer, trapper,
                trigger housekeeper, unreachable poller, vmware collector)

            process-type,N
                Tipo de proceso y número (por ejemplo, history syncer,1)

            pid Identificador de proceso

            scope
                (rwlock, mutex, processing) puede usarse con process-type
                (por ejemplo, history syncer,1,processing)

    -T, --test-config
           Valida el archivo de configuración y sale.

    -h, --help
           Muestra esta ayuda y sale.

    -V, --version
           Muestra información de la versión y sale.

ARCHIVOS
    /usr/local/etc/zabbix_server.conf
```

Ubicación predeterminada del archivo de configuración de Zabbix
server (si no se modificó durante la compilación).

VÉASE TAMBIÉN
Documentación

zabbix_agentd(8), zabbix_get(1), zabbix_proxy(8), zabbix_sender(1),
zabbix_js(1), zabbix_agent2(8), zabbix_web_service(8)

Zabbix 2022-01-06 ZABBIX_SERVER(8)

zabbix_web_service

ZABBIX_WEB_SERVICE(8) Manual del administrador del sistema ZABBIX_WEB_SERVICE(8)

NOMBRE
zabbix_web_service - Servicio web de Zabbix

SINOPSIS
zabbix_web_service [-c config-file]
zabbix_web_service [-c config-file] -T
zabbix_web_service -h
zabbix_web_service -V

DESCRIPCIÓN
zabbix_web_service es una aplicación para proporcionar servicios web a
los componentes de Zab-
bix.

OPCIONES
-c, --config config-file
Utiliza el archivo de configuración alternativo config-file en
lugar del predeterminado.

-T, --test-config
Valida el archivo de configuración y sale.

-h, --help
Muestra esta ayuda y sale.

-V, --version
Muestra la información de versión y sale.

ARCHIVOS
/usr/local/etc/zabbix_web_service.conf
Ubicación predeterminada del archivo de configuración del
servicio web de Zabbix (si no se modificó durante el tiempo de
compilación).

VÉASE TAMBIÉN
Documentación

zabbix_agentd(8), zabbix_get(1), zabbix_proxy(8), zabbix_sender(1),
zabbix_server(8), zabbix_js(1), zabbix_agent2(8)

Zabbix 2022-01-06 ZABBIX_WEB_SERVICE(8)

Aviso de derechos de autor

La documentación de Zabbix NO se distribuye bajo la licencia AGPL-3.0. El uso de la documentación de Zabbix está sujeto a los siguientes términos:

Puede crear una copia impresa de esta documentación únicamente para su uso personal. Se permite la conversión a otros formatos siempre que el contenido real no se altere ni edite de ninguna manera. No debe publicar ni distribuir esta documentación en ninguna forma ni en ningún medio, excepto si distribuye la documentación de manera similar a como Zabbix la difunde (es decir, electrónicamente para su descarga en un sitio web de Zabbix) o en una memoria USB o medio similar, siempre que la documentación se difunda junto con el software en el mismo medio. Cualquier otro uso, como la difusión de copias impresas o el uso de esta documentación, total o parcialmente, en otra publicación, requiere el consentimiento previo por escrito de un representante autorizado de Zabbix. Zabbix se reserva todos los derechos sobre esta documentación que no se concedan expresamente anteriormente.